

# WEB APPLICATION DEVELOPMENT

PGDCA 202

## BLOCK 1: INTRODUCTION TO PHP

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# WEB APPLICATION DEVELOPMENT



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Mr. K. J. Sharma

### **Language Editor**

Mr. Vipul Shelke

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# WEB APPLICATION DEVELOPMENT

## Contents

---

### **BLOCK 1: INTRODUCTION TO PHP**

#### **UNIT 1 INTRODUCTION TO WEB APPLICATION**

The architecture of a web application, static and dynamic web application, Installing WAMP.

#### **UNIT 2 PHP BASICS**

Embedding PHP Code in Your Web Pages, Commenting Your Code, Outputting Data to the Browser, PHP's Supported Data types, Identifiers, Variables, Constants, Expressions, String Interpolation, Control Structures.

---

### **BLOCK 2: ARRAY, FUNCTION AND EXPRESSION**

#### **UNIT 1 ARRAY IN PHP**

Array introduction, creating an array, adding and removing array element, determining array size and uniqueness sorting array,

#### **UNIT 2 FUNCTION IN PHP**

In built functions: String functions, array functions, mathematical functions, File System functions, Date and Time Functions, Miscellaneous Functions, User Defined Functions, arguments passing by reference , Argument passing by value

#### **UNIT 3 FUNCTION IN PHP**

Regular Expression, Error Handling Regular Expressions, Exception handling



## **BLOCK 3: OO AND FILE HANDLING IN PHP**

### **UNIT 1 OBJECT ORIENTED PHP**

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

### **UNIT 2 FILE AND DIRECTORY HANDLING**

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

## **BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP**

### **UNIT 1 PHP AND MYSQL DATABASE**

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

### **UNIT 2 STATE MANAGEMENT IN PHP**

Session, Cookies





Dr. Babasaheb  
Ambedkar  
Open University

PGDCA 202

## Web Application Development

---

### BLOCK 1: INTRODUCTION TO PHP

---

#### UNIT 1

Introduction to Web Application

03

#### UNIT 2

PHP Basics

20

---

# **BLOCK 1: INTRODUCTION TO PHP**

---

## **Block Introduction**

Web application or services are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. PHP is Hypertext Pre-processor a programming language which is applied by web developers in order to frame dynamic content which is easy to interact with databases.

In this block, we will detail about the basic understanding of the architecture of web application with knowledge on static and dynamic web application. The block will focus on features of installation of WAMP with study about their characteristics. The concept of Identifiers, Variables, Constants and Expressions are also explained.

In this block, you will make to learn and understand about basic of outputting data to web browser and its techniques. The concept related to web application as interactive system allowing users to interact will also be explained to you. You will be demonstrated practically about String Interpolation.

## **Block Objective**

**After learning this block, you will be able to understand:**

- The architecture of a web application
- Knowledge about static and dynamic web application
- Features of installation of WAMP
- Idea about Outputting Data to the Browser
- Knowledge related to PHP Supported Data types
- Basic of Identifiers, Variables, Constants and Expressions
- Concept of Control Structures

## **Block Structure**

**Unit 1: Introduction to Web Application**

**Unit 2: PHP Basics**

---

# UNIT 1: INTRODUCTION TO WEB APPLICATION

---

## Unit Structure

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 The architecture of a web application**
- 1.3 Static and dynamic web application**
- 1.4 Installing WAMP**
- 1.5 Let Us Sum Up**
- 1.6 Answers for Check Your Progress**
- 1.7 Glossary**
- 1.8 Assignment**
- 1.9 Activities**
- 1.10 Case Study**
- 1.11 Further Readings**

---

## 1.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About architecture of web application
- About Static and dynamic web application
- About WAMP

---

## 1.1 Introduction

---

Web application or services are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. In this, the components can be internal or external to an organization, which are used to communicate using Internet-based protocols

which can be Hyper Text Transfer Protocol or else. These services run on servers and carried out large variety of data which users want to access quickly and easily.

As seen, the famous programming model where individual web services are joined or mixed in order to frame functional results in service-oriented architecture. It is an architectural model, where service consumer put on the requests to service provider using standard connection. In this, request and successive response are explained in a way which is understandable to consumer and provider.

Many web services make use of Extensible Markup Language (XML) in order to define format of request and response messages. This feature is tagged structure which shows required flexibility for changing of information that exists among disparate components.

Simple Object Access Protocol shows standard for persistence and sending of web service messages which is part of XML messaging specification that shows message format with certain rules applied in exchanging of data in required sequence that appears among structured data types and arrays.

---

## 1.2 The architecture of a web application

---

A web application is an interactive system which allows its users to do business logic which is inside a server and helps in seeing the results of particular logic using web browser on client workstation. In this the defining factor makes the system a web application where server and client are able to communicate over the Internet. Such web applications makes data to process using web services which is present for users for fast access using web browsers.

Web applications are built on client/server architecture where business logic is present in the application that works on web server and uses HTTP in order to communicate with clients over Internet. Here the web server helps the application by passing requests from clients to application which will further return with responses to the client.

On client side, web application can be visible using browser. The application's user interface with the use of HyperText Markup Language pages gets interpreted and displayed with the help of browser. Apart from this, such HTML pages carry web forms, image files, audio and video clips and several other displayable data types.

Fig 1.1 describes the basic architecture of web components and services which are commonly applied to host websites and build web applications.

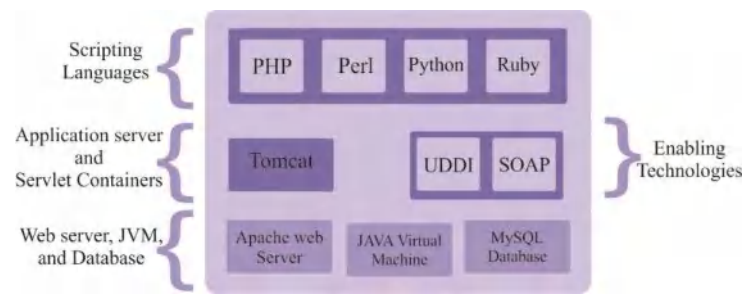


**Fig 1.1 Architecture of Web application**

The structure will be shown in fig 1.2

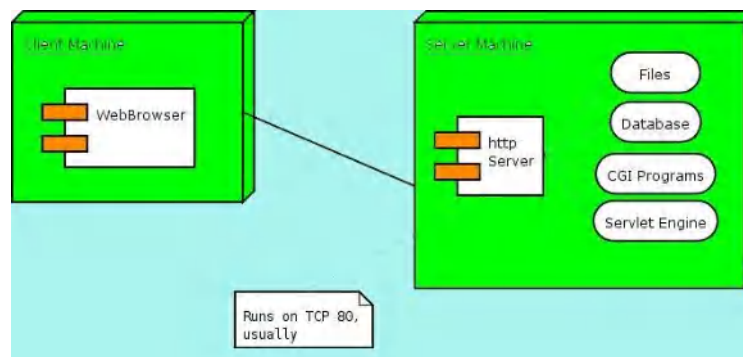


**Fig 1.2 Structure of Web application**

### Fetching Static Content

Client sends HTTP request like

GET /hostpage.html HTTP/1.1CRLF

Host: www.mypage.comCRLF

CRLF

Server response as:

HTTP/1.1 200 OKCRLF

Server: XXXCRLF

Date: XXXCRLF

Content-type: XXXCRLF

...CRLF

CRLF

<html>

...

</html>

Here the browser will read the response from server and provide the page. If the page has fixed content which can be images, then the client will request for it. In HTTP/1.0, requests will be on different connections where as HTTP/1.1 and later carries same connection.

### Server-Side Dynamic Content

In server side, applications are not so important if they just have static pages which can be responded by the user input. The server will construct the content of HTTP responses on fly in response to user input.

It is noted that fig 1.3 describes the Architecture Diagram of PHP web application that shows how every components in a system gets joined or connected among each other and how data flows across them.

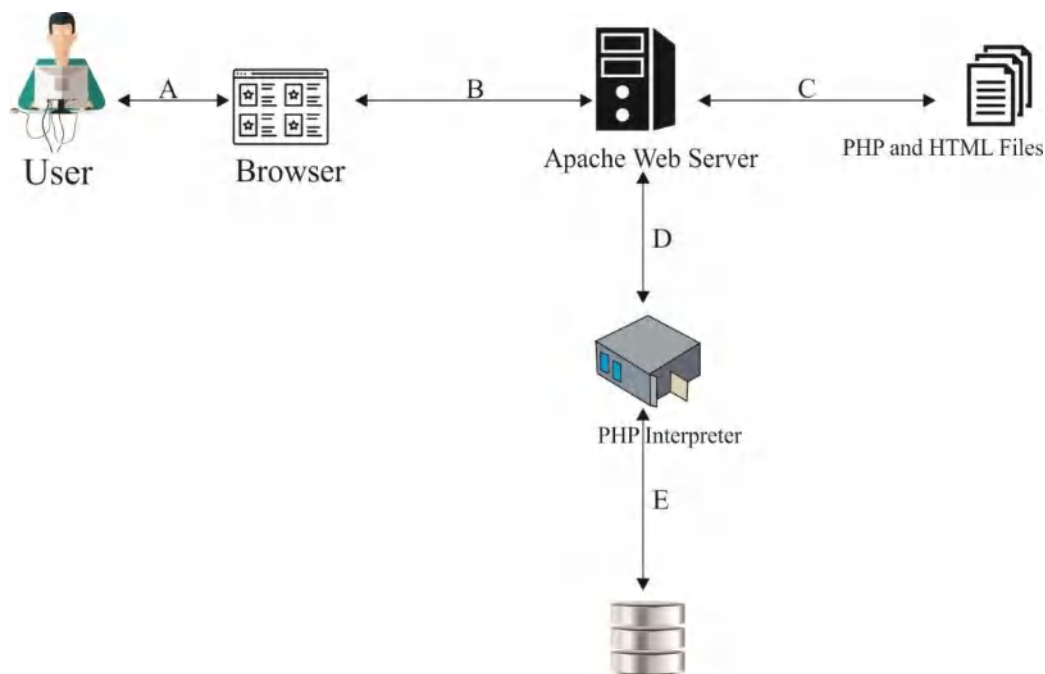


**Fig 1.3 PHP web application**

From the figure we see that initially the user will access the website using the browser where user types the URL of website in browser and click on go option.

Then the page request on browser will reach to Web Server.

After this, the web server will collect requested page from its document root.

When a static element such as HTML, CSS, image or Script file appears then web server will send it directly to browser and in such case the browser will leave it to user on screen.

When the PHP file appears then the web browser will send content of file to PHP Interpreter which further will interpret such code and start execute.

After executing the code, the PHP interpreter will generate an output and sends it to web server. Further, the web server will send the required content to browser which provides it to users' screen.

We see all static components which can be HTML, CSS files, Image Files, Scripts etc will not require interpreter. In such case the web browser will provide them and shows on screen.

Only if requested page is in PHP, then the web server will send it to PHP interpreter in order to do translation and execution. For such reason, those listed static components will remain inside the server and are consider as part of User Interface which provides at user's browser which is a Client side.

Due to this, the PHP files are referred to as Server side components since they carry out dependencies on other with this, we see that:

1. PHP files are kept on Server - Server Side
2. PHP Interpreter will interprets PHP language and perform calculation of instructions as per required code without any compilation.

### **Check your progress 1**

1. PHP scripts are executed on\_\_\_\_\_.
  - a. Client Side
  - b. Server side
  - c. Both of these
  - d. None of these
2. The extension of php file is\_\_\_\_\_.
  - a. html
  - b. .ph
  - c. .xml
  - d. .php

---

## 1.3 Static and dynamic web application

---

Web server process out every piece of information to users so as to compose the web pages. It is seen that web page structure can of many pieces, which result in sum that affects series of performance metrics which can be in terms of bandwidth consumption; user perceived latency without relevance to workload required so as to arrange all information of particular web pages together. It describes whether a web page needs large data processing or queries to permanent store so as to put together.

A web application's content can be of two types:

- Dynamic
- Static

Dynamic content is what needs some type of processing that can be applied where as static content is that that never changes. When a request is made for either type of content, in such case the web server will do execution which is required to send out dynamic content present in the file. This process is shown in figure 1.4.



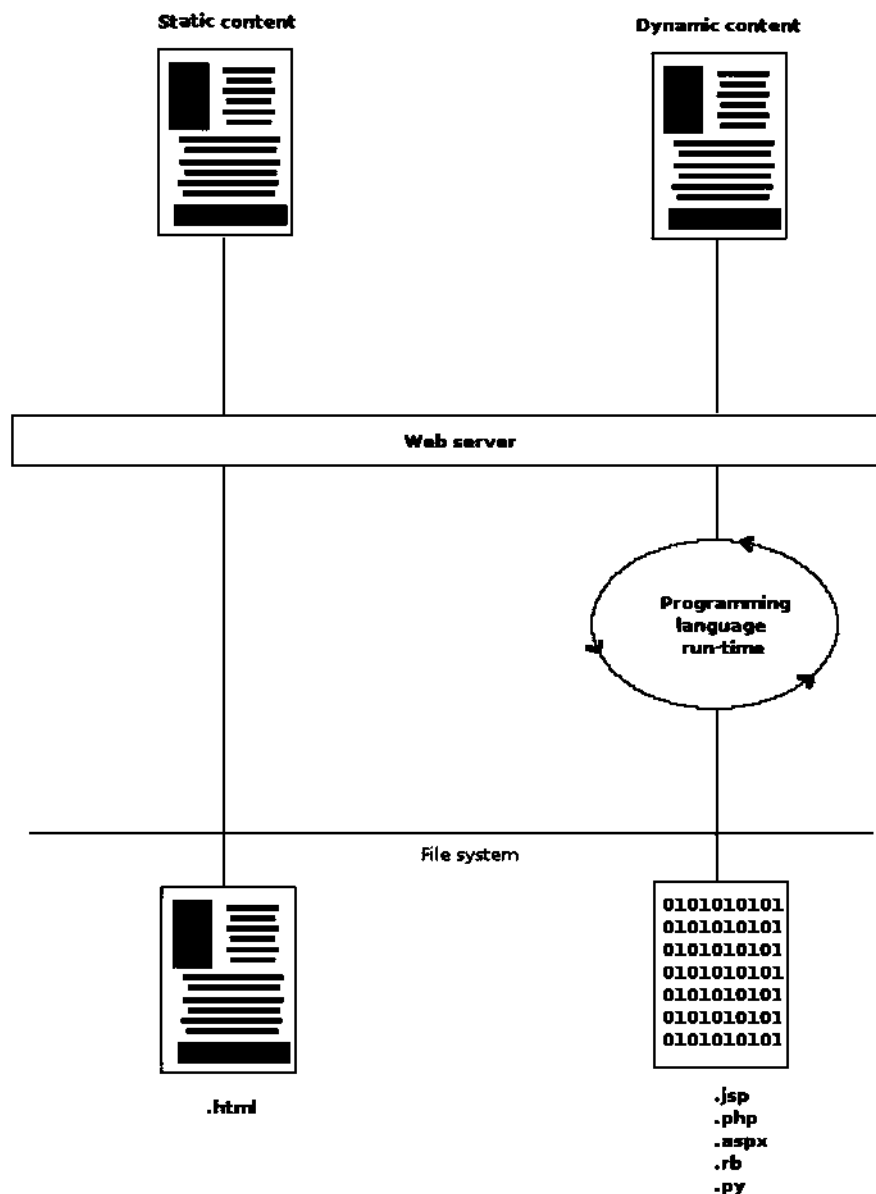


**Fig 1.4 Process execution**

It is noticed that the art of dispatching dynamic or static content will take place when a particular load on web server using dynamic content places heavier load as compared to static content. It describes inefficient allocation of resources on web server that is required to send dynamic and static content. So it is analysed that decoupling of static content from application is considered as steps in order to upgrade the performance on web servers. The problem with static content separation is not clear of having full static web page.

When a user's browser receives the original application page of the elements they are declared in the content where browser start requesting for referenced content. Figure 1.5 shows such multiple request process.

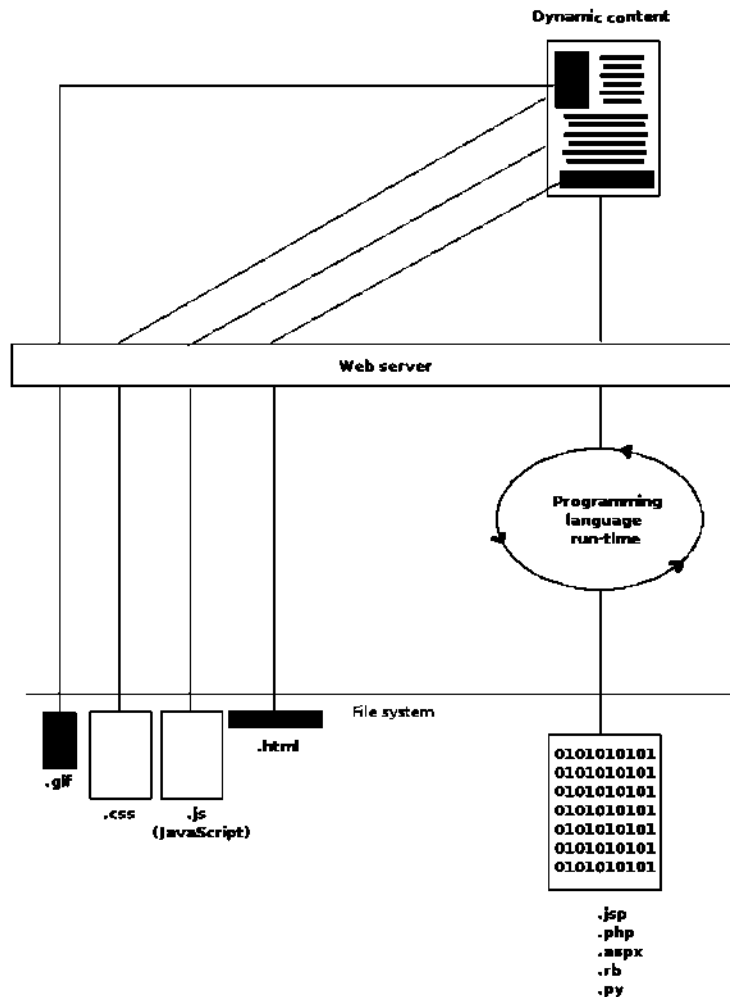


**Fig 1.5 Referenced content requests**

As observe in figure 1.5, the extra requests generates incremental load on web server. As this process is inevitable, it can be designed to be more performance friendly. Rather than having all content send across by same web server, referenced content can also be send by separate web server which alleviates potential web server load.

As seen, in the last listing, domain URLs applied in HTML elements for static content references which allow content to be decoupled and attended by a different web server on a different sub-domain. Thus lowering the burden on the primary web server used for original dynamic content. This process is illustrated in figure 1.6.

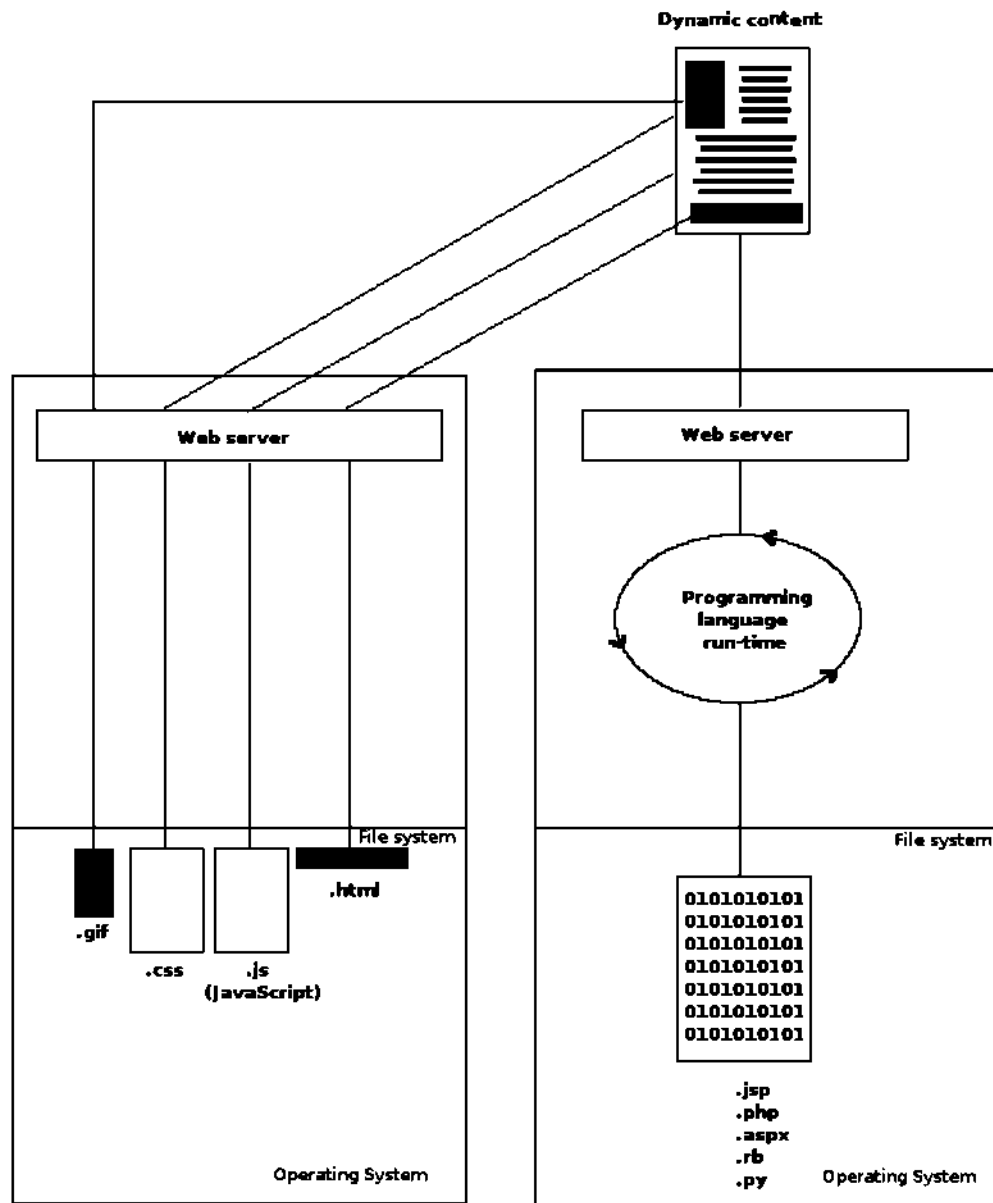


Fig 1.6 Different web servers for dynamic and static content.

## Check your progress 2

1. What is meant by dynamic web content?
  - a. Content that cannot change without a developer editing its source code,
  - b. Different content from the same source code.
  - c. Both of these
  - d. None of these

## 1.4 Installing WAMP

WAMP allows you to move between different versions of PHP if it is installed properly. There are certain steps to be followed while installing Wamp Server.

To start installing, you have to open the folder where you have saved your file and double click on installer file. After double clicking, a security warning window will open which will ask you whether to run this file or not. If you select on Run you will find that the installation process started and you will see Welcome To WampServer Setup Wizard screen on which you have choose next to forward with the installation.

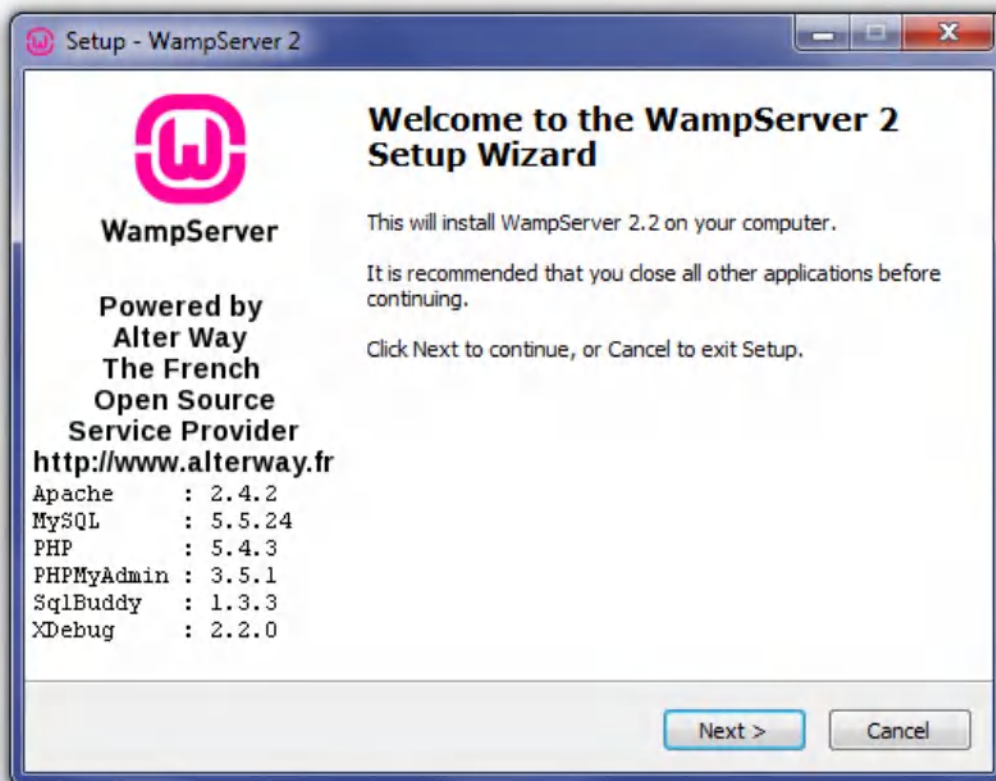


Fig 1.7 Welcome screen

After this, the License Agreement screen will appear which will ask you to accept all agreements by selecting the radio button. Selecting the radio button and clicking on next will allow moving to next screen.

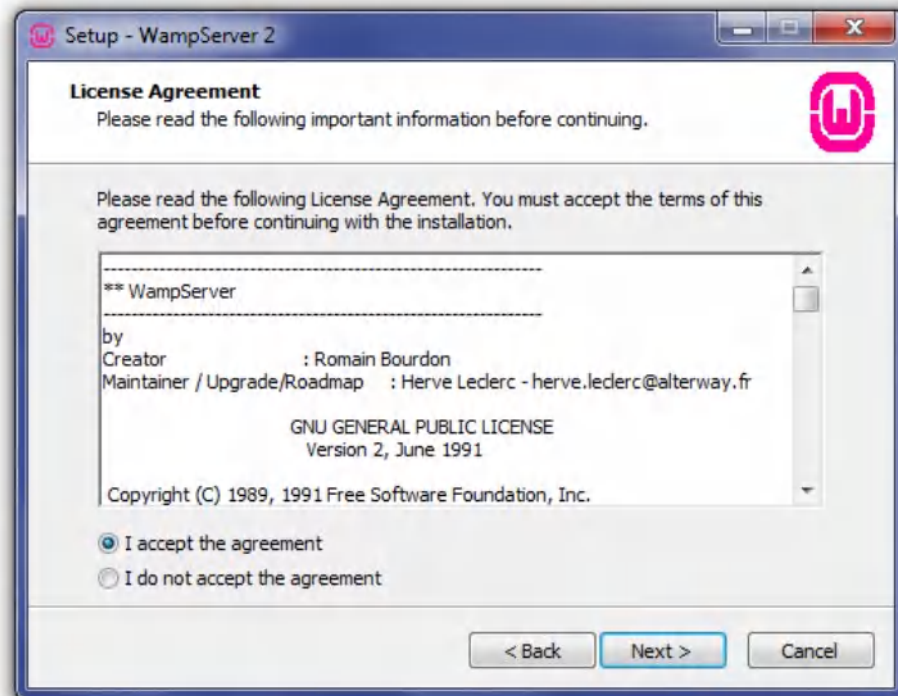


**Fig 1.8 License Agreements**

This screen will ask for the location where you have to select Destination Location. Unless you would like to install WampServer on another drive, you should not need to change anything. Click Next to continue.

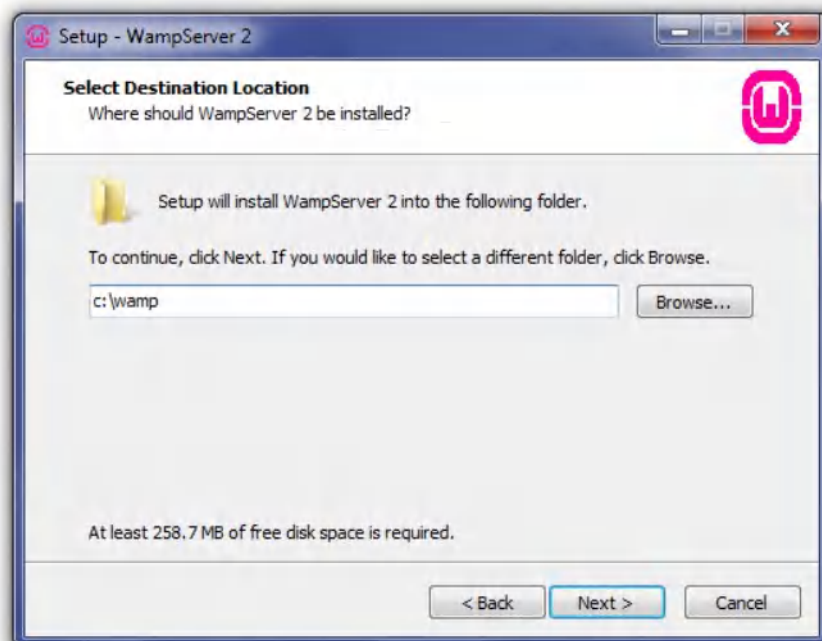


**Fig 1.9 Destination location**

After this, the next screen shows Select Additional Tasks screen where you need to select whether you would like Quick Launch icon to be added to taskbar or Desktop icon after the installation. After making the selections, click Next to continue.

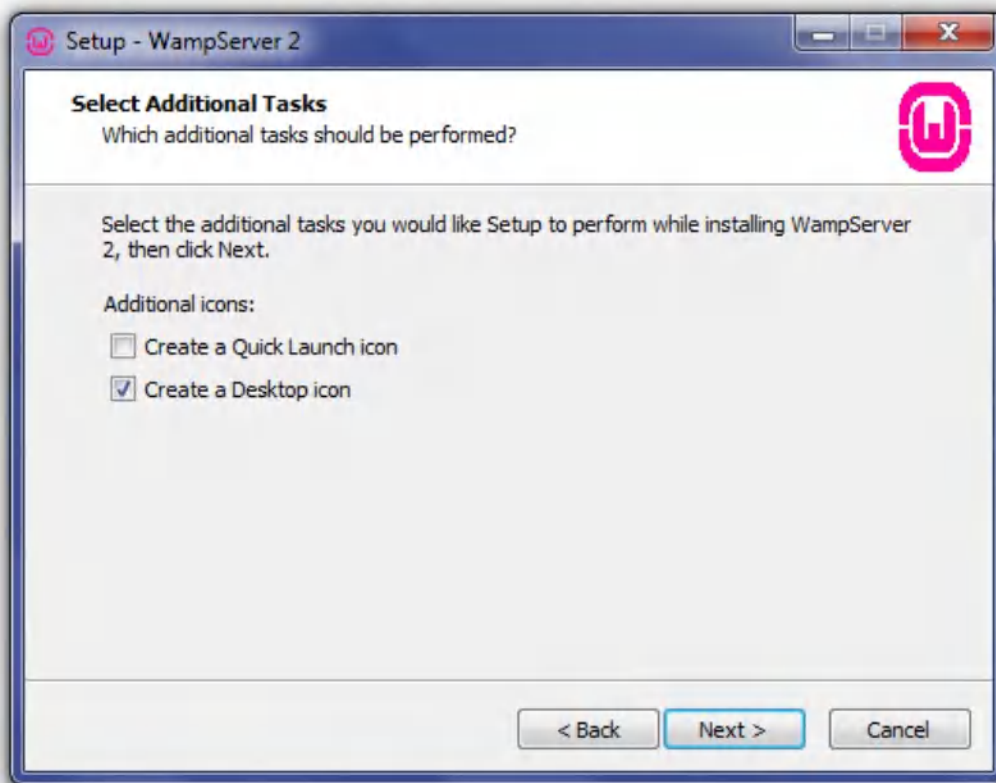


**Fig 1.10 Task selection**

Clicking on Next will take you to Ready To Install screen where you can review your setup choices and change any by clicking Back to required screen. Once you have reviewed your choices, click Install to continue.

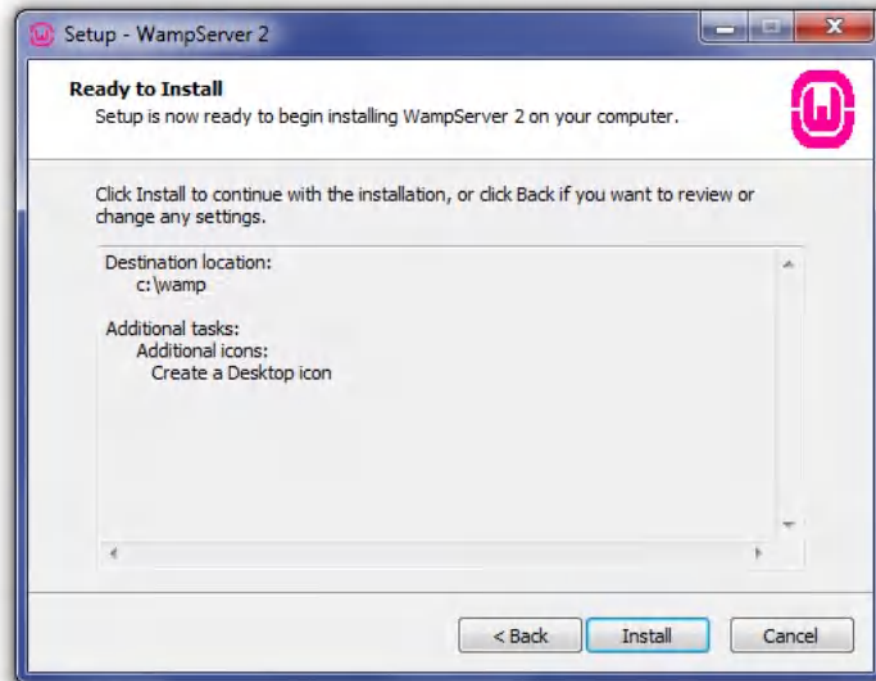


**Fig 1.11 Ready Install screen**

After this, the WampServer will start extracting files to required selected location.

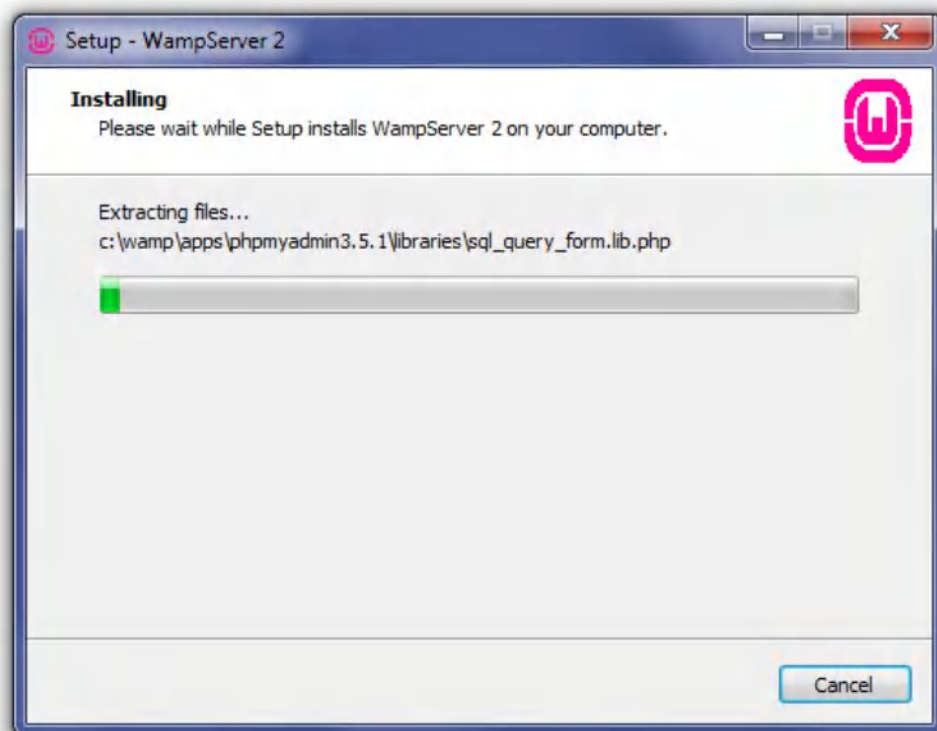


**Fig 1.12 Installation Screen**

After extraction of files, you will be asked to select default browser. Wamp Server defaults to Internet Explorer upon opening the local file browser window. If your default browser is not an Internet Explorer, then find required .exe file:

- Opera: C:\Program Files (x86)\Opera\opera.exe
- Firefox: C:\Program Files (x86)\Mozilla Firefox\firefox.exe
- Safari: C:\Program Files (x86)\Safari\safari.exe
- Chrome:  
C:\Users\xxxxx\AppData\Local\Google\Chrome\Application\chrome.exe

Now select default browser's .exe file and click Open to continue.

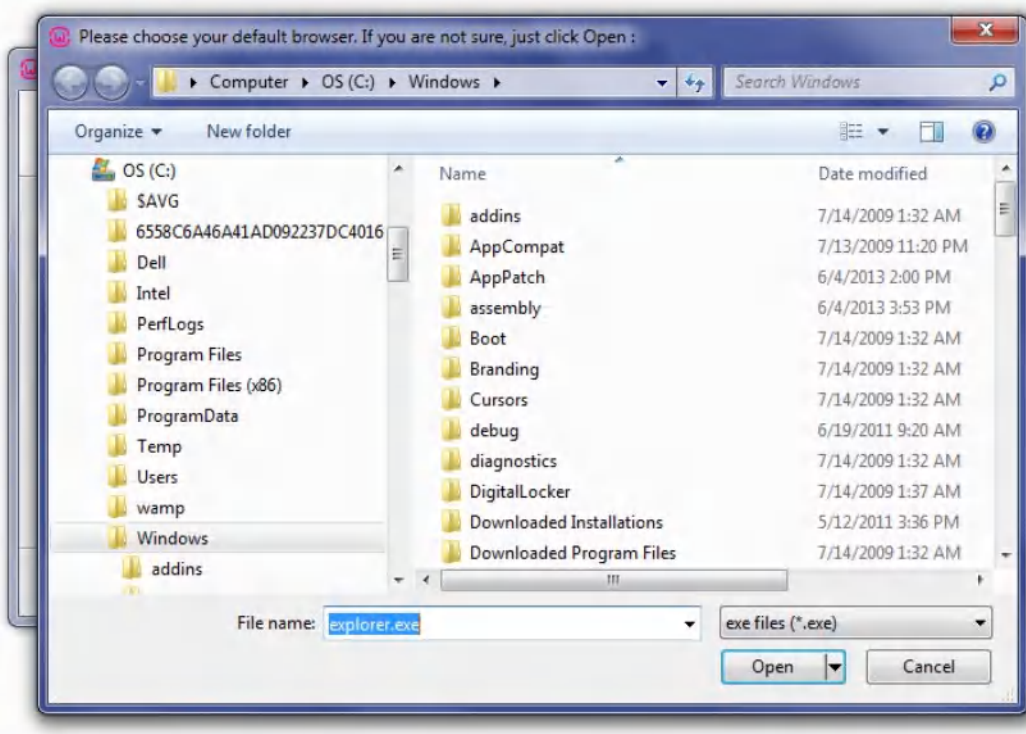


**Fig 1.13 Selecting default browser**

On clicking you will find that a Windows Security Alert window will appear prompting that Windows Firewall has blocked some features of program. Here you need to check whether you want to allow Apache HTTP Server to communicate on private or public network and if this and then click Allow Access. You will see that the setup screen will appear showing status of installation process.



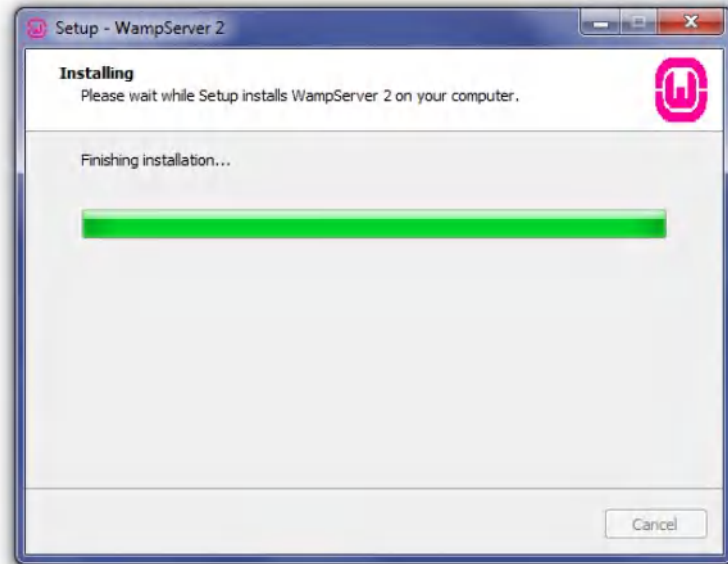


**Fig 1.14 Installing process**

You will that the progress bar will work and once it is completely green, then PHP Mail Parameters screen will appear. Here you have to leave SMTP server local host and change email address to one of your selection. Click Next to continue.

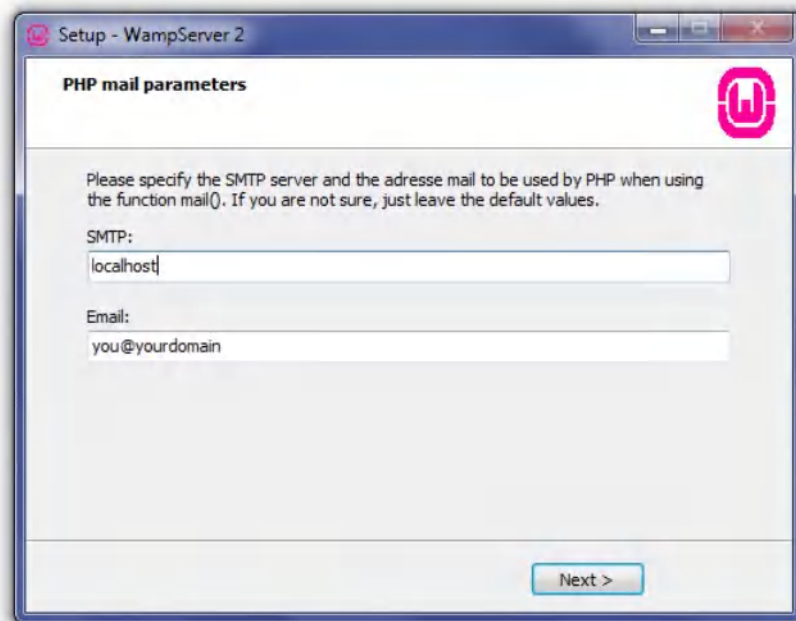


**Fig 1.15 Local host parameters**

After this you will find that the Installation complete screen will appear where you have to select Launch WampServer Now and click on Finish to complete installation.

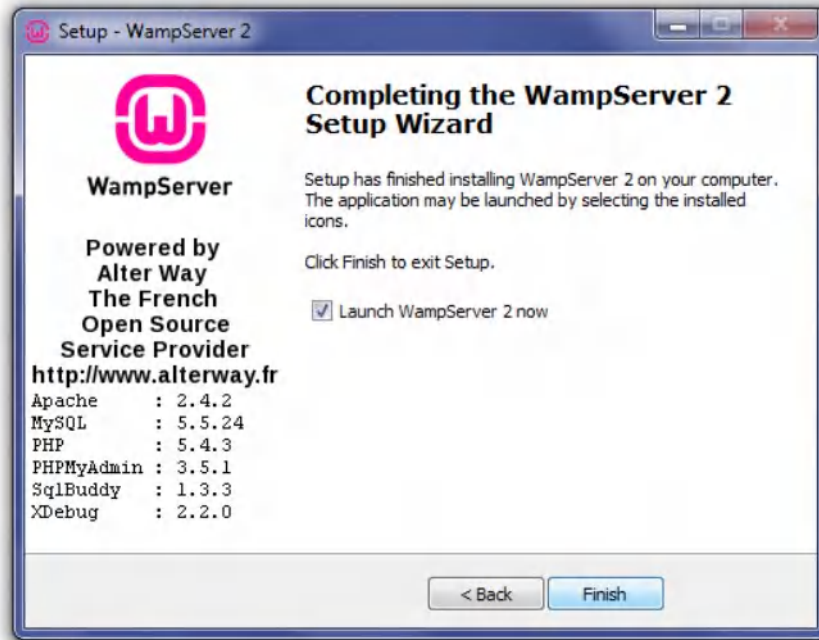


**Fig 1.16 Complete setup wizard**

Finally you will find WampServer icon in systray on right side of taskbar.

### **Check your progress 3**

1. Full form of WAMP is\_\_\_\_\_.
  - a. Windows, Apache, MySQL and PHP
  - b. Word, Adobe, MySQL and PHP
  - c. Windows, Apache Mainstream Program
  - d. None of these
2. Which of the following is needed to be installed on computer as to run PHPscript?
  - i) Adobe Dreamweaver
  - ii) PHP
  - iii) Apache
  - iv) IIS
  - a. All of these
  - b. Only ii)
  - c. ii) and iii)
  - d. ii), iii) and iv)

---

## 1.5 Let Us Sum Up

---

In this unit we have learnt that Web application are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. Simple Object Access Protocol shows standard for persistence and sending of web service messages which is part of XML messaging specification that shows message format with certain rules applied in exchanging of data in required sequence that appears among structured data types and arrays.

A web application is interactive system allowing its users to do business logic which is inside a server and helps in seeing the results of particular logic using web browser on client workstation. In server side, applications are not so important if they just have static pages which can be responded by the user input.

Web server process out every piece of information to users so as to compose the web pages. In WAMP, you can move between different versions of PHP if it is installed properly with various steps.

---

## 1.6 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –b), (2-d)

**Check your progress 2**

**Answers:** (1 –b)

**Check your progress 3**

**Answers:** (1 –a), (2-d)

---

## 1.7 Glossary

---

1. **Web application** - Services used for business logic components connected across with exchange data to do work.
2. **Simple Object Access Protocol** - Standard for persistence and sending of web service messages.

3. **Server side applications** - Applications performed at server end on static responded by user input.
4. **Web server** - Server running web applications data process information to users through web pages.
5. **WAMP** - Version in PHP where you can move in-between the process.

---

## 1.8 Assignment

---

Design a process to install PHP web application.

---

## 1.9 Activities

---

Create an activity on WampServer.

---

## 1.10 Case Study

---

Discuss about the architecture of a web application.

---

## 1.11 Further Readings

---

1. Web Applications in PHP, Jim Smith, Oxford, 2006
2. Applications in WAMP by Coulouris and Kindberg, 2010

---

## UNIT 2: PHP BASICS

---

### Unit Structure

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 Embedding PHP Code in Your Web Pages**
- 2.3 Commenting Your Code**
- 2.4 Outputting Data to the Browser**
- 2.5 PHP's Supported Data types**
- 2.6 Identifiers**
- 2.7 Variables**
- 2.8 Constants**
- 2.9 Expressions**
- 2.10 String Interpolation**
- 2.11 Control Structures**
- 2.12 Let Us Sum Up**
- 2.13 Answers for Check Your Progress**
- 2.14 Glossary**
- 2.15 Assignment**
- 2.16 Activities**
- 2.17 Case Study**
- 2.18 Further Readings**

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About PHP Code
- About Outputting Data to the Browser
- About Variables and Constants

---

## 2.1 Introduction

---

PHP which is Hypertext Preprocessor is a programming language which is applied by web developers in order to frame dynamic content which is easy to interact with databases. It is normally used for creating and developing web based software applications.

---

## 2.2 Embedding PHP Code in Your Web Pages

---

PHP is an important coding technique which can be easily embedded directly in HTML. For doing this, the page must be passed to PHP engine for interpretation. Since web server doesn't pass every web page but simply passes those pages which are located by particular file extension. By selectively passing certain pages to engine is inefficient for engine in order to consider every line as potential PHP command. So engine needs certain means to find at the same time which areas of page are PHP enabled. There are four delimitation of variants such as:

### Default Syntax

It is a default delimiter syntax which opens with `<?php` and results with `?>` as:

```
<h3>Hi!</h3>
```

```
<?php
```

```
echo "<p>K. J. Sharma</p>";
```

```
?>
```

```
<p>Engineer</p>
```

Output:

Hi!

K. J. Sharma

Engineer

### Short-Tags

It is a shorter delimiter syntax where syntax gives up php location that is needed in default syntax. For this you have to allow PHP's `short_open_tag` directive as shown:

```
<?
print "He is an Author";
?>
```

Since short-tag delimiters are simple and convenient so it is not applicable while creating PHP driven software which is needed for redistribution as this feature potentially gets disabled in php.ini file.

Such tag when enabled will allow you to quickly escape from PHP to output with small dynamic text thereby deleting statements using output variation:

```
<?=" He is an Author.";>
```

This is functionally equivalent to both variations:

```
<? echo " He is an Author."; ?>
```

```
<?php echo " He is an Author.";>
```

### **Script**

It is noted that many programmers has problems with PHP escape syntax variants, so support for another mainstream delimiter variant, <script>, is offered:

```
<script language="php">
print " He is an Author.";
</script>
```

### **ASP Style**

If you are having experience on ASP, the delimiting strategy/escape syntax PHP supports it as shown:

```
<%
print " He is an Author.";
%>
```

It is noted that ASP Style and Script delimiting variants are not commonly used and should be avoided till you have reasons for doing.

### **Embedding Multiple Code Blocks**

You can escape to and from PHP as many times as needed within a page. Take an example:

```
<html>
```

```
<head>
<title><?php echo "Academic Solution";?></title>
</head>
<body>
<?php
$date = "21th September 2015";
?>
<p>Today's date is <?=28th Feburary 2016?></p>
</body> </html>
```

It is noted that any variables can be declared a prior code block that are remembered for later blocks.

### Check your progress 1

1. Choose the correct way of writing short-open tag.

- a. <? echo"hello" ?>
- b. <% echo"hello" ?>
- c. <? echo"hello" ?>
- d. None of these

2. Choose the correct way of writing ASP tag.

- a. <? echo"world" ?>
- b. <% echo"world" %>
- c. <? echo"world" >
- d. None of these

---

## 2.3 Commenting Your Code

---

Comment in PHP code is a line which is not read as part of program but to read by person editing the code. If you are working with team using your script, then comments allows the programmers to understand what you were doing during the process each time. It makes the easier for programmers to work and can do editing if require.



It is noted that programmers having experience in editing their work and can re-figure will introduce comment just to remind about thinking in order to avoid decoding work.

You can add comments in PHP code by many ways. Initially you can use `//` to comment a line as shown:

```
<?php echo "Hi"; //this is a comment echo " How are you"; ?>
```

If you have single line comment, so there is another option of introducing comment by using a `#` sign as shown:

```
<?php echo "Hi"; #this is a comment echo " How are you"; ?>
```

If you have no multi line comment, then you can comment by using `/* */` as shown in example:

```
<?php echo "Hi"; /* Using this you can create big block of text that comments as */ echo " How are you"; ?>
```

### Check your progress 2

1. We can write comment in PHP by using\_\_\_\_\_ .
  - a. #
  - b. /\* \*/
  - c. //
  - d. All of these

---

## 2.4 Outputting Data to the Browser

---

It is seen that in PHP, the data can be outputted to the browser with various methods which are discussed below:

### The print() Statement

The `print()` statement will outputs data when passed to it which looks like:

```
int print(argument)
```

All of following are plausible `print()` statements:

```
<?php
print("<p>Hello World</p>");
?>
```

```
<?php
$author = "K. J. Sharma";
print "<p>I like $author.</p>";
?>
```

The output of above statements will be:

I like K. J. Sharma.

We see that official syntax calls for use of parentheses will enclose the argument, and are not applied as `print()` is not technical function but language construct. The `print()` statement's will return false value as it return 1 immaterial of outcome.

### **The echo() Statement**

Apart from `print`, you can use `echo()` statement for similar work. But both `echo()` and `print()` statements are different and is not of many used to readers and so are avoided. The example of `echo()`'s prototype will be:

```
void echo(string argument1 [, ...string argumentN])
```

To use `echo()`, provide with:

```
echo " I like K. J. Sharma.";
```

From the prototype, `echo()` is able to output many strings. Study the example shown:

```
<?php
$he = "Rohit";
$she = "Vandana";
echo $he, " and ", $she, " are great couple.";
?>
```

This code produces the following:

Rohit and Vandana are great couple.

On executing following variation of above syntax will generate output as:

```
echo "$he and $she are great couple.";
```

### **The printf() Statement**

The printf() statement is good in order to blend output with static text and dynamic information which is kept in one or more variables. This statement separates static and dynamic data in two different sections which allow easy maintenance and also does large control over dynamic information required to screen in terms of type, precision, alignment and position as shown:

```
integer printf(string format [, mixed args])
```

To insert single dynamic integer value otherwise static string:

```
printf("what is the time %d hours", 21);
```

On executing this command will produce:

What is the time 21 hours

We see that %d is placeholder which is type specifier and d is an integer value that is kept in that position. When printf() statement work, lone argument, 100, will be inserted in placeholder.

### **The sprintf() Statement**

The sprintf() statement is similar in function as printf() but in this output assigns to string instead of rendering to browser as shown:

```
string sprintf(string format [, mixed arguments])
```

An example follows:

```
$value = sprintf("$%.2f", 55.5); // $value = $55.50
```

### **Check your progress 3**

1. What is the difference between printf and sprint?
  - a. Both are same
  - b. Both are different
  - c. Outputs are assigned to strings in sprintf, Unlike printf in which it is assigned to browser.
  - d. None of these

2. What is the difference between echo and print?

- a. Echo has no return value while print has a return value of 1
- b. Echo can take multiple parameters while print can take one argument.
- c. Echo is faster than print.
- d. All of these

---

## 2.5 PHP's Supported Data types

---

Variables store data of many types which does different work. PHP supports following types of data types:

- String
- Integer
- Float (floating point numbers - also called double)
- Boolean
- Array
- Object
- NULL
- Resource

### PHP String

A string is sequence of characters that can be "K. J. Sharma" which is inside quotes or double quotes as shown in example:

```
<?php
```

```
$x = "K. J. Sharma";
```

```
$y = 'K. J. Sharma';
```

```
echo $x;
```

```
echo "<br>";
```

```
echo $y;
```

```
?>
```

Output

K. J. Sharma

K. J. Sharma

### **PHP Integer**

The integer data type is without decimal that ranges from -2,147,483,648 and 2,147,483,647 and carries certain rules:

- An integer should have at least 1 digit
- An integer should have 1 decimal point
- An integer can be positive or negative
- An integer specifies formats such as decimal (10-based), hexadecimal (16-based) or octal (8-based)

The PHP `var_dump()` function returns the data type and value as shown in function example:

```
<?php
$x = 1234;
var_dump($x);
?>
```

Output:

```
Int(1234)
```

### **PHP Float**

Float is a floating point number having decimal point or exponential form. In an example shown, `$x` is float where PHP `var_dump()` function returns data type as:

```
<?php
$x = 12.345;
var_dump($x);
?>
```

Output:

```
float(12.345)
```

### **PHP Boolean**

A Boolean has only two values which can be TRUE or FALSE.

```
$x = true;
```

```
$y = false;
```

Booleans are often used in conditional testing.

### PHP Array

Array stores many values in single variable. From the example, \$scooters is an array which returns PHP var\_dump() function as:

```
<?php
$cars = array("Bajaj", "Priya", "LML");
var_dump($scooters);
?>
```

Output:

```
array(3) { [0]=> string(5) "Bajaj" [1]=> string(3) "Priya" [2]=> string(6) "LML" }
```

### PHP Object

Object are data type that keep data and information about data processing. It is noted that in PHP, object should clearly be declared. Initially we have to declare a class of object. A class is a structure that has properties and methods such as:

```
<?php
class Scooter {
    function Scooter() {
        $this->model = "Priya 112";
    }
}

// create an object
$herbie = new Scooter();

// show object properties
echo $herbie->model;

?>
```

Output:

```
Priya 112
```

## PHP NULL Value

Null is special data type having single value: NULL. A variable of data type NULL is variable having no assigned value. If variable is without a value, it is directly given with NULL value. Example

```
<?php  
$x = "Name!";  
$x = null;  
var_dump($x);  
?>
```

Output:

null

## PHP Resource

It is a different type of resource which is not real data type which stores reference to functions and resources that is external to PHP.

### Check your progress 4

1. How can we declare a variable in PHP?
  - a. \$num
  - b. #num
  - c. \$21
  - d. None of these
2. What is resource in PHP?
  - a. It is special data type having value: NULL
  - b. They are special variables that hold references to resources external to PHP
  - c. are instances of programmer-defined classes
  - d. All of these

---

## 2.6 Identifiers

---

Identifier is a name used to name variables, functions, constants and classes. In this the first character be either ASCII letter, underscore or any characters between ASCII 0x7F and ASCII 0xFF. After initial character, such characters and digits 0-9 are valid.

Variable names

Variable names begin with dollar sign (\$) and are case-sensitive such as:

\$copy

\$head\_count

\$MaxArea

\$I\_HEART\_PHP

\$\_self

\$\_int

There exist certain uncommon variable names like:

\$not valid

\$|

\$3wa

Function names

Function names are not case-sensitive such as:

tally

list\_all\_users

deleteTclFiles

Class names

Class names have standard rules for identifiers that are not case-sensitive like:

People

Account

Constants

Constants are identifier for simple scalar values which can be Boolean, integer, double and string constants.



## Keywords

It is a reserved word by language for its main function that cannot assign variable, function, class or constant.

### Check your progress 5

1. Which of the following statement is true?
  - a. Variable names are not case sensitive.
  - b. Function names are not case sensitive.
  - c. Class names are case sensitive.
  - d. All of these.

---

## 2.7 Variables

---

A variable in PHP serves as containers which used for keeping information. It starts with \$ sign followed by name of variable as shown in example:

```
<?php
```

```
$txt = "Solving Integers";
```

```
$x = 11;
```

```
$y = 16.7;
```

```
?>
```

Output:

Solving Integers

11

16.7

On executing above program, variable \$txt will keep the value Solving Integers, where variable \$x will keep value 11 while variable \$y keep value 16.7. Variable have short name or can be descriptive.

Rules for PHP variables:

- Variable starts with \$ sign followed by name of variable
- Variable name to begin with letter or underscore

- Variable name cannot begin with number
- Variable name only has alpha-numeric characters and underscores
- Variable names are case-sensitive

### Output Variables

The PHP echo statement is mainly applied to output data to screen as shown in example:

```
<?php  
$txt = "K. J. Sharma Books";  
echo "I like!";  
?>
```

Output:

I like K. J. Sharma Books

The program shown will generate sum of two variables, as shown:

```
<?php  
$x = 8;  
$y = 8;  
echo $x + $y;  
?>
```

Output:

16

In above example we see that we have not instructed PHP about data type. PHP will directly convert variable to correct data type as per its value.

### PHP Variables Scope

In PHP, variables are declared in the script. The scope of variable is part of script where variable can be used. PHP has three different variable scopes:

- local
- global
- static

### Global and Local Scope

A variable declared outside function has GLOBAL SCOPE and can be accessed outside a function as seen in example:

```
<?php
$x = 10; // global scope
function myTest() {
    // using x inside this function will generate an error
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();
echo "<p>Variable x outside function is: $x</p>";
?>
```

Output:

Variable x inside function is:

Variable x outside function is: 10

A variable that is declared in a function contains LOCAL SCOPE which can be accessed in a function as shown in example:

```
<?php
function myTest() {
    $x = 12; // local scope
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();
// using x outside the function will generate an error
echo "<p>Variable x outside function is: $x</p>";
?>
```

Output:

Variable x inside function is: 12

Variable x outside function is:

The global Keyword

The global keyword is applied in order to access global variable within a function. To do this, use global keyword before the variables such as:

```
<?php
$x = 12;
$y = 18;
function myTest() {
    global $x, $y;
    $y = $x + $y;
}
myTest();
echo $y; // outputs 30
?>
```

Output:

30

It is noted that PHP stores global variables in array \$GLOBALS[index] where index keeps name of variable. Such array is accessible in functions and applied to adjust global variables as shown in example:

```
<?php
$x = 9;
$y = 19;
function myTest() {
    $GLOBALS['y'] = $GLOBALS['x'] + $GLOBALS['y'];
}
myTest();
echo $y; // outputs 28
?>
```

Output:

28

The static Keyword

In PHP when functions executes, all variables gets vanished. The local variables are required for further work as shown in example:

```
<?php
function myTest() {
    static $x = 5;
    echo $x;
    $x++;
}
myTest();
myTest();
myTest();
?>
```

Output:

5  
6  
7

Here every time the function is called with variable having information contained from last when function was called.

**Check your progress 6**

1. What is the output of following code?

```
<?php
$num1 = 7;
$num2 = 15;
function test() {
    global $num1, $num2;
    $num2 = $num2 + $num1;
}
test();
echo $num2;
?>
```

- a. 8
- b. -8
- c. 0
- d. None of these

2. What is the output of the following code?

```
<?php
$a = 'alen';

$b = "My name is $a";
echo $b;
?>
```

- a. "My name is \$a"
- b. ""My name is alen"
- c. My name is alen
- d. None of these

---

## 2.8 Constants

---

Constants are variables which cannot be defined unless changed or undefined. It is an identifier (name) for simple value which cannot be changed during script. So valid constant name begins with letter or underscore. To create a constant, you require to define() function.

Syntax

```
define(name, value, case-insensitive)
```

Parameters:

name: Specifies name of constant

value: Specifies value of constant

case-insensitive: Shows whether constant name should be case-insensitive where default is not correct. The example below will form constant with case-sensitive name such as:

```
<?php
define("GREETING", "Welcome to my Library of Books!");
echo GREETING;
?>
```

Output:

Welcome to my Library of Books|

Further we see that an example below describes case-insensitive name:

```
<?php
define("GREETING", "Welcome to my Library of Books!", true);
echo greeting;
?>
```

Output:

Welcome to my Library of Books!

Constants are direct global applied across the script. The example uses constant in a function, even if it is defined outside the function:

```
<?php
define("GREETING", " Welcome to my Library of Books!");
```

```
function myTest() {  
    echo GREETING;  
}  
myTest();  
?>
```

Output:

Welcome to my Library of Books!

### Check your progress 7

1. What is the output of the following code?

```
<?php  
define("HELLO", 100);  
echo HELLO;  
?>
```

- a. Hello
- b. HELLO
- c. 100
- d. None of these

---

## 2.9 Expressions

---

Expressions are bits of PHP which calculates to generate a value. It carries simple expressions having literal values and variables that shows literal value calculates it as variable calculates value stored in variable.

In PHP, expressions are important building stones which can be applied for writing an expression. The simplest expression is anything that has value. The basic feature of expressions results are constants and variables.

It is noted that when you type "\$a = 10", then it shows that '10' assigns to \$a. '10', has value 10, or we see that '10' is an expression with value of 10.



After an assignment, \$a's value as 10, hence if we wrote \$b = \$a, then \$b = 10. In other words, \$a is expression with value 10. It is noted that PHP takes expressions much further in same way as other languages. In an example '\$a = 10' it is noted that there exists two values, value of integer constant '10' and value of \$a that updates to 10 as well. In practice, we see that '\$a = 10' is an expression having value 10. So writing like '\$b = \$a = 10' is like '\$a = 10; \$b = 10;'. As assignments are arranged from right to left, so we can write '\$b = \$a = 10'.

### Check your progress 8

1. Expression for adding two number is\_\_\_\_\_.
  - a. a+b
  - b. \$a+ \$b
  - c. @a+@b
  - d. None of these

---

## 2.10 String Interpolation

---

In PHP when it is required to control code to be worked and should want to beautify code, in such case these string interprets which provide with complete flexibility while working with string values such as

The \$car moved over the breaker.\n

In this, we see that \$car is a variable and \n is newline character so both be interpreted. If you want to output string exactly as written or want newline to render but want variable to display in literal form(\$car) or vice versa.

### Double Quotes

Strings in double quotes are applied in PHP scripts as they give flexibility since both variables and escape sequences are parsed as shown in example:

```
<?php
$book = "Electronics";
echo "Sanjay's favourite book is $book.";
?>
```

This example returns the following:

Sanjay's favorite book is Electronics.

### Single Quotes

Keeping a string in single quotes is important while interpreted string exactly similar as stated that has variables and escape sequences not interpreted when string is parsed as shown:

```
print 'This string will $print exactly as it\'s \n declared.';
```

It shows output as:

This string will \$print exactly as it's \n declared.

We see that single quote gets out which delete backslash escape character and result in syntax error as shown:

```
print 'Find other string.\\';
```

It shows output as:

Find other string.\

In above example we see that backslash appears at conclusion of string which has escaped else PHP will parser that understand single quote that escaped.

```
print 'Find \another string.';
```

It shows output as:

Find \another string.

### Check your progress 9

1. Choose correct statement to display the line as it is Hello World.HaVe a nIce dAy
  - a. echo “ Hello World..HaVe a nIce dAy”
  - b. echo ‘Hello World.HaVe a nIce dAy’
  - c. Syntax error
  - d. None of these

---

## 2.11 Control Structures

---

In PHP, control structures are loops, conditionals and if which are familiar to javascript programmer such as:

- if and else
- elseif
- while
- do...while
- for
- foreach
- break and continue
- switch
- Reference

### **if and else**

It is same as javascript like:

```
if ($a==$b) {  
    print "The same";  
}  
  
else {  
    print "They aren't the same";  
}
```

### **elseif**

elseif also else if gives a second if-statement where first if-condition if met run other elseif conditions even if not run.

```
$a="Sanjay";  
  
if($a=="Sanjay"){  
    print"Hello Sanjay";}  
  
elseif($a=="Sharma"){  
    print"Hello Sharma";
```

```

}
else if ($a=="Sanjay"){
print "Hello Sanjay 2";
}
else {
print"I don't know you!";
}
//result: Hello Sanjay

```

### **while**

While loop is similar as in javascript:

```

$i=1;
while ($i<10){
print "Hello";
$i++;
}

```

### **do..while**

do..while loop is same as while loop but while loop will not run if condition does not exists. The do..while loop always run at least once as shown in example:

```

$i=1;
do{
print "The number is $i";
}
while($i>100);
//result: The number is 1

```

### **for**

The for loop in PHP is same as in javascript's:

```

for($i=1;$i<=2;$i++){
print "This is $i<br>";
}

```

```
//This is 5
```

```
//This is 6
```

The for loop can be written in various ways. One additional way is as follows:

```
for($i=1;$i<=3;print "This time it's $i ",$i++);
```

```
//result: This time it's 5 This time it's 6 This time it's 7
```

### **foreach**

This control structure access items in array:

```
$arr=array("cow","deer","gost");
```

```
foreach($arr as $value){
```

```
echo"Value: $value<br>\n";
```

```
//Value: cow
```

```
//Value: deer
```

```
//Value: gost
```

This example shows the keys and values of an associative array:

```
$a=array("fruit"=>"pineapple","meat"=>"gost","vegetables"=>"beans","sweet"=>"ice cream");
```

```
foreach($a as $k=>$v){
```

```
print"\$a[$k]=>$v.\n";
```

```
};
```

```
//result:   $a[fruit]=>pineapple.   $a[meat]=>gost.   $a[vegetables]=>beans.  
$a[sweet]=>ice cream.
```

### **break and continue**

Break and continue structures are common in javascript that exits for, while, or switch structure and exits in loop and begins with next.

### **switch**

The switch statement will continue as shown which print all statements after finding as true as shown:

```
$i=1;
```

```
switch($i){ //variable in brackets and cases in curly brackets
```

```
case 0:
```

```
44
```

```
print "zero";  
break;  
case 1:  
print "one";  
break;  
default: //You can do default case!  
print "whatever it is, it isn't 0, 1 or 2";  
}  
//result: one
```

### Check your progress 10

1. What is the output of the following code?

```
<?php  
$a = 25;  
if ($a < "30") {  
    echo "Hello World!";  
}  
?>
```

- a. Hello World!
- b. 25
- c. 30
- d. None of these

2. What is the output of the following code?

```
<?php  
$x = "1";  
switch ($x)  
{  
case 1:  
    print "hello";
```

```
case 2:  
    print "to";  
default:  
    print "world";  
}  
?>
```

a. helloworld  
b. hellotoworld  
c. syntax error  
d. None of these

---

## 2.12 Let Us Sum Up

---

In this unit we have learnt that PHP is Hypertext Preprocessor applied by web developers to frame dynamic content which is easy to interact with databases. It is found that PHP is an important coding technique that can be easily embedded in HTML which uses page to be passed through PHP engine for interpretation.

It is known that comment in PHP code is a line that is not part of program but to read by person editing the code. Identifier are name used to name variables, functions, constants and classes where first character is ASCII letter, underscore or characters between ASCII 0x7F and ASCII 0xFF.

It is noted that constants are variables that are not defined unless changed or undefined which serves as an identifier for simple value that cannot be changed during script. As seen, expressions are bits of PHP that calculates value having simple expressions with literal values and variables.

---

## 2.13 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –c), (2-b)

**Check your progress 2**

**Answers:** (1 –d)

**Check your progress 3**

**Answers:** (1 –c), (2-d)

**Check your progress 4**

**Answers:** (1 –a), (2-b)

**Check your progress 5**

**Answers:** (1 –b)

**Check your progress 6**

**Answers:** (1 –a), (2-c)

**Check your progress 7**

**Answers:** (1 –c)

**Check your progress 8**

**Answers:** (1 –b)

**Check your progress 9**

**Answers:** (1 –b)

**Check your progress 10**

**Answers:** (1 –a), (2-b)

---

## 2.14 Glossary

---

1. **Variables** - They are containers used for keeping information.
2. **Constants** - Variables that are not defined till it changes or undefined and also serves as identifier.
3. **Expression** - These are bits in PHP that finds generated value.



---

## 2.15 Assignment

---

Describe the structure of control statements.

---

## 2.16 Activities

---

Write short note on PHP's Supported Data types.

---

## 2.17 Case Study

---

How to embed PHP Code in Web Pages?

---

## 2.18 Further Readings

---

1. Programming with PHP, Programmer's Guide by Ronald.
2. Building Secure PHP Apps by Coulouris, Dollimore, Kindberg.

---

## **Block Summary**

---

In this block, you will understand about the basic of identifiers with concept about ASCII letter, underscore or characters as ASCII 0x7F and ASCII 0xFF. The block gives an idea on architecture of PHP supported data types with study about their working characteristics. The examples related to Control Structures and Comments in PHP code are also discussed.

In this block, you will understand about the basic of remote procedure calls and its techniques. The concept related to basic of Identifiers, Variables, Constants and Expressions are also detailed. You will be demonstrated practically about important coding technique that can be easily embedded in HTML.

---

## Block Assignment

---

### Short Answer Questions

1. Write a short note on PHP datatypes.
2. Define the architecture of a web application.
3. Explain decision making in PHP using examples.

### Long Answer Questions

1. Discuss PHP constants and variables in detail with the help of examples.
2. Write a PHP code to add two numbers and display the output.
3. Write a program to check whether the given number is odd or even using if-else loop.
4. How static web pages are different from dynamic web page? Explain in detail

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                              | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|----------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| <b>Presentation Quality</b>                        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Language and Style</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Illustration used<br/>(Diagram, tables etc)</b> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Conceptual Clarity</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Check your progress<br/>Quest</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Feed back to CYP<br/>Question</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# MOBILE APPLICATION DEVELOPMENT

PGDCA 203

## BLOCK 1: BASICS OF ANDROID APPLICATION



**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**

# MOBILE APPLICATION DEVELOPMENT

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'





## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# Mobile Application Development

## Contents

---

### **BLOCK 1:      BASICS OF ANDROID APPLICATION**

#### **UNIT 1            INTRODUCTION TO ANDROID, TOOLS AND BASICS**

The Android Platform, Installing Android Studio, Java for Android, Android Studio for Android Software Development, Building a sample Android application

#### **UNIT 2            ANDROID APPLICATION DESIGN ESSENTIALS - I**

A Framework for a Well-Behaved Application, Application Context, Activities, Services, Intents, Intent Filter, Permissions, Receiving and Broadcasting Intents,

---

### **BLOCK 2:      ANDROID APPLICATION AND USER INTERFACE DESIGN**

#### **UNIT 1            ANDROID APPLICATION DESIGN ESSENTIALS - II**

Using Intent Filter, Permissions, Android Manifest File and its common settings, managing different types application resources in a hierarchy

#### **UNIT 2            Android User Interface Design and Common APIs**

User Interface Screen elements, Designing User Interfaces with Layouts, Drawing and Working with Animation ,Drawing 2D and 3D Graphics and Multimedia,



**BLOCK 3:      ANDROID NETWORKING AND DEVELOPMENT**

**UNIT 1      ADVANCED TOPICS - I**

Android Networking, Web and Telephony API, Search, Location and Mapping, Sensors, NFC, Speech, Gestures, and Accessibility,

**UNIT 2      ADVANCED TOPICS - II**

Communication, Identity, Sync, and Social Media, The Android Native Development Kit (NDK)

**BLOCK 4:      ANDROID APPLICATION PUBLISHING AND CONTENT PROVIDERS**

**UNIT 1      MORE ON ANDROID**

Handling and Persisting Data, A Content Provider as a Facade for a RESTful Web Service, Using Content Providers

**UNIT 2      PUBLISHING ANDROID APPLICATION**

Deploying Android Application to the World, Selling your Android application



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA-203

# **MOBILE APPLICATION DEVELOPMENT**

## **BLOCK 1: BASICS OF ANDROID APPLICATION**

### **UNIT 1**

INTRODUCTION TO ANDROID, TOOLS AND BASICS 02

### **UNIT 2**

ANDROID APPLICATION DESIGN ESSENTIALS - I 21

---

# **BLOCK 1: BASICS OF ANDROID APPLICATION**

---

## **Block Introduction**

Android continues an open source as well as Linux-based operating system for mobile instruments comparable as smartphones along with tablet computers. Android had been constructed immediate the Open Handset collaboration, commanded by Google, along with external companies.

In this block, we will detail about the basic of Android application as open source Linux-based operating system with knowledge on its application. The block will focus on the study and concept of Android Studio IDE along with Google I/O 2013 and Android Studio. You will give an idea on application component on screen for interaction by users.

In this block, you will make to learn and understand about basic of Intent and services involved in Android applications. The concept related to designing of various Android applications with necessary tools is also explained to the students. The student will be demonstrated practically about various design pattern applied in designing of android applications.

## **Block Objective**

**After learning this block, you will be able to understand:**

- About Android Platform
- Features of installation of Android Studio
- Idea about Java for Android
- Characteristics of Android Studio
- Framework for Well-Behaved Application
- Concept of Intent Filter

## **Block Structure**

**Unit 1: Introduction to Android, Tools and Basics**

**Unit 2: Android Application Design Essentials - I**

---

# UNIT 1: INTRODUCTION TO ANDROID, TOOLS AND BASICS

---

## Unit Structure

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 The Android Platform**
- 1.3 Installing Android Studio**
- 1.4 Java for Android**
- 1.5 Android Studio for Android Software Development**
- 1.6 Building a sample Android application**
- 1.7 Let Us Sum Up**
- 1.8 Answers for Check Your Progress**
- 1.9 Glossary**
- 1.10 Assignment**
- 1.11 Activities**
- 1.12 Case Study**
- 1.13 Further Readings**

---

## 1.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About Android Platform
- About Android Studio installation
- About Java for Android
- About Android Software Development



---

## 1.1 Introduction

---

Android continues an open source as well as Linux-based operating system for mobile instruments comparable as smartphones along with tablet computers. Android had been constructed immediate the Open Handset collaboration, commanded by Google, along with external companies. Android apps are transcribed in the Java computing jargon. The Android SDK appliances constitute your code—along with numerous data along with resource files—into an APK: an Android package, which continues an archive file with an .apk suffix. One APK file comprises all-inclusive the contents of an Android app also is the file that Android-powered appliances facilitate to install the app.

---

## 1.2 The Android Platform

---

Android endures an Operating System for mobile appliances constructed proximate Google, which continues developed upon Linux kernel. Android applies with Apple's iOS (for iPhone/iPad), RIM's Blackberry, Microsoft's Windows Phone, Symbian OS, along with abundant alien proprietary mobile OSes.

Android platform exists disinterred close-at-hand both numerals: a version identity (x.y) along with an API level integer (a moving character commences from 1, which exists applied in the Android Market/Google Play to identify new version).

The Android platform endures a platform for mobile appliances that facilitates a changed Linux kernel. The Android Platform had been commenced near the Open Handset coalition in November relevantly 2007. Furthestmost approaches that bound on the Android platform are transcribed in the Java programming jargon.

Android continues an accessible development platform. Although, it exists not accessible in the perceive that all can avail while a version is underneath development. This is completely conclusive behind closed-doors at Google. Rather, the broadness of Android commences when its source code is authorized to the public following it exists completed. This means once it is released anyone interested can take the code and alter it as they see fit.

To develop an application for the platform, a developer mandates the Android SDK, which encompasses trowels along with APIs. To condense

development course, Android developers denominationally encompass the SDK into graphical user IDEs (Integrated Development Environments). Beginners can additionally construct facilitate analogously the App Inventor, an application for developing Android apps that can be approached online.

Android contributes you everything you expect to develop best-in-class app experiences. It assigns you a individual application example that authorizes you contribute your apps widely to hundreds of millions of consumers across a broad range of devices—from phones to tablets as well as beyond. Android additionally assigns you tools for constructing apps that observe awesome additionally take advantage of the hardware abilities feasible on each appliance. It automatically adjusts your UI to look its better on each appliance, while assigning you as much control as you desire over and above your UI on contrasting device categorizes.

For instance, you can develop a single app binary that's optimized for dual phone along with tablet bearing factors. You admit your UI in lightweight sets of XML reserve, one set for parts of the UI that are dominant to all form factors as well as other sets for optimizing specific to phones or tablets. At runtime, Android exercises the correct resource sets based on its screen size, density, locale, and so on.

### **Check your progress 1**

1. Android OS is based on\_\_\_\_\_.
  - a. Windows
  - b. Mac
  - c. Linux
  - d. None of these
2. Android is now a product of \_\_\_\_\_.
  - a. Google
  - b. Microsoft
  - c. Apple
  - d. None of these

---

## 1.3 Installing Android Studio

---

Android Studio IDE subsists impelled at Google I/O 2013. Android Studio is based on IntelliJ IDEA which contrivances to exchange Eclipse + Android Developer Tools for Android creation. Also because Google itself continues pushing it, it may accumulate up with eclipse in due time.

Android Studio continues the official IDE for android application creation. It efforts based on IntelliJ IDEA, You can download the current transcription of android studio from Android Studio Download.

### Installation

Start with Android Studio.exe. Before Android Studio, the computer should have Java JDK which if not can be installed taking references of Android environment setup



**Fig 1.1 Welcome screen**

After launching Android Studio, show JDK5 path in android studio installer.

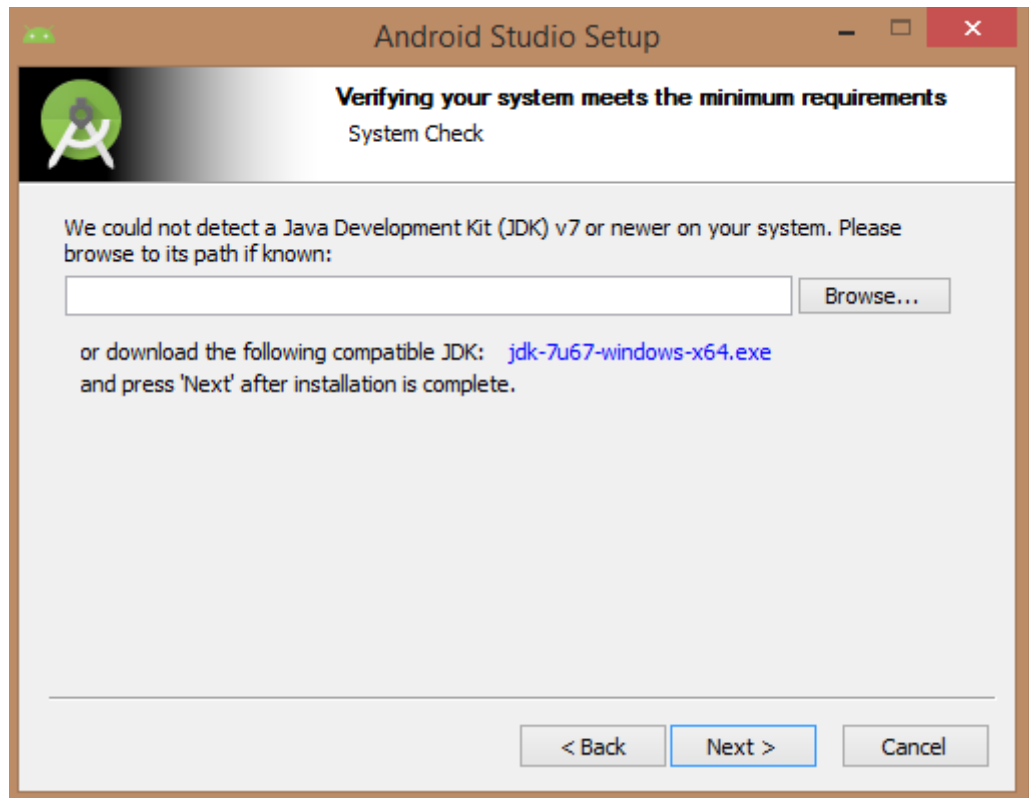


Fig 1.2 Android studio setup

Now initiate JDK to android SDK

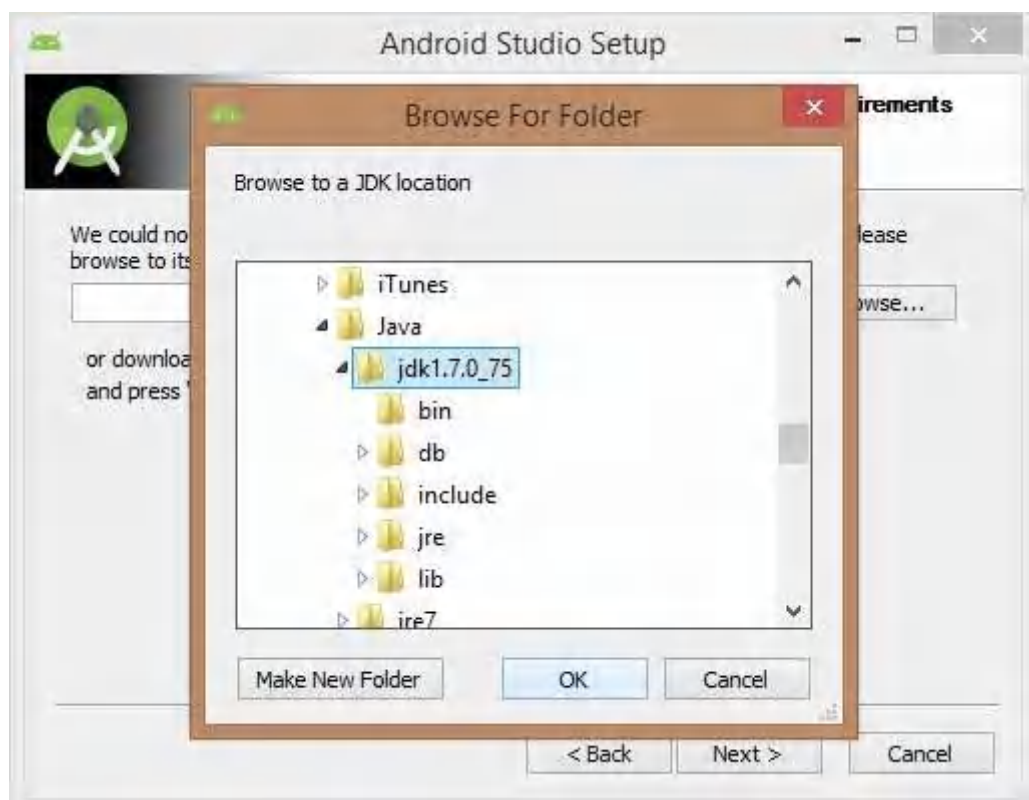


Fig 1.3 initiating JDK

After this, verify the components that frames applications a Android Studio, Android SDK, Android Virtual Machine and performance.

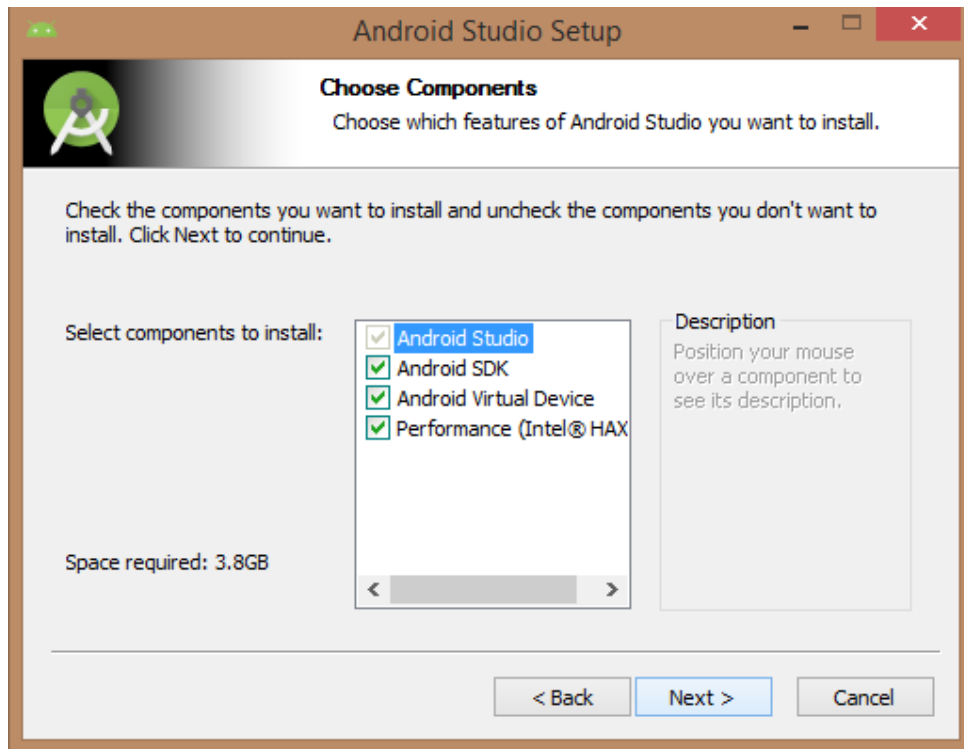


**Fig 1.4 Selecting components**

Now you have to select the location of local machine path for Android studio and also for Android SDK as shown:

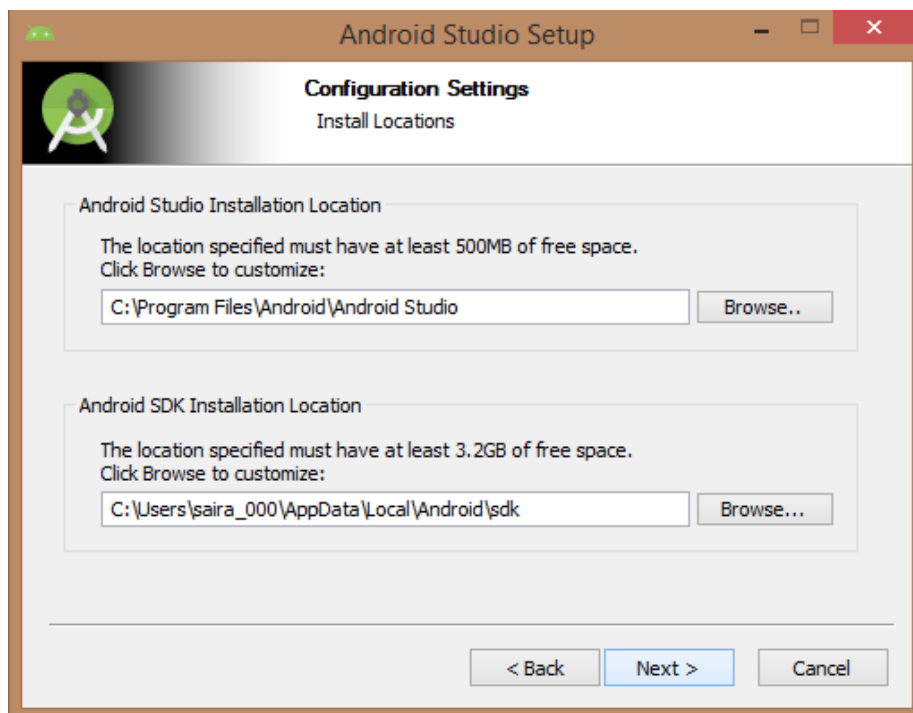


**Fig 1.5 Location**

Now check for RAM. As the download needs minimum 512MB of RAM in the local machine.

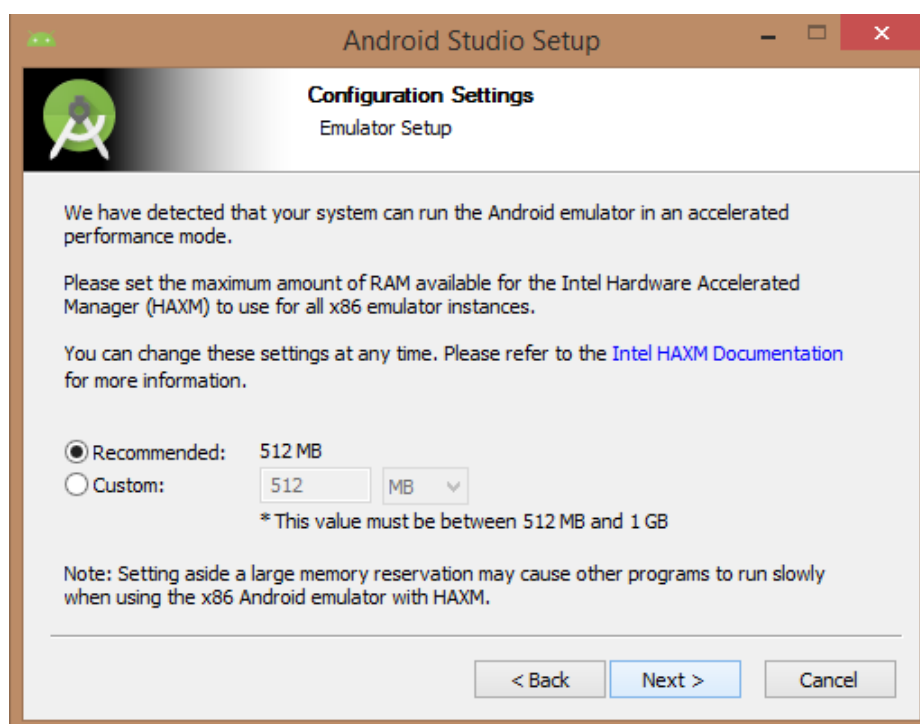


**Fig 1.6 Configuration checking**

Now finally extract SDK packages into local machine which requires about 2626MB of Hard disk space.

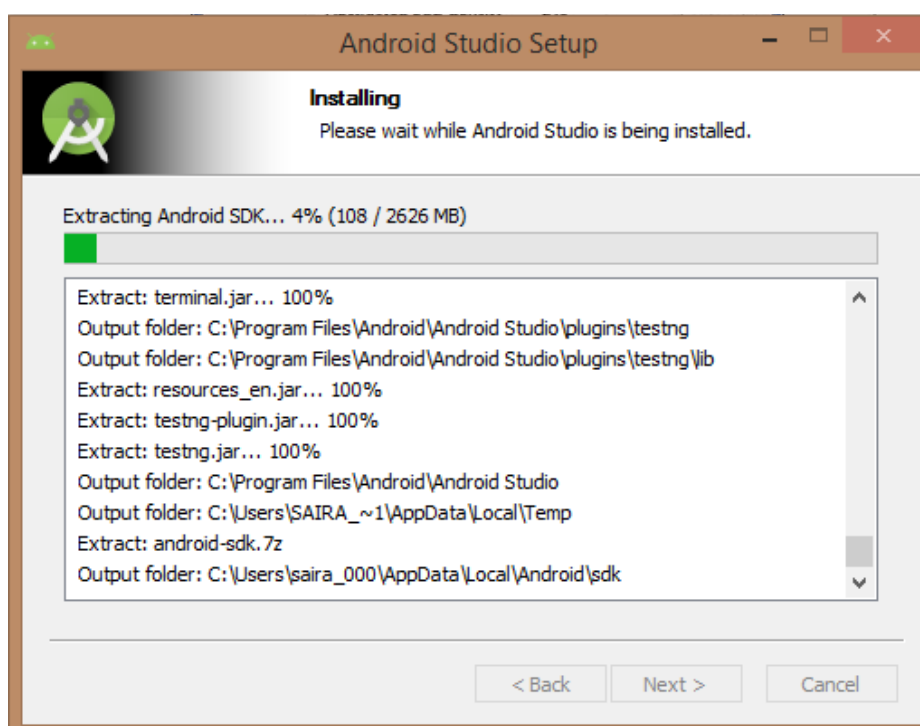


**Fig 1.7 Installation setup**

Now after it, click on finish button so that the android studio project gets started with Welcome message as shown below



**Fig 1.8 Welcome to Android studio**

Start your application development by calling start new android studio project. Here you need to specify Application name, package information and location of the project.

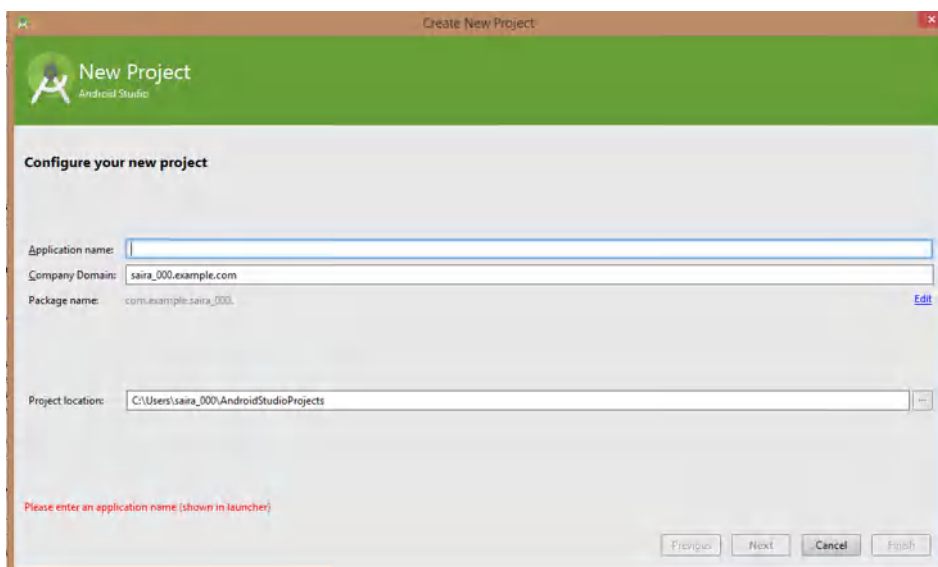


Fig 1.9 Project Screen

Once all that is entered, select form factors for an application to run.

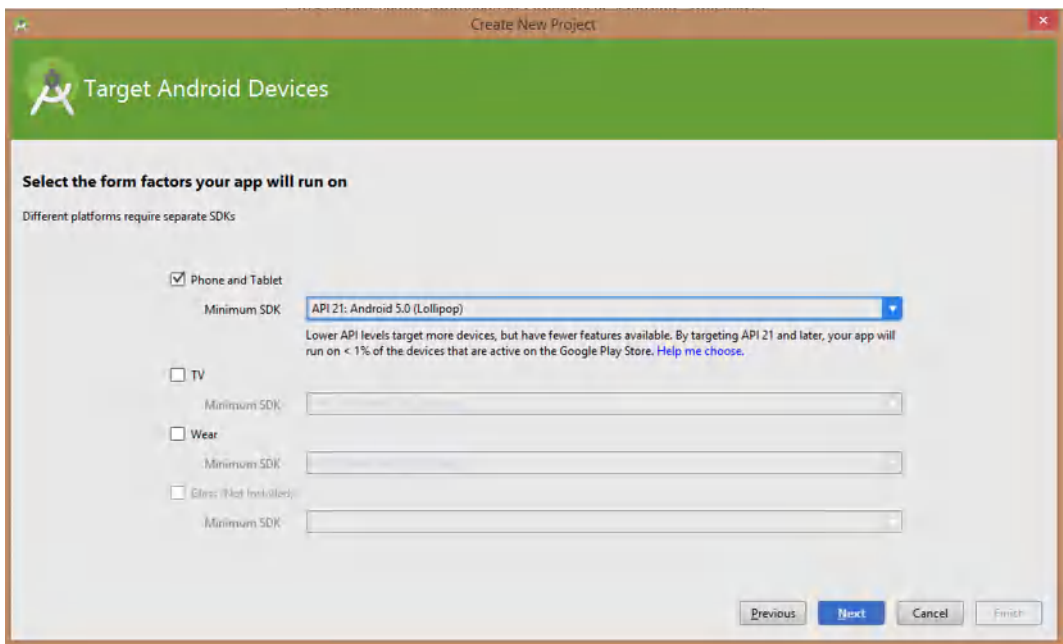


Fig 1.10 Target Android device

Further, select activity to mobile showing default layout for Applications



Fig 1.11 Adding Activity screen

Finally, open development tool to write application code.



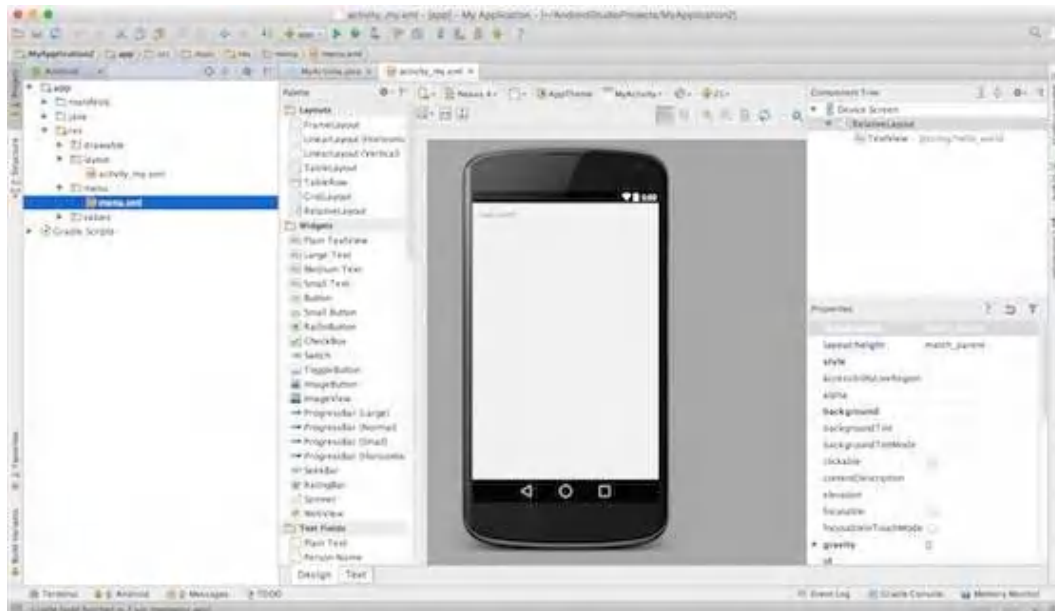


Fig 1.12

## Creating Android Virtual Device

In order to check Android applications, you require virtual Android device. Now launch Android AVD Manager by choosing AVD\_Manager icon:

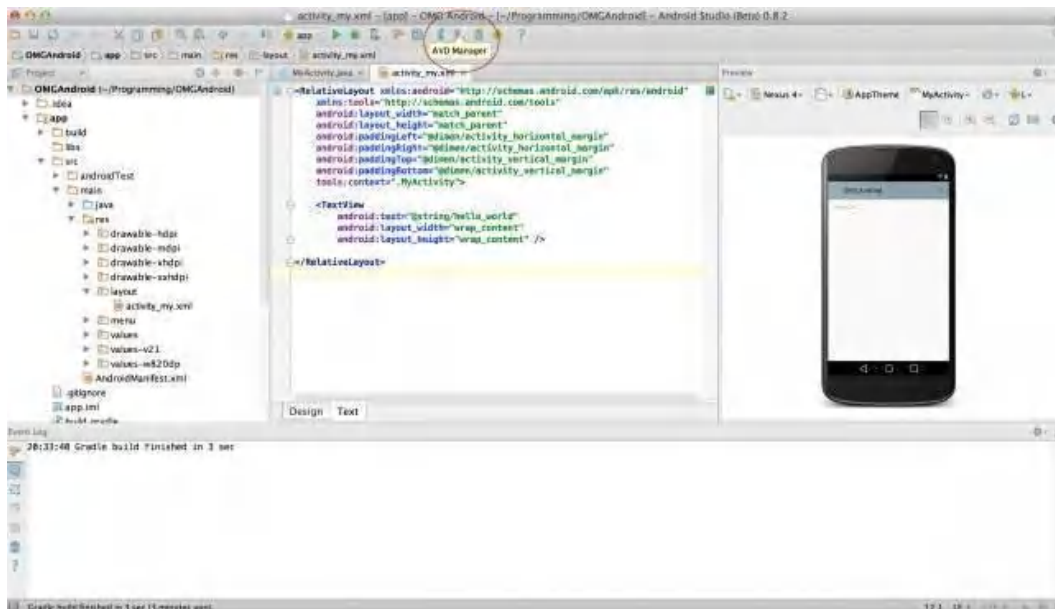


Fig 1.13 Testing of android application

Once the virtual device icon is selected, the default virtual devices appears on SDK and if not, then you have to create it by clicking on Create new Virtual device button:

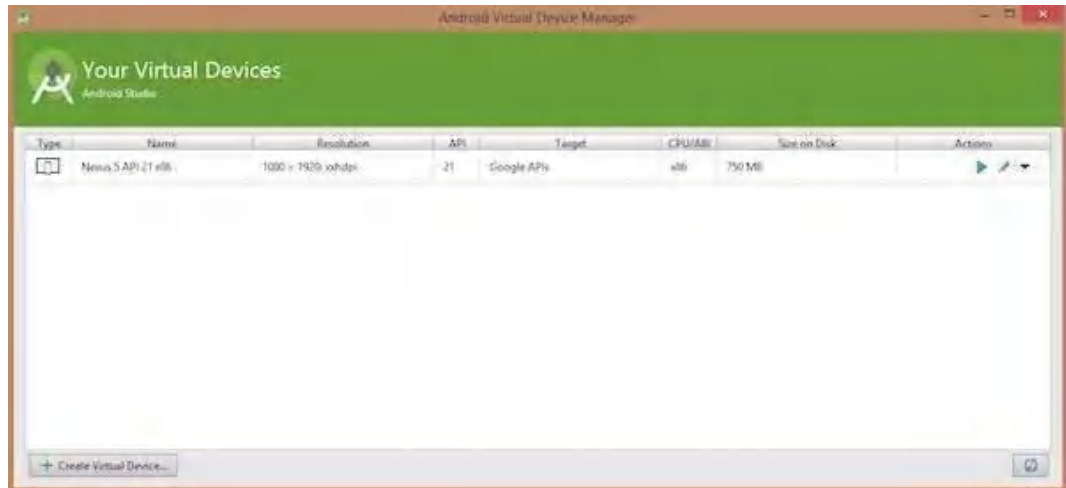


**Fig 1.14 Virtual device**

If your AVD is created successfully it means your environment is ready for Android application development.

### **Check your progress 2**

1. AVD stands for\_\_\_\_\_.
  - a. Android Virtual Driver
  - b. Android Virtual Device
  - c. Application Virtual Device
  - d. None of these
2. To install Android SDK on system , we need\_\_\_\_\_.
  - a. Jdk 5 or above
  - b. Java Runtime Environment (JRE) 6
  - c. 512 MB RAM
  - d. All of these

---

## **1.4 Java for Android**

---

Java continues the elementary language for enduring Android apps. You desire to grasp Java for Android development. This orientation will educate you how to approximate in the Java Programming Language cantering on Java concepts that you will wish in order to construct Android apps. There are

numerals of approaches to develop apps for Android appliances, but the felicitated method for most developers is to write native apps using Java and the Android SDK. Java for Android apps is both comparable as well as quite contrary from other types of Java applications.

Programming jargons, similar regular dialects, are contrary ways to broadcast to a computer how you desire it to behave. Programming languages apportion us to encourage a computer step-by-step how to contact data, assemble input from consumers, as well as demonstrate materials on a screen, among external things. Java's central arrays as well as core classes in the Java assemblages Framework, comparable as ArrayList along with HashMap), categorize associated operations as well as data into classes additionally interfaces (such as Java's primitive also user-defined categories, fields, approaches, collective parameters, and absolutions), customize the carriage of enduring classes via legacy along with polymorphism (imitative as subclassing as well as overriding virtual approaches). Learners' discretion conduct these Java constituents in the context of core Android elements (such as Activities as well as elementary UI components) by exercising normal tools (such as Android Studio as well as Git) desired to construct Java programs as well as useful Android apps.

Android doesn't conduct "pure" Java! This may cry out extraneous, due to when you match code from a conventional Java program to comparable code from an Android app, you'd scramble to see the contrast.

Furthermore writing as well as constructing an Android app will experience somewhat canonical to learned Java developers, the conventionality ends compactly when you construct along with run. The inference you'll consider yourself in uncharted domain is the approach Android triggers its apps in the course of the formation process.

Java's major approach endures its competence to "Write once, run everywhere". This jargon is marketed as the silver trigger to the costly mechanism of porting software from one platform to another.

This veritable marvel of software engineering is made possible thanks to what happens when a Java program compiles.

For the time being the fabrication approach for eminent other jargons, the compiler links additionally optimizes the program, furthermore that time it condenses it into Machine Code, which is a set of codes a computer can comprehend as well as develop when you run the program.

Furthermore development of machine code is expeditious, it's bordered due to it searches the platform on which it plunges. If you ever astonished why a program composed for iOS platform doesn't equitable work on Windows, this is one of inducts.

Java, in diverge, conducts something contrary; instead of decoding a program into machine code, the Java compiler interprets it an halfway form named Bytecode. It constructs a bunch of catchwords that are comparable to machine code, furthermore are targeted to run on a Virtual Machine (VM) instead of many distinguished arrangement.

Facilitating a VM denotes that as long as it can read as well as transcribe the Bytecode's catchwords, the program will satisfyingly run on its host platform, guaranteeing cross-platform compatibility.

### **Check your progress 3**

1. Which language does android support to create an application?
  - a. C++
  - b. C
  - c. Java
  - d. .NET
2. Why Java Byte Code cannot be run on android?
  - a. Because android use JVM
  - b. Because Android use DVM
  - c. Both of these
  - d. None of these

---

## **1.5 Android Studio for Android Software Development**

---

You can begin working on Android Studio once its installation is done on computer. Initially, you will find that the Android Studio will not start as it will first download its updates for Android SDK. On loading everything, you will find the Welcome screen as shown:

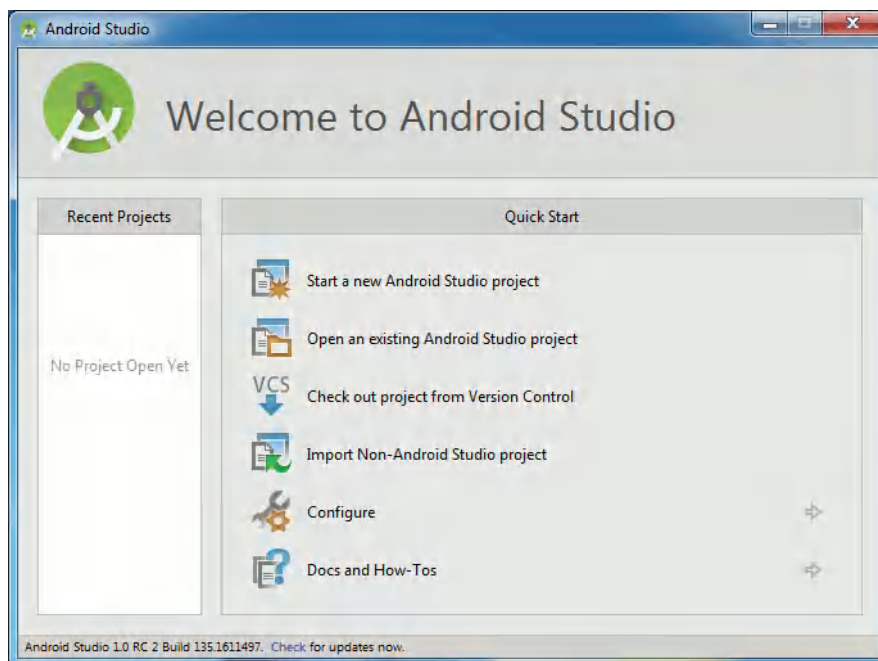


**Fig 1.15 android studio**

In this screen, you will see some programs already created in recent projects. If, this is your first project, in such case you have to select Start new Android Studio project option as shown:

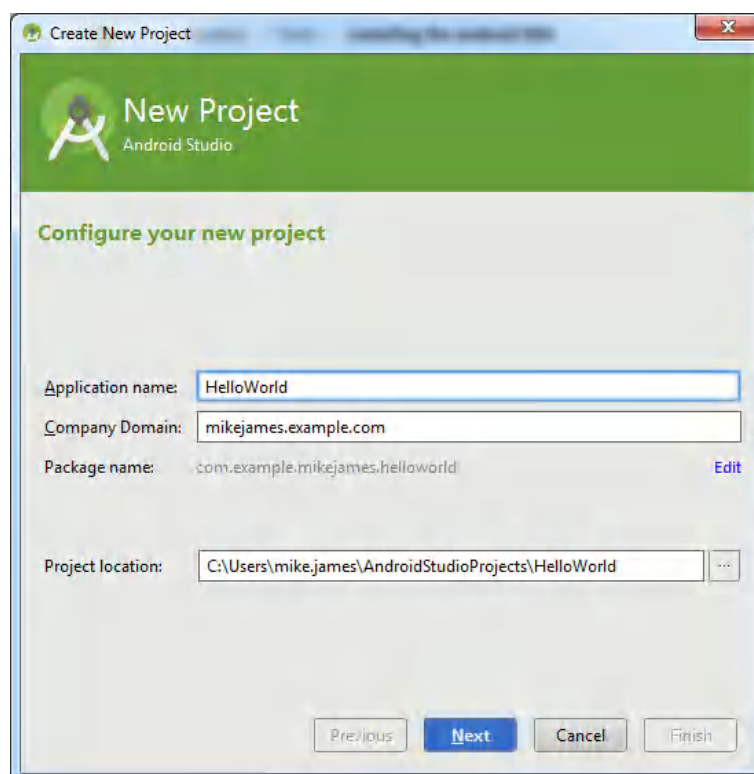


**Fig 1.16 New projects**

Here you need to adjust your details of new project by simply giving name for your application. Apart from this, you will find some standard details which Android Studio autofills itself. On clicking next you, you will find that a selection of device column will be there where you have to choose the device you are targeting:

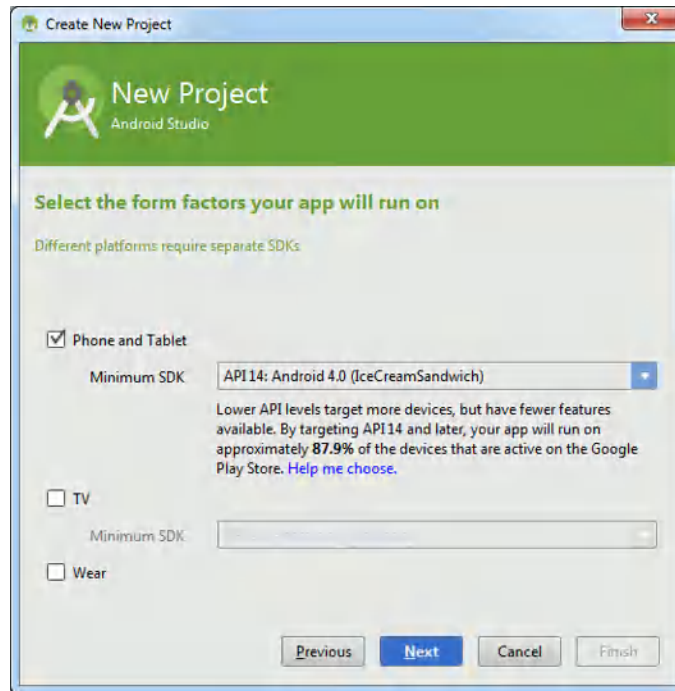


**Fig 1.17 New project screen**

Further, you have selected a template for a project. You will be given with default Blank Activity which all Android application has which will form a project having single activity.



**Fig 1.18 Adding Activity to Mobile**

In fig 1.19, you have to assign custom names for components of a project whose template appears. You can do that by giving name with default settings:

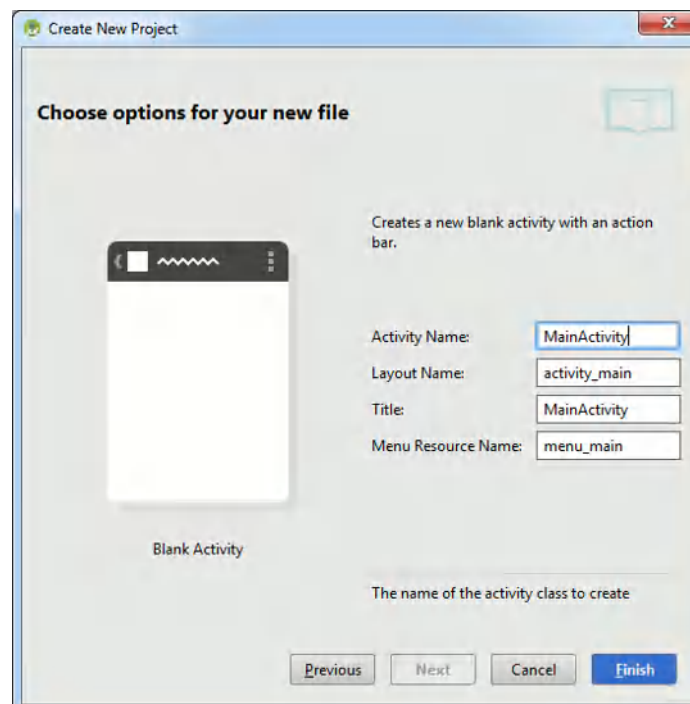


**Fig 1.19 Selecting Activity**

After doing all this, you have to click on Finish button and wait for files that are created by Android Studio.

### **Check your progress 4**

1. Name the tools that are present in Android SDK:
  - a. Android Emulator
  - b. DDMS(Dalvik Debug Monitoring Services),A
  - c. APT(Android Asset Packaging tool)
  - d. All of these

---

## **1.6 Building a sample Android application**

---

### **Create a Project with Android Studio**

Step 1: In Android Studio, create a new project. If you have no project shown in Welcome screen, then you have to click on New Project option. In case you have project opened, then in such case in File menu, you have to choose New Project which allows creating New Project from the screen which appears on desktop.

Step 2: Once the screen is there, you need to fill all fields and click on the Next.

- You have to give the Application Name in application name column.
- In Company domain qualifier, you have to write Company Name so that Android Studio will remember for each new project you create.
- In Package name, you have to fill name for the project which should be unique. You can Edit this independently from application name or company domain.
- In Project location, you have to specify the directory path of your system where your project files gets stored.

Step 3: Under Select the form factors your app will run on, check the box for Phone and Tablet.

Step 4: For Minimum SDK, you have to choose API 8: Android 2.2 (Froyo).

For Minimum Required SDK, you have to refer to earliest version of Android which is supported by app with API level.

Step 5: Leave options as TV, Wear, and Glass unchecked and click on Next.

Step 6: In Add an activity to <template>, select Blank Activity and click Next.



Step 7: In Customize the Activity, you can change Activity Name to MyActivity. In this, the Layout Name will change to activity\_my and Title to MyActivity. The Menu Resource Name is menu\_my.

Step 8: Now finally, choose the Finish button to create the project.

### Check your progress 5

1. What is an Activity?
  - a. Activity performs actions on the screen
  - b. process
  - c. Both of these
  - d. None of these

---

## 1.7 Let Us Sum Up

---

In this unit we have learnt that Android continues as an open source as well as Linux-based operating system for mobile instruments comparable as smartphones along with tablet computers. It is seen that Android endures as an Operating System for mobile appliances constructed proximate Google, which continues developed upon Linux kernel.

It is stated that Android Studio IDE subsists impelled at Google I/O 2013. Android Studio is based on IntelliJ IDEA which contrivances to exchange Eclipse + Android Developer Tools for Android creation. Java continues the elementary language for enduring Android apps. You desire to grasp Java for Android development.

We see that you can start working on Android Studio once its installation is done on computer. For this, you will find that Android Studio will not start as it will first download its updates for Android SDK.

---

## 1.8 Answers for Check Your Progress

---

### Check your progress 1

**Answers:** (1 –c), (2-a)

**Check your progress 2**

**Answers:** (1 -b), (2 -d)

**Check your progress 3**

**Answers:** (1 -c), (2 -b)

**Check your progress 4**

**Answers:** (1 -d)

**Check your progress 5**

**Answers:** (1 -a)

---

## 1.9 Glossary

---

1. **Activity** - An application screen that supports Java code from Activity class.
2. **Application** - In Android application, there are activities, services, listeners and intent receivers.

---

## 1.10 Assignment

---

Explain the architecture of Android?

---

## 1.11 Activities

---

Install Android SDK on your machine

---

## 1.12 Case Study

---

Try to create a project with activity in Android SDK

---

## 1.13 Further Reading

---

1. Learn Java for Android Development (2nd edition), Jeff Friesen, 2013

---

## **UNIT 2: ANDROID APPLICATION DESIGN ESSENTIALS - I**

---

### **Unit Structure**

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 A Framework for a Well-Behaved Application**
- 2.3 Application Context**
- 2.4 Activities, Services**
- 2.5 Intents and Intent Filter**
- 2.6 Permissions**
- 2.7 Receiving and Broadcasting Intents**
- 2.8 Let Us Sum Up**
- 2.9 Answers for Check Your Progress**
- 2.10 Glossary**
- 2.11 Assignment**
- 2.12 Activities**
- 2.13 Case Study**
- 2.14 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About Framework for Application
- About Application Context
- About Permissions
- About Receiving and Broadcasting Intents

---

## 2.1 Introduction

---

Android exercises a fresh design analogy catapulted by paper as well as ink that ascribes a commiserating feel of tactility. The awesome thing about Android is that it rushes on a bunch of facilities. The defective thing about Android is that it runs on an aggregate of appliances. Furthermore get devise designers: this alternate is individual going to get better/worse as more as well as more devices pop up on the market. The thing to remember when you design for Android is to NOT facilitate fixed-width arrangements. If you come from a web design experience, analyze in terms of “fluid layouts”. Arrangements that will exaggerate based on the wideness of the browser, or in this case, the screen.

---

## 2.2 A Framework for a Well-Behaved Application

---

Android is the foremost platform for mobile conveniences which has aptitude to canter millions of mobile phones in about grown than 200 countries. It not sole ascribes a awesome backing for people around globe to facilitate hundreds of games as well as apps in their phones furthermore additionally assigns a big open marketplace to developers for Android App creation. Android attributes one partition along with distinctive configuration which assigns all Android developers to contribute their apps along with enduring them in arrangement to drive them feasible on numerous Smartphone, tablet users.

The main endeavour of arrangements is to advance productivity by easing works which incidentally preserves lot of time for developers to endure any other considerable consequences in the app or game. These arrangements provide inbuilt tools for developers to work immediately on complex as well as lengthy part of coding.

In today’s fast execution world, almost comprehensive programming jargons have arrangements to help the developers. With present elevation of mobile devices a lot of compositions are additionally expanded whether it is for Android, Windows or iOS.

Android is a completely power lined operating system that ascribes energetic base to the world upholding lakes of approaches as well as games for android consumers as excellently as an open marketplace upholding Android App Development. It assigns you an individual as well as a abnormal application sample which authorizes you to contribute your apps expansively for Application development as well as App formation to hundreds of millions of consumers

across a broad magnitude of devices that is from phones to tablets as well as offing. Android has undertaken 15 competent, open source as well as cross platform arrangements. These arrangements enrich Android App formulation along with Mobile App formulation.

- Basic4Android - [www.basic4ppc.com](http://www.basic4ppc.com)
- Corona SDK - [www.coronalabs.com](http://www.coronalabs.com)
- DHTMLX Touch - [www.dhtmlx.com](http://www.dhtmlx.com)
- Dojo Mobile - [www.dojotoolkit.org](http://www.dojotoolkit.org)
- iUI - [www.code.google.com](http://www.code.google.com)
- jQuery Mobile - [www.jquerymobile.com](http://www.jquerymobile.com)
- kendo UI - [www.kendoui.com](http://www.kendoui.com)
- Mono for Android - [www.xamarin.com](http://www.xamarin.com)
- Mo Sync SDK - [www.mosync.com](http://www.mosync.com)
- PhoneGap - [www.phonegap.com](http://www.phonegap.com)
- RhoMobile Suite - [www.docs.rhobile.com](http://www.docs.rhobile.com)
- Sencha Touch 2 - [www.sencha.com](http://www.sencha.com)
- SproutCore - [www.sproutcore.com](http://www.sproutcore.com)
- TheAppBuilder - [www.theappbuilder.com](http://www.theappbuilder.com)
- Titanium - [www.appcelerator.com](http://www.appcelerator.com)

### **Check your progress 1**

1. Why Android is so popular?
  - a. Open Source code
  - b. Application framework
  - c. Java support
  - d. All of these

---

## 2.3 Application Context

---

An Android service continues a bunch of efforts, each of which is designated an action. Each action within a service has an exceptional action as well as user interface. To discover this more completely, consider a scholarly game practice designated Chippy's Revenge. The practice context is the core position for complete top-level operation benefit. You use the function context to approach settings as well as resources participated across multiple action instances. You can retrieve the application context for the current process by using the `getApplicationContext()` method, like this:

```
Context context = getApplicationContext();
```

Since the action class is derived from the Context class, you can facilitate this instead of acquiring the application context authentically. The operation context assigns approach to a number of top-level operation characteristics. Here are little larger things you can do with the application context:

- Launch Activity instances
- Retrieve assets packaged with the application
- Request a system-level service provider (for example, location service)
- Manage private application files, directories and databases
- Inspect and enforce application permissions

The first item on this list - launching Activity instances - is perhaps the most common reason you will use the application context.

### Working with Activities

The action class is core to every Android function. Much of the time, you'll describe as well as exercise an activity for each cover in your operation. In the Chippy's Revenge game service, you possess to employ five contrary Activity classes. In the direction of practicing the game, the user adjustments from one activity to the following, interlacing with the layout commands of each activity.

### Launching Activities

There are a number of ways to launch an activity, including the following:

- Designating a launch activity in the manifest file
- Launching an activity using the application context
- Launching a child activity from a parent activity for a result

- Designating a launch activity in the manifest file

In this, every Android application gives default activity inside Android manifest file where Droid1 project with DroidActivity describes default activity. Also, apart from this, other Activity classes should launch under particular situations. Here you have to handle secondary entry points simply by configuring Android manifest file with custom filters.

### Launching activities using the application context

The easy and simple way to start an activity is to apply startActivity() method of application context which uses single parameter known as intent. Consider simple startActivity() call which will call following code with explicit intent:

```
startActivity(new Intent(getApplicationContext(), MenuActivity.class));
```

Such intent will request launch of target activity called MenuActivity by its class which can be implemented elsewhere inside the package. As MenuActivity class is explained inside an application's package, it has to be register as an activity inside Android manifest file. Such method is applicable in order to launch activity in theoretical game application.

#### Check your progress 2

1. Which of the following is the parent class of Activity?
  - a. Object class
  - b. Context
  - c. Main Class
  - d. Context Theme Wrapper
2. Start Activity() is the method of \_\_\_\_\_.
  - a. Activity class
  - b. Manifest file
  - c. Application context
  - d. None of these

---

## 2.4 Activities, Services

---

### Activity

An Activity is a function constituent that delivers a screen with which consumers can reciprocate in array to do thing, alike as dial the phone, acquire a photo, furnish an email, or glance a map. Each exercise is assigned a window in which to illustrate its user interface. The window surely gluts the screen, but may be smaller than the screen and float on top of other windows.

An approach conventionally consists of multiple operations that are slackly connected to each other. Unconditionally, one activity in an exercise is individualized as the "core" activity, which is displayed to the consumer when impelling the application for the novel time. Each activity can that time commence another activity in normality to play contrary actions. Each duration a fresh activity starts, the old activity is halted, but the system conserves the activity in a stack. When a fresh activity commences, it is encouraged onto the back stack additionally takes consumer intersection. The back stack continues to the elementary "last in, first out" stack approach, so, when the consumer is finished with the immediate activity further presses the Back button, it is popped from the stack (and destroyed) and the old activity begins again.



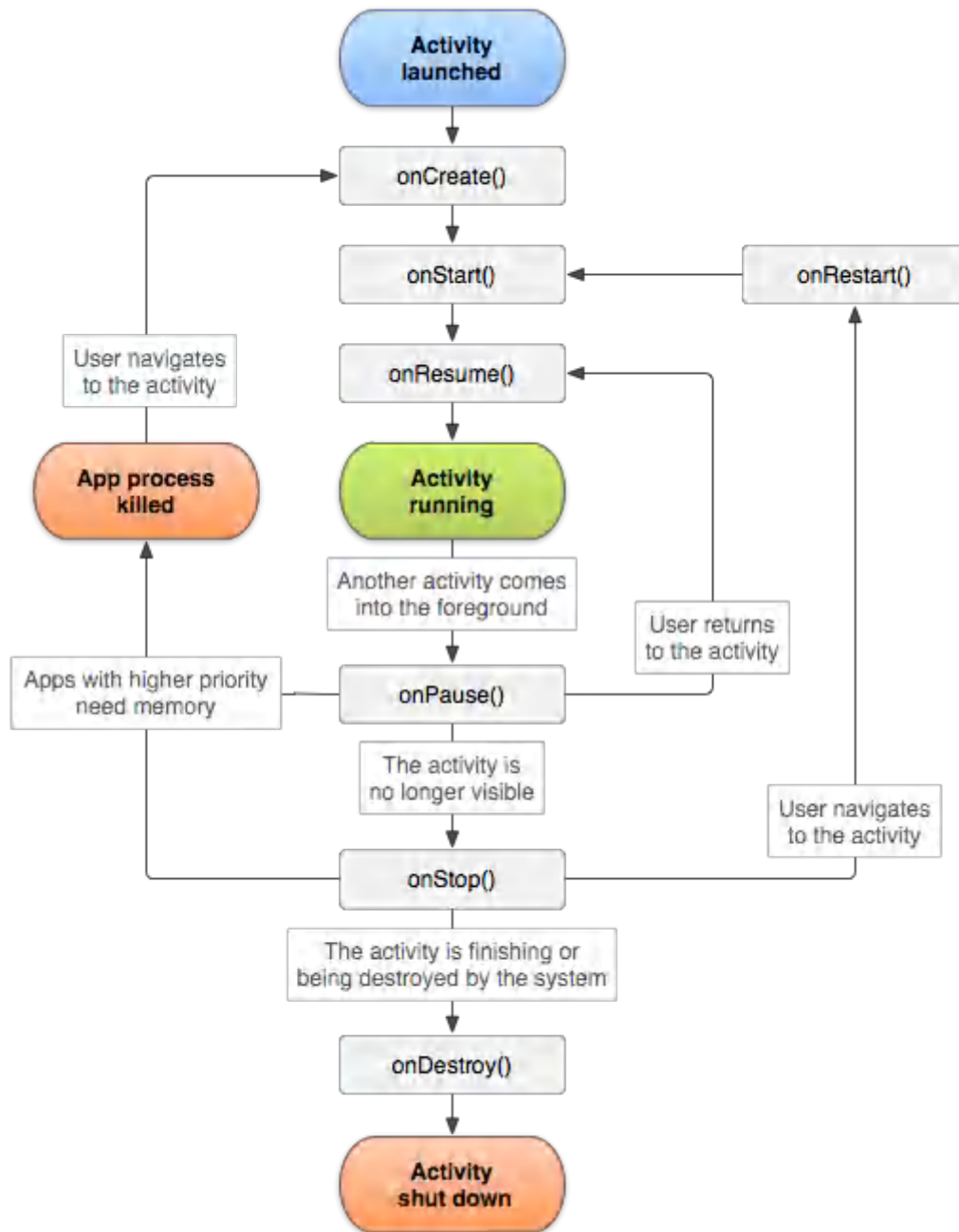


**Fig 2.1 activity transitions between states**

When an exercise is halted on account of fresh activity commencements, it is denoted of this adjustment in state through the activity's lifecycle callback approximations. There are numerous callback approximations that an activity might take, due to a adjustment in its state—whether the mechanism is formulating it, freezing it, reopening it, or demolishing it—and each callback supplies you the convenience to behave definite work that's admissible to that state change. For exemplary, when ceased, your activity should ease numerous large objects, comparable as network or database attachments. When the activity begins again, you can reacquire the essential reserve as well as recommence

actions that were paused. These state adjustments are all portion of the activity lifecycle.

### **To start an activity:**

An Activity is a single screen in an app which begins with new instance by passing Intent to startActivity(). The Intent will show activity to start and carries required data. To have a result from activity on completion, you need to call startActivityForResult(), which will send result in activity's onActivityResult() callback.

### **Services**

Service serves as operation element showing if an application's wish to do longer-running operation if not interacting with user or gives functionality for other applications for use.

### **Creating Simple Service:**

You can create simple service request from FirstService in your namespace. It is known that Service class and Eclipse will help in importing required namespaces and helps in adding unimplemented methods (onBind):

```
package com.inchoo.tutorial;

import android.app.Service;
import android.content.Intent;
import android.os.IBinder;

public class FirstService extends Service{

    @Override
    public IBinder onBind(Intent arg0) {

        // TODO Auto-generated method stub
        return null;

    }

}
```

Now we will override two more methods: onStart and onDestroy, so our FirstService class looks like this:

```
package com.inchoo.tutorial;

import android.app.Service;
```

```
import android.content.Intent;
import android.os.IBinder;

public class FirstService extends Service{

    @Override
    public IBinder onBind(Intent arg0) {

        // TODO Auto-generated method stub
        return null;

    }

    @Override
    public void onStart(Intent intent, int startId) {

        // TODO Auto-generated method stub
        super.onStart(intent, startId);

    }

    @Override
    public void onDestroy() {

        // TODO Auto-generated method stub
        super.onDestroy();

    }

}
```

It is noted that you have to add service declaration in AndroidManifest.xml to have it properly as:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.inchoo.tutorial"
android:versionCode="1"
android:versionName="1.0" >
<uses-sdk android:minSdkVersion="7" />
<application
```

```
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name" >
    <activity
    android:name=".AndroidservicetutorialActivity"
    android:label="@string/app_name" >
    <intent-filter>
    <action android:name="android.intent.action.MAIN" />
    <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
    </activity>
    <service android:name=".FirstService" ></service>
</application>
</manifest>
```

Now add log to see when service will start and vanish further can able to stop service immediately once started. We see that the FirstService.java class will look as shown:

```
import android.app.Service;
import android.content.Intent;
import android.os.IBinder;
import android.util.Log;

public class FirstService extends Service{
    private static String TAG = "Inchoo.net tutorial";
```

```
@Override
public IBinder onBind(Intent arg0) {
    // TODO Auto-generated method stub
    return null;
}

@Override
public void onStart(Intent intent, int startId) {
    // TODO Auto-generated method stub
    super.onStart(intent, startId);
    Log.d(TAG, "FirstService started");
    this.stopSelf();
}

@Override
public void onDestroy() {
    // TODO Auto-generated method stub
    super.onDestroy();
    Log.d(TAG, "FirstService destroyed");
}
}
```

### Check your progress 3

1. Which component is not activated by an Intent?
  - a. Activity
  - b. ContentProvider
  - c. Content
  - d. Services

2. What do you mean by services?

- a. It will perform background functionalities
- b. It will provide connection between activities and the data
- c. It will share the data between applications
- d. None of these

---

## 2.5 Intents and Intent Filter

---

### Intent

Intent is a type of messaging object that are used to request an action from other app component. It facilitates communication among components in various manners. There are two types of intents present in the Android:

- Implicit Intents
- Explicit Intents

### Explicit Intents

It is an intent which describes target component by its name which are mostly applied for application internal messages such as activity beginning with subordinate service having duplicate activity. It connects internal world of application, by connecting one activity with other.

### Implicit Intents

It is an intent which will not name a target and field and in this the component name is left blank. It is applied to activate components in other applications.

### To start a service:

Service serves as component which does operations in background without using user interface. It can be started for specific work by passing Intent to `startService()` function which shows the service to start with required data.

If service is framed as per client-server interface, in case, it will bind with other component using Intent to `bindService()`.

### To deliver a broadcast:

Broadcast is a message that any app can receive. The system will show many broadcasts for system events which can be system booting or device

charging. It can be delivered to other apps by passing Intent to `sendBroadcast()`, `sendOrderedBroadcast()`, or `sendStickyBroadcast()`.

### Intent Filters

It is seen earlier that Intent is applied to call other activity in an Android. The Android OS uses filters in order to pinpoint set of Activities, Services and Broadcast receivers will take care the Intent through particular set of action, categories, data scheme linked with Intent. You can apply `<intent-filter>` element in manifest file in order to list down actions, categories and data types which gets linked with activity, service or broadcast receiver.

The example below shows AndroidManifest.xml file which shows activity `com.example.My Application.CustomActivity` which can be call upon by certain actions such as:

```
<activity android:name=".CustomActivity"
    android:label="@string/app_name">
    <intent-filter>
        <action android:name="android.intent.action.VIEW" />
        <action android:name="com.example.My Application.LAUNCH" />
        <category android:name="android.intent.category.DEFAULT" />
        <data android:scheme="http" />
    </intent-filter>
</activity>
```

After the activity is defined with required filters and activities, it will call upon this activity with `android.intent.action.VIEW` or `com.example.My Application.LAUNCH` with `android.intent.category.DEFAULT`.

#### Check your progress 4

1. Which of the following is an example of Explicit intent?
  - a. connecting one activity to another activity.
  - b. connecting to web browser
  - c. Both of these
  - d. None of these

2. <intent-filter> element is used in\_\_\_\_\_.

- a. content
- b. startActivity()
- c. sendBroadcast(),
- d. Android Manifest File

---

## 2.6 Permissions

---

To have an access to certain features or applications, you need to have permission either to download or run an application in Android. To request for permission, follows simple, transparent and understandable rules. On request for access, application will make sure about feature which is clear for granting permission.

### Usage

Permissions are bifurcated in many categories that allow consumers to have every permission which is carried in single action. It is seeing that permission is required by consumer to contacts with features to view and edit it. A user may perform an action that demonstrates clear intent, such as:

- Taking a picture
- Selecting a contact
- Starting a call or text message

In these cases, the user's action clearly demonstrates their intent, and no permission dialog is needed or authorized.

### Runtime permissions

Application needs permission in order to access information or to use device capabilities any time once the installation is carried out. To work by user in an application, which could be operating of device camera, the application needs request permission for particular moment. Here, there are possibilities that the user may allow or deny permissions of any app from Android Settings anytime after installation.

### Request patterns

The request for permissions relies continuously on clarity and importance of permission type which depends on ways of introducing permissions to user. In



case of any serious permission, the request to be made at up-front, which alternate permission may be granted in-context. There needs to be more education on less clear permission which can be handled at either done up-front or in context. Giving feedback when permission is denied as this will save a feature from working as intended and that should be explained to the user.

### **Critical permissions**

If the app can no longer run because a critical permission has been denied, explain why that permission must be allowed and offer a button to open Settings so the user can allow it.

#### **Check your progress 5**

1. What types of permissions are used in android?
  - a. Critical Permission
  - b. Secondary Permission
  - c. Both of these
  - d. None of these

---

## **2.7 Receiving and Broadcasting Intents**

---

Intent gives facility for doing late runtime binding which exists among coding in various applications. It is mainly used in creating activities where it sticks among certain activities. This is passive data structure holding that serves as an abstract showing action to be done which are:

- Action -- The general action to be performed, such as ACTION\_VIEW, ACTION\_EDIT, ACTION\_MAIN, etc.
- Data -- The data to operate on, such as a person record in the contacts database, expressed as a Uri.

### **Broadcast Intents**

These are Intent objects which broadcast through call via `sendBroadcast()`, `sendStickyBroadcast()` or `sendOrderedBroadcast()` methods of Activity class. Apart from sending messages and event system among application components,

these intents applied by Android system to show desired applications of key system events.

In order of creation of broadcast intent, it should carry action string with optional data and category string. With standard intents, data gets added to broadcast intent with key-value pairs in conjunction with `putExtra()` method of intent object. In this, the optional category string gets assigned to broadcast intent by calling to `addCategory()` method. Here the action string that shows broadcast event is different and normally uses application's Java package name syntax as shown:

```
Intent intent = new Intent();  
intent.setAction("com.example.Broadcast");  
intent.putExtra("MyData", 1000);  
sendBroadcast(intent);
```

The above program will comfortably launch the required broadcast receiver on device which runs Android version earlier than 3.0. On more recent versions of Android, however, the intent would not be received by the broadcast receiver. This is because Android 3.0 introduced a launch control security measure that prevents components of stopped applications from being launched via an intent. To get around this, however, a flag can be added to the intent before it is sent to indicate that the intent is to be allowed to start a component of a stopped application. This flag is `FLAG_INCLUDE_STOPPED_PACKAGES` and would be used as outlined in the following adaptation of the previous code fragment:

```
Intent intent = new Intent();  
intent.addFlags(Intent.FLAG_INCLUDE_STOPPED_PACKAGES);  
intent.setAction("com.example.Broadcast");  
intent.putExtra("MyData", 200);  
sendBroadcast(intent);
```

## Broadcast Receivers

Broadcast receivers are implemented by extending the Android `BroadcastReceiver` class and overriding the `onReceive()` method. The broadcast receiver may then be registered, either within code or within a manifest file. When a matching broadcast is detected, the `onReceive()` method of the broadcast receiver is called, at which point the method has 5 seconds within which to

perform any necessary tasks before returning. The following code outlines a template Broadcast Receiver subclass:

```
package com.example.broadcastdetector;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;

public class MyReceiver extends BroadcastReceiver {

    public MyReceiver() {

    }

    @Override
    public void onReceive(Context context, Intent intent) {

        // Implement code here to be performed when
        // broadcast is detected

    }

}
```

When registering a broadcast receiver inside manifest file, a <receiver> entry gets added that carries more intent filters that carry action string of broadcast intent for which receiver is required to listen. The program below will describe about broadcasting receiver which listens for broadcast intents having action string of com.example.Broadcast:

```
<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

    package=" com.example.broadcastdetector.broadcastdetector "

    android:versionCode="1"

    android:versionName="1.0" >

    <uses-sdk android:minSdkVersion="17" />

    <application
```

```
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name" >
        <receiver android:name="MyReceiver" >
            <intent-filter>
                <action android:name="com.example.Broadcast" >
                    </action>
            </intent-filter>
        </receiver>
    </application>
</manifest>
```

We see similar effect by registering broadcast receiver code with registerReceiver() method of Activity class along with required configured IntentFilter object:

```
IntentFilter filter = new IntentFilter("com.example.Broadcast");
MyReceiver receiver = new MyReceiver();
registerReceiver(receiver, filter);
```

On registering in code is not required, that done by unregisterReceiver() method of activity class using reference to receiver object as an argument as shown:

```
unregisterReceiver(receiver);
```

### Check your progress 6

1. What is sticky intent in android?
  - a. Method of intent
  - b. Object of Activity class
  - c. It is a type of intent which allows the communication between a function and a service
  - d. None of these

2. Which of the following class is extended while implementing broadcast receivers?
- a. Broadcast Receiver class
  - b. Activity class
  - c. Context wrapper class
  - d. None of these

---

## 2.8 Let Us Sum Up

---

While studying this unit, we have learnt that Android uses new design metaphor inspired by paper and ink that provides a reassuring sense of tactility. The great thing about Android is that it runs on a bunch of devices. Android is the greatest platform for mobile devices which has capability to run millions of mobile phones in about more than 200 countries. It not only provides a great support for people around world to use hundreds of games and apps in their phones but also provides a huge open marketplace to developers for Android App Development.

An Activity is an application component that provides a screen with which users can interact in order to do something, such as dial the phone, take a photo, send an email, or view a map. A Service is an application component representing either an application's desire to perform a longer-running operation while not interacting with the user or to supply functionality for other applications to use. An Intent is a messaging object you can use to request an action from another app component. Although intent facilitate communication between components in several ways.

Permission requests should be simple, transparent, and understandable. When requesting access, apps should ensure that either the feature itself or an explanation provided makes it clear why permission is needed.

---

## 2.9 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –d)

**Check your progress 2**

**Answers:** (1 -d), (2-c)

**Check your progress 3**

**Answers:** (1 -b), (2-a)

**Check your progress 4**

**Answers:** (1 -b), (2-d)

**Check your progress 5**

**Answers:** (1 -c)

**Check your progress 6**

**Answers:** (1 -c), (2-a)

---

## 2.10 Glossary

---

1. **Activity** - An application screen that supports Java code from Activity class.
2. **Application** - In Android application, there are activities, services, listeners and intent receivers.
3. **Intent** - Message object which uses to communicate with certain applications/activities asynchronously.
4. **Service** - It is application component representing either an application's desire to perform a longer-running operation while not interacting with the user or to supply functionality for other applications to use.

---

## 2.11 Assignment

---

Write short note on Activities in android.

---

## 2.12 Activities

---

Try to create activity and services using Android SDK.

---

## 2.13 Case Study

---

Study Android Manifest File of your project.

---

## 2.14 Further Readings

---

1. Learn Java for Android Development (2nd edition), Jeff Friesen, 2013.
2. Android Application Development for Java Programmers, James C. Sheusi, 2012.

---

## Block Summary

---

In this block, you have learnt and understand about the basic of Intents along with their working. The block gives an idea on the study and concept of various permission strategies in building an application in android. You have been well explained on the concepts of various android application formats.

The block detailed about the basic of Java in designing of Android applications with study about its targeted components. The concept related to installation regarding Java studio along with screenshots helps the students to understand better about development of application. You will be demonstrated practically with app component.



---

## **Block Assignment**

---

### **Short Answer Questions**

1. What do you understand by Activity in Android?
2. Explain Intents
3. Create a new Project in Android SDK and describe it briefly
4. What is Android Manifest File?

### **Long Answer Questions**

1. Explain Services in Android in detail
2. Describe Android Intent filters
3. Create an activity in Android SDK to register user and shows his/her detail.

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# OBJECT ORIENTED ANALYSIS AND DESIGN

PGDCA -204

## BLOCK 1: OBJECT ORIENTED MODELLING

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# OBJECT ORIENTED ANALYSIS AND DESIGN

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

PROF. K J Sharma

### **Language Editor**

Mr. Vipul Shelke

## **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

## **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by means of, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

## **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)





## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# OBJECT ORIENTED ANALYSIS AND DESIGN

## Contents

---

### **BLOCK 1: OBJECT ORIENTED MODELLING**

#### **UNIT 1 INTRODUCTION TO OBJECT ORIENTED MODELLING**

Object Oriented Modeling, Characteristics Object Oriented Modeling: Class and Objects, Links and Association, Generalization and Inheritance, An Object Model. Benefits of OO Modeling, Introduction to OOAD tools,

#### **UNIT 2 ADVANCE MODELING CONCEPTS**

Aggregation, Abstract Class, Multiple Inheritance, Generalization as an Extension, Generalization as a Restriction, Metadata, Constraints, An Object Model

---

### **BLOCK 2: OBJECT ORIENTED ANALYSIS AND SYSTEM DESIGN**

#### **UNIT 1 OBJECT ORIENTED ANALYSIS**

Object Oriented Analysis, Problem Statement: an Example, Differences between Structured Analysis and Object Oriented Analysis, Analysis Techniques: Object Modeling, Dynamic Modeling, and Functional Modeling. Adding Operations. Analysis Iteration

#### **UNIT 2 SYSTEM DESIGN**

System Design: An Object Oriented Approach, Breaking into Subsystems, Concurrency Identification, Management of data store, Controlling events between Objects, Handling Boundary Conditions



**BLOCK 3: OBJECT DESIGN AND DYNAMIC MODELING**

**UNIT 1 OBJECT DESIGN**

Object Design for Processing, Object Design Steps, Designing a Solution, Choosing Algorithms, Choosing Data Structures, Defining Classes and delegation of Responsibilities to Methods

**UNIT 2 DYNAMIC MODELING**

Events, State and State Diagram, Elements of State Diagrams, Examples of State Diagrams, Advance Concepts in Dynamic Modeling, Concurrency, A Dynamic model

**BLOCK 4: FUNCTIONAL MODELING AND UML**

**UNIT 1 FUNCTIONAL MODELING**

Functional Models, Data Flow Diagrams, Features of a DFD, Design flaws in DFD, A Functional model, Relationship between Object, Dynamic, and Functional Models

**UNIT 2 USING UML**

UML: Introduction, Object Model Notations: Basic Concepts, Structural Diagrams: Class, Object, Composite, Package, Component, Deployment. Behavioral Diagrams: Use Case, Communication, Sequence, Interaction Overview, Activity, State. Modeling with Objects

**UNIT 3 CASE STUDY**

This unit will cover all the OOAD aspects Covered in previous units of this course.



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA-204

# **OBJECT ORIENTED ANALYSIS AND DESIGN**

---

## **BLOCK 1: OBJECT ORIENTED MODELLING**

---

### **UNIT 1**

|                                           |    |
|-------------------------------------------|----|
| INTRODUCTION TO OBJECT ORIENTED MODELLING | 03 |
|-------------------------------------------|----|

### **UNIT 2**

|                           |    |
|---------------------------|----|
| ADVANCE MODELING CONCEPTS | 17 |
|---------------------------|----|

---

# **BLOCK 1:      OBJECT                      ORIENTED MODELING**

---

## **Block Introduction**

Programming languages make it easier to design and implement programming ideas, by ensuring that you don't have to learn binary, which is the series of on/off or 1/0 commands the computer uses to calculate everything. The JVM is an abstract instead of actual machine or processor. It specifies certain instructions set, register sets, stack, garbage heap as well as required method. The Java program is a set of instructions which a computer understand.

In this block, we will detail about object oriented modelling features as new thinking technology for problem solving with knowledge on visualization in real world. The block will focus on the study and concept of generalization and inheritance abstractions in terms of structure and behaviour sharing of classes and relationship. You will give an idea on metadata with respect to data or data describing other data.

In this block, you will make to learn and understand about the basic of Inheritance process in terms of object properties and usability in information. The concept related to generalization with its shared characteristics in terms of classes and super class. The shared characteristics with respect to attributes, associations or methods are well detailed to you.

## **Block Objective**

**After learning this block, you will be able to understand:**

- About features of Object Oriented Modeling
- About advantages of Object Oriented Modeling
- About qualities of Class and Objects
- About qualities of Links and Association
- About Object and Class
- About Multiple Inheritance
- About generalization as a Restriction

## **Block Structure**

**Unit 1:     Introduction to Object Oriented Modelling**

**Unit 2:     Advance Modelling Concepts**

---

# **UNIT 1: INTRODUCTION TO OBJECT ORIENTED MODELLING**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Object Oriented Modelling**
- 1.3 Characteristics Object Oriented Modelling: Class and Objects**
- 1.4 Links and Association**
- 1.5 Generalization and Inheritance**
- 1.6 An Object Model**
- 1.7 Benefits of OO Modelling**
- 1.8 Introduction to OOAD tools**
- 1.9 Let Us Sum Up**
- 1.10 Answers for Check Your Progress**
- 1.11 Glossary**
- 1.12 Assignment**
- 1.13 Activities**
- 1.14 Case Study**
- 1.15 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- The basics of object oriented Modeling
- Define Objects and Classes
- Explain the concepts of links and Associations
- Explain the concept of Generalization and Inheritance
- Describe benefits of Object Oriented Modeling

---

## 1.1 Introduction

---

Object-oriented technology is incredibly wide and extensive. The users of computer systems found the effects of such technology as increasingly easy-to-use software applications and operating systems that are flexible that's catered by many industries like banking, telecommunications and television. Just in case of software technology, such object-oriented technology can cover object-oriented management of projects, computer hardware and computer aided software engineering, among others.

An interface is mixture of abstract methods. A class implements an interface, and can inherit abstract ways of interface. It's not a class, however if we tend to write it, it's equivalent to writing a class. Any it's seen that a class describes itself as an attributes and behaviours of an object. An interface contains behaviours that a class implements.

---

## 1.2 Object Oriented Modelling

---

It is one of the most objectives of the software engineering discipline to support the complicated and thus error-prone software development task by providing appropriate sophisticated concepts, languages, techniques, and tools to any or all stakeholders involved. A crucial and nowadays commonly accepted approach within software engineering is that the usage of a software development process model, where above all the software development task is separated into a series of dedicated subtasks. a substantial constituent of such a stepwise approach is the development of a system model. Such a model describes the requirements for the software system to be realized and forms an abstraction.

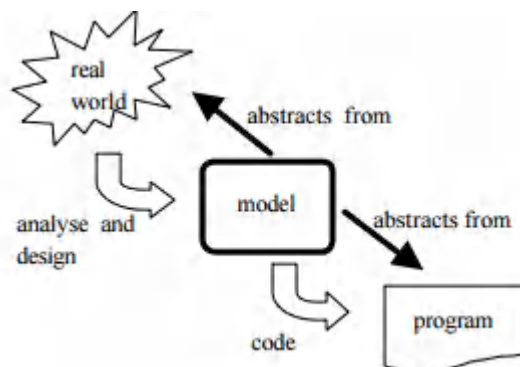


**Fig 1.1 Model Representations**

The success of object-oriented modelling approaches was hindered within the beginning of the 90ies due to the very fact that surely over 50 object-oriented



modelling approaches claimed to be the proper one. This so-called technique war came to an end through an industrial initiative, that pushed the development of the meanwhile industrially standardized object-oriented modelling language UML.

### **Check your progress 1**

1. What is object oriented modeling strategy?
  - a. Use of algorithms
  - b. Use of objects
  - c. Use of classes
  - d. Both B and C

---

## **1.3 Characteristics Object Oriented Modelling: Class and Objects**

---

Object oriented modelling is entirely a brand new way of thinking about problems. This methodology is all concerning visualizing the items by using models organized around real world ideas. Object oriented models help in understanding issues, communicating with experts from a distance, modelling enterprises, and designing programs and database. In object oriented modelling objects and their characteristics are described. In any system, objects come into existence for playing some role. In the method of defining the roles of objects, some options of object orientation are used.

### **Class and Objects**

A class is a collection of things, or ideas that have the same characteristics. Each of these things or concepts is named as object. Classes define the basic words of the system being modelled. Using a set of classes as the core vocabulary of a software project tends to greatly facilitate understanding and agreement concerning the meanings of terms, and other characteristics of the objects in the system. Classes can serve as the foundation for data modelling. In OOM, the term classes is sometimes the base from that visual modelling tools—such as Rational Rose XDE, Visual Paradigm function and design the model of systems.

A class could be a pattern, template, or blueprint for a category of structurally identical items. The items created using the class are known as instances. This is

often referred to as the "class or `cookie cutter'" view. As you might guess, the instances are the "cookies."

A class could be a thing that consists of both a pattern and a mechanism for creating items supported that pattern. This is the "class as an `instance factory'" view; instances are the individual items that are "manufactured" using the class's creation mechanism.

A class is the set of all items created using a specific pattern. In another way, the class is that the set of all instances of that pattern.

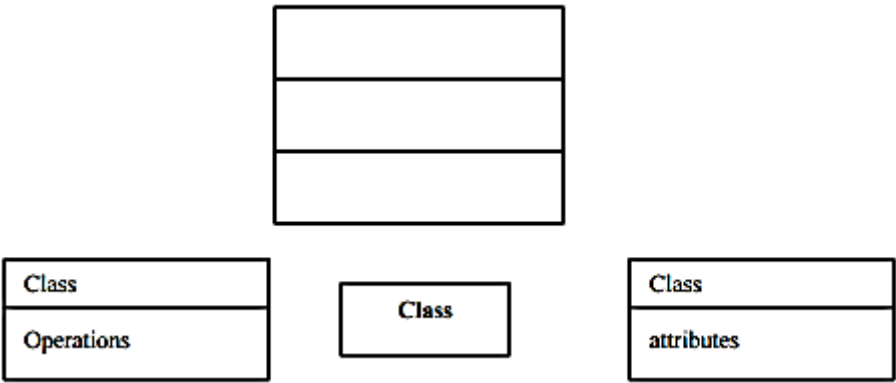


Fig 1.2 Class Notations

**Objects**

Objects are physical as they are present in universe around us. It is found that hardware; software, documents and concepts are all object samples. For use of modelling, an officer views staff, buildings, divisions, documents and benefits packages as related objects. An automobile person would see tires, doors, engines, speed and fuel level in terms of objects, while atoms, molecules, volumes, and temperatures are objects according to chemist which can be thought of as making an object oriented simulation of chemical reaction.

Normally it is viewed that objects can be considered as particular state. The state of an object is basically the condition of an object, or set of circumstances describing the item. It is generally seen that that people are talking about state information which in particular relates to specific object. It is noticed that state of bank account object will cover latest and available balance, which shows state of clock object that is available at required time showing state of lightweight bulb which could be placed at on or off state. For complex objects like a human being or an automobile, an entire description of the state may be very complex. Fortunately, when we use objects to model real world or imagined things, we have

a tendency to generally restrict the possible states of the objects to only people who are relevant to our models.

### **Check your progress 2**

1. What is class?
  - a. Collection of similar objects.
  - b. Collection of different types of objects
  - c. Group of information
  - d. None of these

---

## **1.4 Links and Association**

---

Links and association are modes of creating link which exists among objects and classes. It is noticed that both links and association bears similar feature whereas links establishes among objects while association establishes among class.

**Links:** In object modelling links provides a relationship between the objects. These objects or instance may be same or different in data structure and behaviour. Thus a link is a physical or conceptual connection between instances (or objects).

**Associations:** the object modelling describes as a group of links with common structure and customary semantics. All the links among the object are the varieties of association among the same classes. The association is that the relationship among classes.

1. Association
2. Association with inverse direction
3. Association between student and university

Degree of association is:

1. Unary association (degree of one)
2. Binary Association (degree of two)
3. Ternary Association (degree of three)

4. Quaternary Association (degree of four)
5. Higher order association (more than four)

### **Check your progress 3**

1. What is meant by link in object modeling?
  - a. Way to describe association
  - b. Physical or conceptual connection between instances
  - c. Relationship between classes
  - d. None of these

---

## **1.5 Generalization and Inheritance**

---

Generalization and inheritance are powerful abstractions for sharing the structure and/or behaviour of one or a lot of classes. Generalization is that the relationship between a class and it defines a hierarchy of abstraction in which subclasses (one or more) inherit from one or more super classes. The notation for generalization is a triangle connecting a super class to its subclasses. The super class is connected by a line to the top of the triangle. The subclasses are connected by lines to a horizontal bar attached to the base of the triangle.

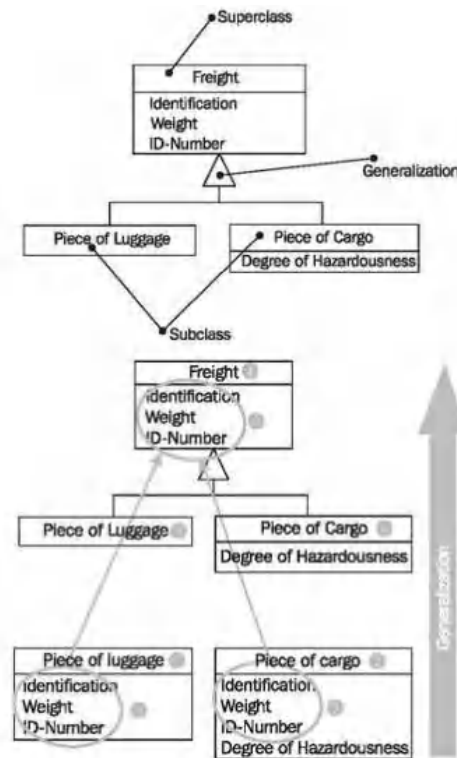


**Fig 1.3 Generalisation Concepts**

Inheritance is taken within the sense of code reuse within the object oriented development. Throughout modelling, we look at the resulting classes, and try to group similar classes along so code utilize can be enforced. Generalization, specialization, and inheritance have terribly close association. Generalization is used to refer to the relationship among classes, and inheritance is employed for sharing attributes and operations using the generalization relationship.

Inheritance and generalisation have in common that they both have the ability to define an abstract "contract" or "protocol" that a group of concrete classes following this "contract" have to implement. this gives you the ability to handle a number of comparable classes in an exceedingly uniform way, that permits you to write down additional compact, abstract code, that once more permits code reuse. Consider a class implements interface example where one interface extends another interface as:

```

interface Printable{
void print();
}

interface Showable extends Printable{
void show();
}
  
```

```
}  
class Testinterface2 implements Showable  
{  
    public void print(){System.out.println("Sanjay");}  
    public void show(){System.out.println("Mathur");}  
    public static void main(String args[]){  
        Testinterface2 obj = new Testinterface2();  
        obj.print();  
        obj.show();  
    }  
}
```

If we run this program we get an output as:

Sanjay

Mathur

### Check your progress 4

1. What is inheritance?
  - a. Code reuse within object modeling
  - b. Getting properties of one class into another class
  - c. It is meant for sharing attributes.
  - d. All of these

---

## 1.6 An Object Model

---

The object model visualizes all the elements in a software application in terms of objects. The example of Object Model is shown in fig 1.3

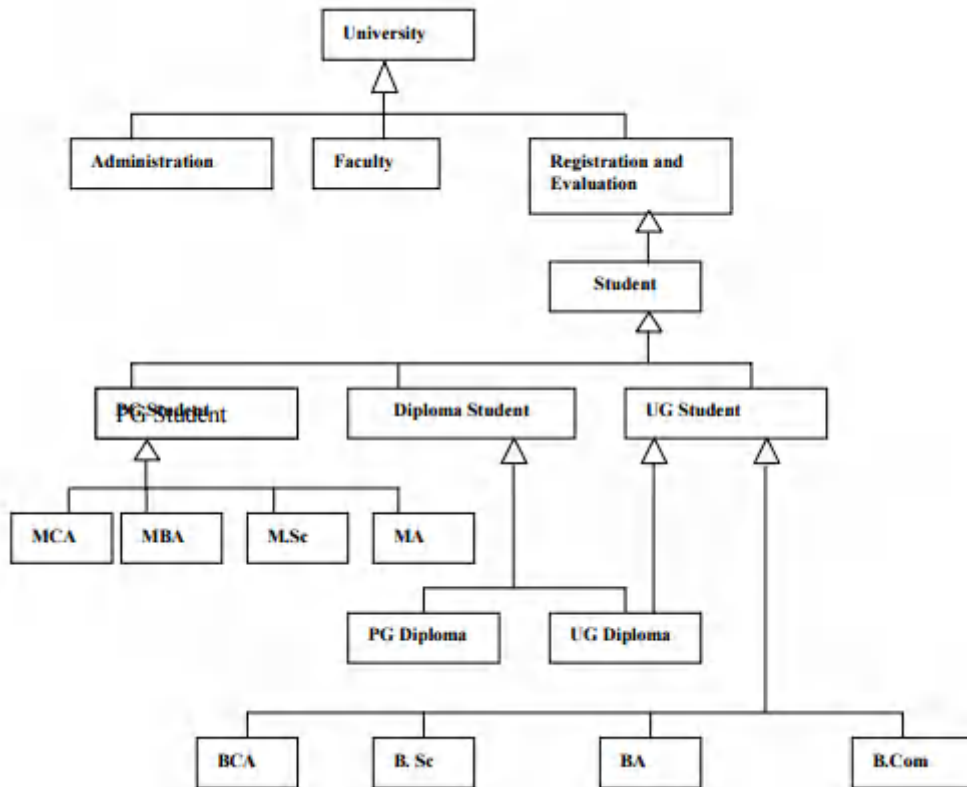


**Fig 1.3 Object Model**

- An object contains values stored in instance variables within the object.
- Unlike the record-oriented models, these values are themselves objects.
- Thus objects contain objects to an arbitrarily deep level of nesting.
- An object also contains bodies of code that operate on the the object.
- These bodies of code are called methods.
- Objects that contain the same types of values and the same methods are grouped into classes.
- A class may be viewed as a type definition for objects.
- Analogy: the programming language concept of an abstract data type.
- The only way in which one object can access the data of another object is by invoking the method of that other object.
- This is called sending a message to the object.

### Check your progress 5

1. What is meant by sending message to object?
  - a. Passing parameters to object
  - b. Invoking the method of another object
  - c. Passing one object definition to another
  - d. None of these

---

## 1.7 Benefits of OO Modelling

---

Here are a number of the advantages of the object-oriented approach:

**Reduced Maintenance:** the first goal of object-oriented development is that the assurance that the system can enjoy a longer life while having far smaller maintenance costs. As a result of most of the processes among the system is encapsulated, the behaviours is also reused and incorporated into new behaviours.

**Real-World Modelling:** Object-oriented system tends to model the real world in a lot of complete fashion than do traditional methods. Objects are organized into classes of objects, and objects are related to behaviours. The model relies on objects, rather than on data and processing.

**Improved reliability and Flexibility:** Object-oriented system promise to be far more reliable than traditional systems, primarily because new behaviours is "built" from existing objects. Because objects is dynamically called and accessed, new objects could also be created at any time. The new objects might inherit data attributes from one, or several other objects. Behaviours are also inherited from super-classes, and novel behaviours are also added without effecting existing systems functions.

**High Code Reusability:** once a new object is created, it'll automatically inherit the data attributes and characteristics of the class from that it absolutely were spawned. The new object will inherit the data and behaviours from all super classes in which it participates. Once a user creates a new type of a widget, the new object behaves "wigitty", while having new behaviours that are defined to the system.



### Check your progress 6

1. What are the benefits of object oriented modeling?
  - a. Improved reliability
  - b. Improved flexibility
  - c. Reduced maintenance
  - d. All of these

---

## 1.8 Introduction to OOAD tools

---

The major phases of software development using object-oriented methodology are object-oriented analysis, object-oriented design, and object-oriented implementation. Object-Oriented Analysis is a stage where the problem is formulated, user requirements are identified, and model is created on real-world objects. The analysis produces models on how the desired system should function and how it must be developed. The models do not include any implementation details so that it can be understood and examined by any non-technical application expert. Object-Oriented Design includes two main stages, namely, system design and object design.

There are three main tools used in object-oriented analysis and design techniques:

- Class diagrams/templates.
- Object diagrams.
- Object state diagrams

Class diagrams are used to model key abstractions in the problem domain and their relationships with each other. Object diagrams are used to model the interactions between objects, whereas object state diagrams model the dynamic behaviour within a single object. An object state diagram shows all the possible states of an object and the allowed transition between the states.

### **Check your progress 7**

1. What are the two main stages of object –oriented design?
  - a. System design
  - b. Object design
  - c. Both of these
  - d. None of these

---

## **1.9 Let Us Sum Up**

---

In this unit we have learnt that object oriented technology is wide and extensive which increases easy-to-use software applications with operating systems that is flexible. It is seen that object oriented modelling is brand new way of thinking about problems in which you can visualize items by modelling that are arranged around real world ideas.

It is known that class consists of pattern and mechanism for creating items which supports pattern where class as an instance factory can be seen as an instance of individual items manufactured by class creation mechanism. We have studied that links and association are modes of creating link that exists in objects and classes with both links and association bears same feature and establishes link among objects and association establishes class.

The generalization and inheritance are strong abstractions for sharing structure and behaviour of classes and bears relationship among class and show branch of abstraction where subclasses inherit from super classes. The object oriented analysis is particular stage where problem gets framed with identification of user requirements and creation of modelling in real world objects.

---

## **1.10 Answers for Check Your Progress**

---

### **Check your progress 1**

**Answers:** (1 –d)

### **Check your progress 2**

**Answers:** (1 –a)

**Check your progress 3**

**Answers:** (1 –b)

**Check your progress 4**

**Answers:** (1 –d)

**Check your progress 5**

**Answers:** (1 –b)

**Check your progress 6**

**Answers:** (1 –d)

**Check your progress 7**

**Answers:** (1 –c)

---

## 1.11 Glossary

---

1. **Package** - It is a collection of types that gives access protection and name space management in Java.
2. **Interface** - In programming, interface is a mixture of abstract methods where class implements an interface.

---

## 1.12 Assignment

---

Explain the Object oriented modelling and its advantages.

---

## 1.13 Activities

---

Study OOAD tools.

---

## **1.14 Case Study**

---

Study the difference between generalisation and specialisation and also justify how they are useful in Object oriented modelling

---

## **1.15 Further Readings**

---

1. Learning Programming by Peter Norvig's.
2. Approach programming with a more positive by P.Brian.Mackey.

---

## **UNIT 2: ADVANCE MODELING CONCEPTS**

---

### **Unit Structure**

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 Aggregation**
- 2.3 Abstract Class**
- 2.4 Multiple Inheritance**
- 2.5 Generalization as an Extension**
- 2.6 Generalization as a Restriction**
- 2.7 Metadata**
- 2.8 Constraints**
- 2.9 An Object Model**
- 2.10 Let Us Sum Up**
- 2.11 Answers for Check Your Progress**
- 2.12 Glossary**
- 2.13 Assignment**
- 2.14 Activities**
- 2.15 Case Study**
- 2.16 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Study about features of Object Model
- Study about advantages of Abstract Class
- Study about Multiple Inheritance
- Study about Extension and Restriction

---

## 2.1 Introduction

---

Inheritance are often defined as the process wherever one object acquires the properties of another. With the use of inheritance the information is made manageable in a hierarchical order. Once we talk about inheritance, the most unremarkably used keyword would be extends and implements. These words would determine whether one object IS-A style of another. By using these keywords we will create one object acquires the properties of another object. Inheritance may be a compile-time mechanism. A super-class will have any number of subclasses. However a subclass will have just one super class. This is because Java does not support multiple inheritances.

---

## 2.2 Aggregation

---

Aggregation is an extension of association mean aggregation is a strong form of association within which an aggregate object is formed of components. Components are a part of the aggregate. So aggregation is the “part-whole” or “a-part-of” relationship.

Aggregation is a special case of association, a directional association between objects. When an object ‘has-a’ another object, then you have got an aggregation between them. Direction between them specified that object contains the other object. Aggregation is additionally known as a “Has-a” relationship.

### Check your progress 1

1. What is Aggregation?
  - a. IS-A relationship
  - b. a-part-of relationship
  - c. Has-a relationship
  - d. Both B and C

---

## 2.3 Abstract Class

---

In Java, a class is a group of objects having certain common properties. It is a sort of template or blueprint from where objects are created. So a class in java contains:

- Data member
- Method
- Constructor
- Block
- Class and interface

The syntax to declare a class is:

```
Class <class_name>{  
    Data member;  
    method  
}
```

### Consider an example of Object and Class

In this example, a Student class is created having two data members' id and name. Now the object of Student class is created with the help of new keyword and printing objects value as:

```
class Student1{  
    int id;//data member (also instance variable)  
    String name;//data member(also instance variable)  
  
    public static void main(String args[]){  
        Student1 s1=new Student1();//creating an object of Student  
        System.out.println(s1.id);  
        System.out.println(s1.name);  
    }  
}
```

If we run the above program, we find:

Output: 0 null

### Abstract Class

An abstract class defines an abstract concept that can't be instantiated. we can't create object of abstract class, it will only be inherited. Abstract class

normally represents concept with general actions associated with it. Abstract class can't be instantiated; it will solely be inherited while interfaces should be implemented. Abstract class will have implemented methods which interfaces will have only definitions of the methods without implementation.

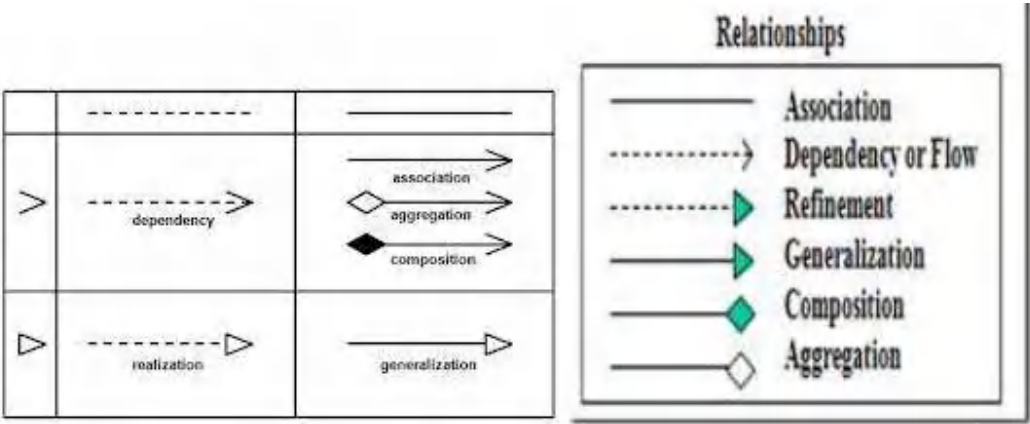


Fig 2.1 Abstract Class relationship

**Check your progress 2**

1. Abstract class are \_\_\_\_\_.
  - a. Inherited
  - b. Implemented
  - c. Instantiated
  - d. None of these

**2.4 Multiple Inheritance**

Inheritance is the most important features of Object oriented Programming that permits a {class|a category} to use properties and methods of another class. In this, the derived class is named as subclass and base class is named as super-class. It's seen that a derived class adds additional variables and methods which can differentiate derived class from base class. The syntax is shown below:

```
public class ChildClass extends BaseClass {  
    // derived class methods extend and possibly override  
}
```



### Consider an example:

```
// A class to display the attributes of the vehicle
class Vehicle {
    String color;
    int speed;
    int size;
    void attributes() {
        System.out.println("Color : " + color);
        System.out.println("Speed : " + speed);
        System.out.println("Size : " + size);
    }
}

// A subclass which extends for vehicle
class Car extends Vehicle {
    int CC;
    int gears;
    void attributescar() {
        // The subclass refers to the members of the superclass
        System.out.println("Color of Car : " + color);
        System.out.println("Speed of Car : " + speed);
        System.out.println("Size of Car : " + size);
        System.out.println("CC of Car : " + CC);
        System.out.println("No of gears of Car : " + gears);
    }
}

public class Test {
    public static void main(String args[]) {
        Car b1 = new Car();
        b1.color = "Blue";
        b1.speed = 200 ;
        b1.size = 22;
        b1.CC = 1000;
        b1.gears = 5;
        b1.attributescar();
    }
}
```

If we run the above program, we get:

```
Color of Car : Blue
Speed of Car : 200
Size of Car : 22
CC of Car : 1000
No of gears of Car : 5
```

Inheritance results as an effective method which will share code among various classes having some traits in common by allowing classes to have different parts. Fig 2.2 called as Vehicle class which carries two subclasses as Car and Truck.

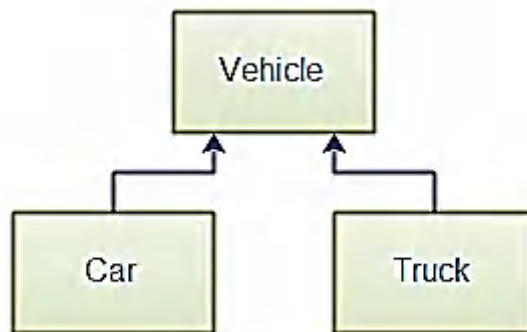


**Fig 2.2 Vehicle class**

It is seen that Vehicle class belongs to superclass of Car and Truck that are subclasses of Vehicle. Here, the Vehicle class contain those fields and methods which are Vehicles needed whereas Car and Truck have fields and methods particular to Cars and Trucks.

It is seen that many people declare that inheritance is a way to categorize particular class. It is studied that a Car is a Vehicle where a Truck is also a Vehicle. So, it is not how you determine super classes and subclasses in your application. It simply shows how you need to work with them.

As seen, when a subclass extends a super class, then all protected and public fields and methods of it gets inherited by subclass. In this, the fields and methods are part of subclass, as if subclass declared itself.

Multiple Inheritance is a technique where one class extends more than one base class.

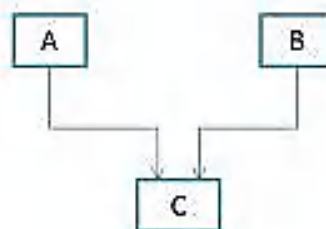


**Fig 2.3 Multiple inheritance**

The latter point is an advanced version of inheritance and is called multiple inheritance. Multiple inheritance provides an alternative approach where the new

class can inherit information from the two relevant classes. This is shown in the figure 2.4.

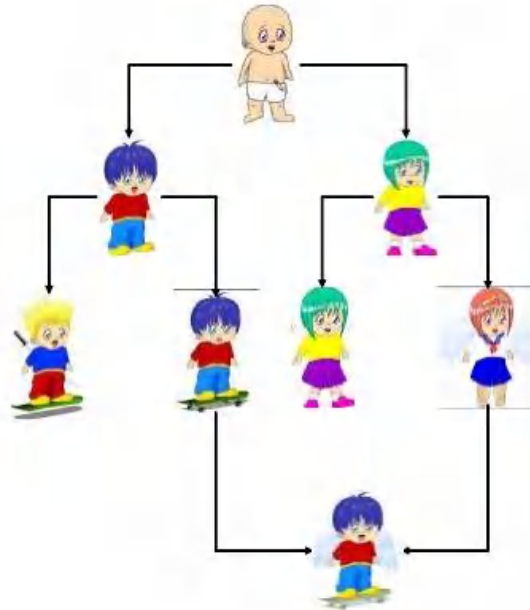


**Fig 2.4 Multiple Inheritance**

In the above figure we noticed some inconsistencies, where sloopy at the bottom has wheels and wings that are inherited from rim sloopy and float sloopy. In this, the boy is sloopy and not the girl and also wheels float sloopy inherited attributes of boy and girl which appear to be a conflict.

### Check your progress 3

1. Select the correct option if class A is inherited by class B.
  - a. class B + class A
  - b. class B inherits class A
  - c. class B extends A
  - d. class B extends class A

## 2.5 Generalization as an Extension

Generalization is a process of defining a super class from a given set of semantically related entity set. Generalization is the process of extracting shared characteristics from two or more classes, and combining them into a generalized super class. Shared characteristics can be attributes, associations, or methods.

Generalization is the relationship between a class and one or more refined versions of it. The class being refined is called the “superclass” and each refined version is called a “subclass”. For example, Equipment is the superclass of Pump and tank. Attributes and operations that are common to a group of subclasses are attached to the superclass and shared by each subclass. Generalization is sometimes called an “is-a” relationship. Each instance of a subclass is an instance of the superclass.

#### **Check your progress 4**

1. What is generalization?
  - a. Relationship between one class and its refined versions
  - b. IS-A relationship
  - c. Defining super class from given set of related entities
  - d. All of these

---

## **2.6 Generalization as a Restriction**

---

In generalization, an instance of a class is an instance of a class is an instance of all ancestors of the class. Therefore you can say that all ancestor class features must apply to the subclass instances. This includes not only the attributes on the ancestor classes but also the operations on the ancestor class.

A subclass may include many features, which is called as an extension. For example, fig.2.5 extends class Employee with three subclasses that inherit all Employee features and add new features of their own.

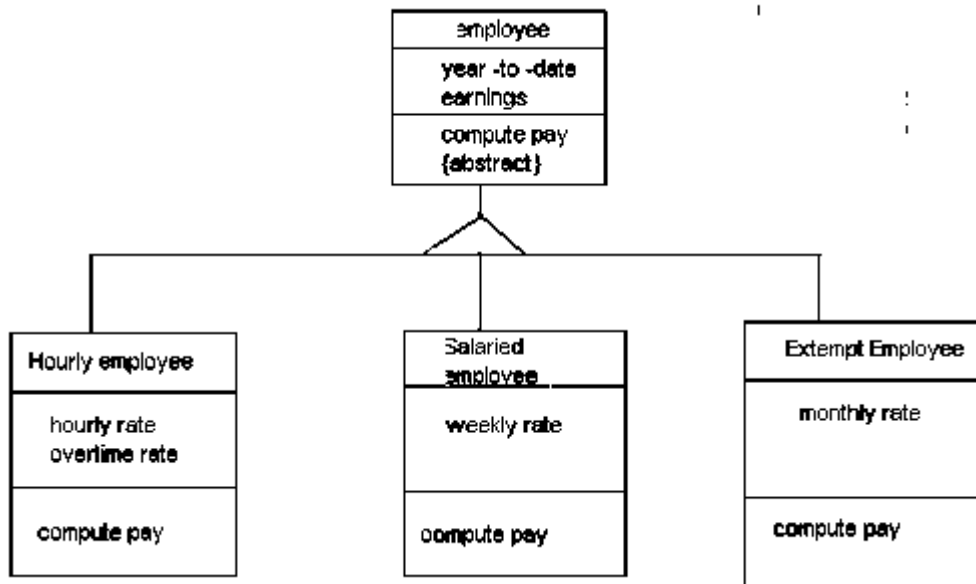


Fig 2.5 Class Employee

A subclass may also constrain ancestor attributes. This is called restriction because it restricts the values that instances can assume. For example, a circle is an ellipse that's major and minor axes are equal. Arbitrary changes to the attribute values of a restricted subclass may cause it to violate the constraints, such that the result no longer belongs to the original subclass. This is not a problem from the perspective of the super class because the result is still a valid super class instance.

### Check your progress 5

1. What is meant by restriction in generalization?
  - a. Attribute
  - b. All ancestor class features must apply to the subclass instances, this is known as restriction
  - c. Inherited features
  - d. None of these

## 2.7 Metadata

Metadata is called as data about data or data that describes about other data. It is everywhere certainly where many data is useless without metadata as it is also data. Metadata summarizes basic information about data, which can make finding

and working with particular instances of data easier. In addition to document files, metadata is used for images, videos, spreadsheets and web pages. The use of metadata on web pages can be very important. Metadata for web pages contain descriptions of the page's contents, as well as keywords linked to the content. These are usually expressed in the form of metatags. The metadata containing the web page's description and summary is often displayed in search results by search engines, making its accuracy and details very important since it can determine whether a user decides to visit the site or not.

### **Check your progress 6**

1. Which of the following is true about meta data?
  - a. It is the data that describes about other data
  - b. It summarizes basic information about data
  - c. It is expressed in the form of metatags
  - d. All of these

---

## **2.8 Constraints**

---

A constraint could be a numeric or geometric relationship between objects. Constraints have a declarative nature. Constraints are a natural manner for describing relationships between objects. Combining constraint systems and object-oriented programming (OOP) seems hard. All existing systems implicitly compromise the encapsulation principle of OOP.

Constraints describe properties that have to be true at each moment in time for the entire system, without determining however they're to be preserved. The specification of constraints is mostly done by using informal text, operational restrictions or integrating constraints in existing model ideas.

Constraints are a way to specific general properties for the system, without specifying however they're being realized.

### Check your progress 7

1. What can constitute constraints?
  - a. Numeric data
  - b. Geometric relationship
  - c. Both of these
  - d. None of these

---

## 2.9 An Object Model

---

An object model is typically a logical interface, software or system that's modelled through the utilization of object-oriented techniques. It permits the creation of an architectural software or system model before development or programming.

An object model is a component of the object-oriented programming (OOP) lifecycle.

Following are the benefits of using the object model are:

- It helps in quicker development of software.
- It is straightforward to maintain. Suppose a module develops an error, then a programmer will fix that specific module, whereas the other elements of the software are still up and running.
- It supports relatively hassle-free upgrades.
- It enables reuse of objects, designs, and functions.
- It reduces development risks, significantly in integration of advanced systems.

### Check your progress 8

1. Why do there is a need to use object model?
  - a. It helps in faster development of software
  - b. reusability
  - c. reduces development risk
  - d. All of these

---

## 2.10 Let Us Sum Up

---

In this unit we have learnt that Inheritance can be defined as the process where one object acquires the properties of another. These inheritances make the information to manage in a hierarchical order. It is studied that a Subclass in Java is a method of inheritance which appears from Java superclass.

There exist three types of variables in Java as Local, Instance and Static. It is studied that local variable is a variable which is declared within method itself, instance variable is a variable which is declared within class but outside method and static variable is a variable which is declared as static is called static variable.

In Java, it is seen that constructor can be inherited as the constructor name is based on class name. While inheriting methods, the method signature should remain same.

---

## 2.11 Answers for Check Your Progress

---

### Check your progress 1

**Answers:** (1 –d)

### Check your progress 2

**Answers:** (1 –a)

### Check your progress 3

**Answers:** (1 –c)



**Check your progress 4**

**Answers:** (1 –d)

**Check your progress 5**

**Answers:** (1 –b)

**Check your progress 6**

**Answers:** (1 –d)

**Check your progress 7**

**Answers:** (1 –c)

**Check your progress 8**

**Answers:** (1 –d)

---

## 2.12 Glossary

---

1. **Static variable** - It is a variable which is declared as static is called static variable
2. **Class** - These are group of objects with common properties and are like a blueprint from where objects are created.

---

## 2.13 Assignment

---

Write a short note on inheritance and its use.

---

## 2.14 Activities

---

Study abstract classes and its usage

---

## 2.15 Case Study

---

Discuss the output of this program?

```
class A {  
    int i;  
    int j;  
    A() {  
        i = 1;  
        j = 2;  
    }  
}  
class Output {  
    public static void main(String args[])  
    {  
        A obj1 = new A();  
        A obj2 = new A();  
        System.out.print(obj1.equals(obj2));  
    }  
}
```

---

## 2.16 Further Readings

---

1. DOWLATSHAHI, S., 1992, Product design in a concurrent engineering environment: an optimization approach, International Journal of Production Research, 30(8), pp. 1803–1818.
2. ESCHENAUER, H., KOSKI, J. and OSYCZKA, A., 1990, Multicriteria Design Optimization: Procedures and Applications (New York: Springer-Verlag).
3. FENG, C. X. and KUSIAK, A., 1995, Constraint-based design of parts, Computer-Aided Design, 27(5), pp. 343–352.

---

## **Block Summary**

---

In this block, you will be detailed with knowledge about Inheritance and its relationship with subclass and interface block. The concepts about types of variables with illustrations are explained in step by step manner. The block stress on information related to packages and interface with examples about inherit abstract methods of interface. The information related to declaring a reference variable of interface is well explained with examples.

After studying this block, you will feel self confident while working on simple Java platform and can enhance their knowledge by studying certain examples and illustrations mentioned in this block. With such detailed knowledge on Inheritance, Interface, Packages and Exceptions in Java, you can be benefitted in future.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. What is object oriented modeling?
2. Explain inheritance and its types
3. What is the difference between generalisation and specialisation?
4. Write short note on Abstract classes

### **Long Answer Questions**

1. What is multiple inheritance. Explain in detail with the help of an example
2. Write short note on metadata.
3. Write note on generalisation as a restriction

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                              | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|----------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| <b>Presentation Quality</b>                        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Language and Style</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Illustration used<br/>(Diagram, tables etc)</b> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Conceptual Clarity</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Check your progress<br/>Quest</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Feed back to CYP<br/>Question</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

'Jyotirmay Parisar', Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# Fundamentals of Computer Networking (FCN)

PGDCA 201



**BLOCK 1:  
NETWORKING CONCEPT**

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# Fundamentals of Computer Networking (FCN)

---



Knowledge Management and  
Research Organization  
Pune





## **Editorial Panel**

### **Author**

Mr. Sanjay Thapar

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# FUNDAMENTALS OF COMPUTER NETWORKING

## (FCN)

### Contents

---

#### **BLOCK 1: NETWORKING CONCEPT**

##### **UNIT 1 INTRODUCTION AND NETWORKING BASICS**

Advantages of computer networking, computer networks and the Internet, WAN, LAN and PAN basics, Topologies, Connecting Media: Wired and Wireless and their characteristics, Introduction to NIDs and their specifications

##### **UNIT 2 NETWORK INTERFACE DEVICES**

Network Adaptor Cards (both wired and wireless), Hubs, Switches, Routers, Access Points (Wireless), Repeaters. Their basic architecture, working and use/application, understanding their technical specifications/data sheets.

---

#### **BLOCK 2: CREATING WIRED AND WI-FI LAN**

##### **UNIT 1 CREATING A SWITCHED WIRED ETHERNET LAN**

Introduction to UTP CAT series cables, RJ-45 connectors, color coding scheme, crimping a UTP cable to RJ-45 connector, physically connecting individual nodes to the switch, selection of server machine, Windows 8.1 Server Installation and Configuration on Server Machine, Windows 8.1 Desktop installation and configuration on client nodes, checking connectivity, basic troubleshooting/diagnostic commands.

##### **UNIT 2 CREATING A WI-FI LAN:**

Introduction to Wi-Fi Technology, how to provide Wi-Fi capability to a PC, creating an ad-hoc Wi-Fi based LAN, creating an infrastructure based LAN using Wireless AP, configuration of AP and client Machines, accessing data from File Server through Wi-Fi Interface from client machine.

**BLOCK 3: ADSL BROADBAND INTERNET AND WI-FI USB DONGLES****UNIT 1 ADSL BROADBAND INTERNET**

Introduction to ADSL broadband technology, motivation for ADSL Broadband, PSTN Basics, ADSL Modem basic architecture, working, standards, ADSL Wi-Fi Modem and Router, configuring a wired ADSL Modem for Internet Access, configuring a Wi-Fi ADSL modem/Router for Internet Access

**UNIT 2 WI-FI USB DONGLES**

Motivation and Need for Wi-Fi Dongles, basic architecture and working, connecting and configuring a Wi-Fi Dongle with a PC.

**BLOCK 4: INTERNET ACCESSING AND APPLICATION****UNIT 1 TETHERING FOR INTERNET ACCESS**

Need and Motivation for Tethering, Tethering with Wi-Fi, Tethering with Bluetooth, Tethering with USB Cable, Reverse Tethering

**UNIT 2 INTERNET/LAN APPLICATIONS**

Popular Browsers like Internet Explorer and Chrome, their configuration and settings, FileZilla File Transfer software, Team Viewer, Remote Desktop, Telnet, Microsoft Outlook Express.



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA 201

# Fundamentals of Computer Networking (FCN)

---

## BLOCK 1: NETWORKING CONCEPT

---

### UNIT 1

Introduction and Networking Basics

02

### UNIT 2

Network Interface Devices

11

---

# **BLOCK 1: NETWORKING CONCEPT**

---

## **Block Introduction**

Internet is a collection of computers where many computers grouped together share their information. Protocols are the set of rules that helps in communicating and controlling across a network. They run behind each service and hence for each internet service there is a specific protocol. By using certain applications, there are lots of services that are present on the internet.

In this block we will study and learn about networking and its devices. The concept related to various networking topologies are well detailed for future use. The mechanism of working and features of switches and routers with different types of networking characteristics are explained to you to gather knowledge about external networking devices.

The block will help readers with the basic understanding of how computers can be connected in a network and perform task. After reading this block you will able to connect to web with the use of modem and various OSI layer. The concept of different types of networking topologies with their arrangements will allow students to know more about different types of arrangements of computers as seen in daily life.

## **Block Objective**

**After learning this block, you will be able to understand:**

- The basic of Networking.
- Features of computer networking.
- Basic of WAN, LAN and PAN.
- Idea about Topologies.
- Familiarization about Network Adaptor Cards.

## **Block Structure**

**Unit 1: Introduction and Networking Basics**

**Unit 2: Network Interface Devices**



---

# **UNIT 1: INTRODUCTION AND NETWORKING BASICS**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Advantages of computer networking**
- 1.3 Computer networks and the Internet**
- 1.4 WAN, LAN and PAN basics**
- 1.5 Topologies**
- 1.6 Connecting Media: Wired and Wireless and their characteristics**
- 1.7 Introduction to NIDs and their specifications**
- 1.8 Let Us Sum Up**
- 1.9 Answers for Check Your Progress**
- 1.10 Glossary**
- 1.11 Assignment**
- 1.12 Activities**
- 1.13 Case Study**
- 1.14 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About computer network
- About Internet
- About Topologies
- About connecting media

---

## 1.1 Introduction

---

Networking involves association among two or more computers. The two computers will be connected across the world with the help of web and networking. There are two forms of modem one is with wires that's connected inside the computer system and other is wireless, that are more comfortable and accessible today. There are certain optical storage devices like CD and DVDs where information will be stored from 10 MB to 4.6 GB.

---

## 1.2 Advantages of computer networking

---

Basically, Networking is a connection between two or more computers. The main purpose of a network is to share the information among different users. Figure 1.1 shows the networking of two computers:

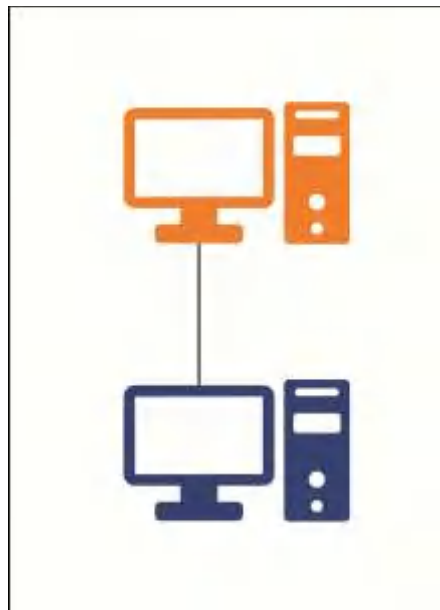


**Fig 1.1 Computer in network**

Computer network consist of the following:

- Two or multiple computers that can be a Server or a Client.
- A Network Interface Card (NIC).
- Connection medium that can either have wires or no wires.
- Network Operating system like MS Windows, NT or MS 2000, Novell NetWare, UNIX and Linux.

Internet is a setup of computers across the globe. Every computer that is connected to the internet is considered as a part of that network. Fig 1.2 shows the arrangement of computers in a network.

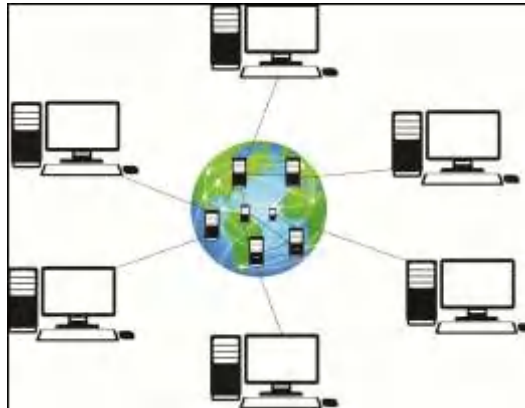


**Fig 1.2 Computers connected by Internet**

In order to share the information among people quickly and easily, we use the Internet. Internet is a collection of computers where many computers grouped together to share their information. In this case the information can be sent by the Sender and the Receiver receives that information. In this chapter we will study about Internet and it's working.

### **Advantages of Computer Network**

- Resource sharing
- Remote login ( Access to remote data base)
- E-Mailing ( person-to person communication)
- Entertainment
- Internet services
- Video conferencing
- Exchange of messages
- Sharing information at Low Cost
- Storing Files in server allows data to be shared easily
- Fast and Quick backing up of Files
- Software and resources can be easily managed.
- Network software have fast installation

- Devices can be shared easily
- Accessing files from any workstation

### Check your progress 1

1. What are the advantages of computer networks?
  - a. Resource sharing
  - b. Internet services
  - c. File storage
  - d. All of these

## 1.3 Computer networks and the Internet

On internet you'll be able to do chatting and exchange of information with many services offered by it. As internet is collection of computers where several computers grouped together share their data, the results of such sharing will led to spread of virus on host computer, that the user download any infectious attachment send by someone.

Network is usually the connection between the Sender and also the Receiver

The figure 1.3 shows two people sitting in a network.

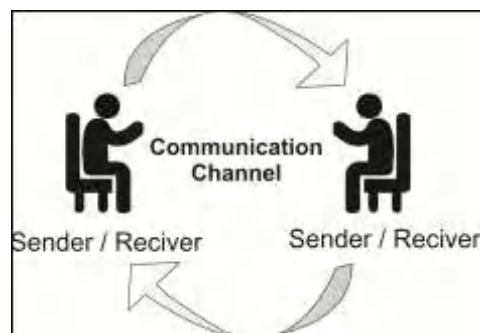


Fig 1.3 Networks

The general network comprises of:

- Sender
- Communication Channel Medium
- Receiver

A computer network is an interconnection of two or more computer systems located at the same or different places. It is a network that can connect two computers as shown in fig 1.4.

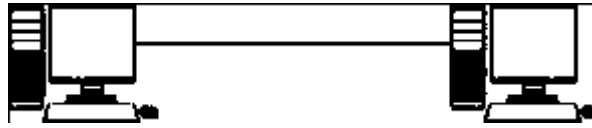


**Fig 1.4 Computers in network**

A computer network is a collection of two or more connected computers. When these computers are joined in a network, people can share files and also share the peripheral devices such as modems, printers, tape backup drives, or CD-ROM drives as shown in figure 1.5.

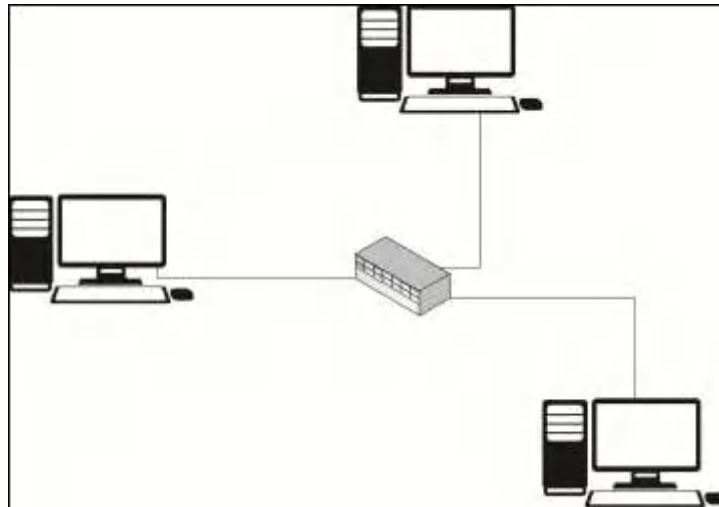


**Fig 1.5 Network of computers**

Computer network consist of:

- Two or more computers Server or Client workstation.
- Networking Interface Card's (NIC)
- A connection medium i.e. wires or wireless.
- Network Operating system software, such as Microsoft Windows NT or 2000, Novell NetWare, UNIX and Linux.

## Check your progress 2

1. Internet is:
  - a. Network of Computers
  - b. Connecting Single Computer in network
  - c. Connecting different Computers in network
  - b. All of these

---

## 1.4 WAN, LAN and PAN

---

Networking is a connection between two or more computers. The purpose of network is to share the information among different users. If more than two computers are to be connected in a network, this requires a HUB or a PORT as shown in fig 1.6.

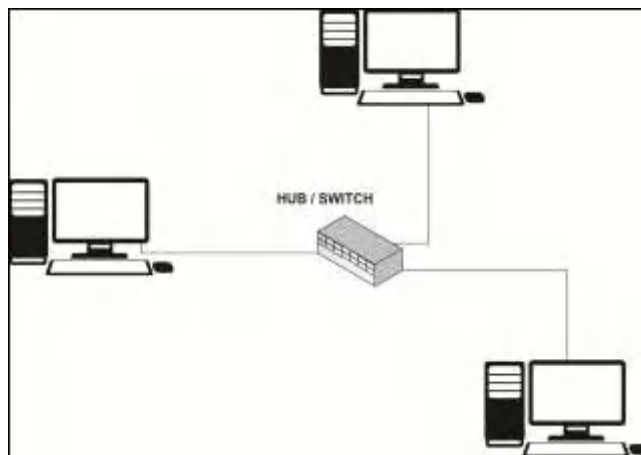


**Fig 1.6 Network with HUB**

Some of the common networks are:

LAN - Local Area Network

WLAN - Wireless Local Area Network

WAN - Wide Area Network

MAN - Metropolitan Area Network

SAN - Storage Area Network, System Area Network, Server Area Network,  
or sometimes Small Area Network

CAN - Campus Area Network, Controller Area Network, or sometimes  
Cluster Area Network

PAN - Personal Area Network

DAN - Desk Area Network

LAN and WAN are the original categories of area networks. The other networks have actually emerged over many years out of technology evolution.

## LAN

It is a typical network which is named as local area network or LAN. This network consists of group of computers along with its connecting devices that has a common communications channel. In this there will be only one main computer with which the rest of the computers are connected and they also shared a common processor speed as shown in fig 1.7.

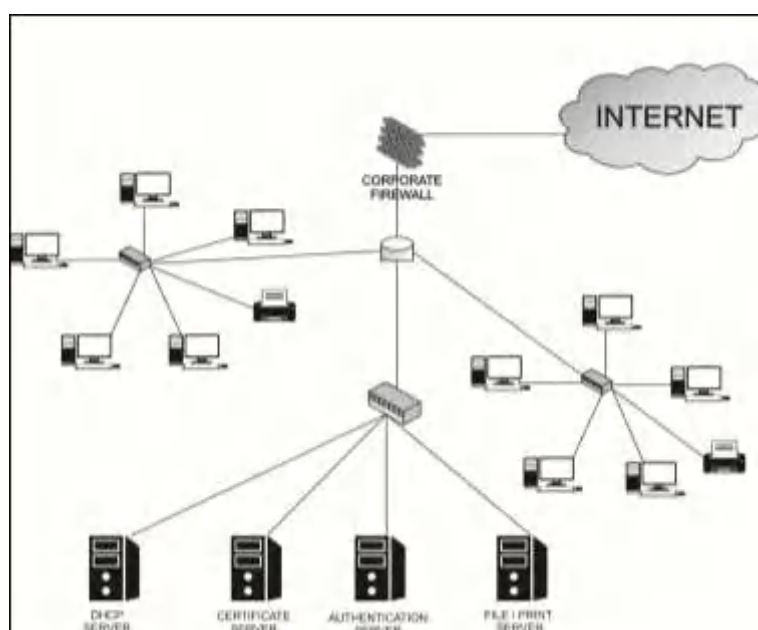


**Fig 1.7 LAN Network**

It is normally installed and available in an office building, school and university. In LAN Network, the server contains an applications and data storage that are commonly shared by many computer users. Such network serves as few as two or three users up to many thousands of users. A LAN server may also be used as a web server provided it is safely handled and precautions are carried to safe its internal applications and data from outside access.

## WAN

WAN is a network that connects users across larger distance. It is mainly used to connect across cities, states, or countries. The figure 1.8 shows the arrangement of WAN across the globe.

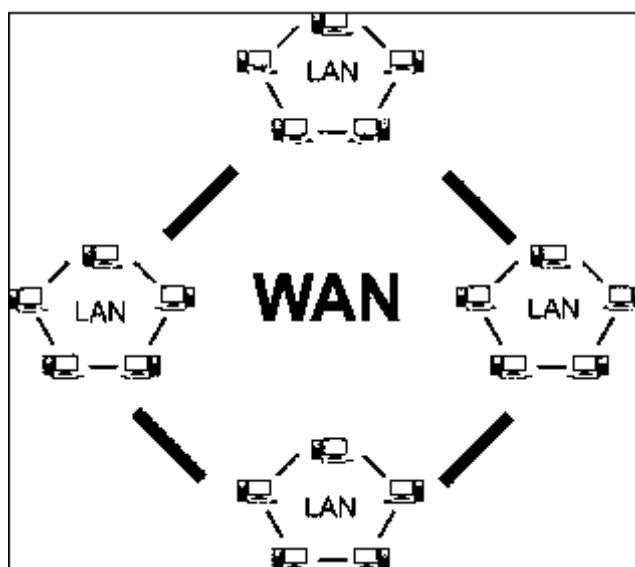


**Fig 1.8 Computer in WAN**

WANs normally use public telephone network & satellite links for data transmission. Data transmission rates are below 1 Mbps for WAN. It is normally owned by multiple organizations. The transmission time is more for WAN because of longer distances & different transmission medium used.

### **Different Types of Area Networks:**

Apart from LAN and WAN, there are many other computer networks such as:

**MAN or Metropolitan Area Network:** It is a network that uses much larger area as compared to LAN but smaller than WAN. It is a computer network that is owned and operated by an individual.

**CAN or Campus Area Network:** It is a network which spreads in area which covers multiple LANs and covers lesser area as compared to MAN.

**SAN or Storage Area Network:** It is a network that uses fibre optics channel for communication and connects servers to data storage devices through such technology.

**SAN or System Area Network:** It is another type of network that links with high performance computers having high speed connections in a zigzag configuration. It is also known as Cluster Area Network.



### Check your progress 3

1. Which network is used to connect people globally?
  - a. Local Area Network
  - b. Wide Area Network.
  - c. Metropolitan Area Network.
  - d. None of above.

## 1.5 Topologies

Another way to classify computer networks is based on the underlying topology used for constructing the networks. Topology is defined as the geometrical arrangement of nodes. Nodes are the various computer resources and communication devices.

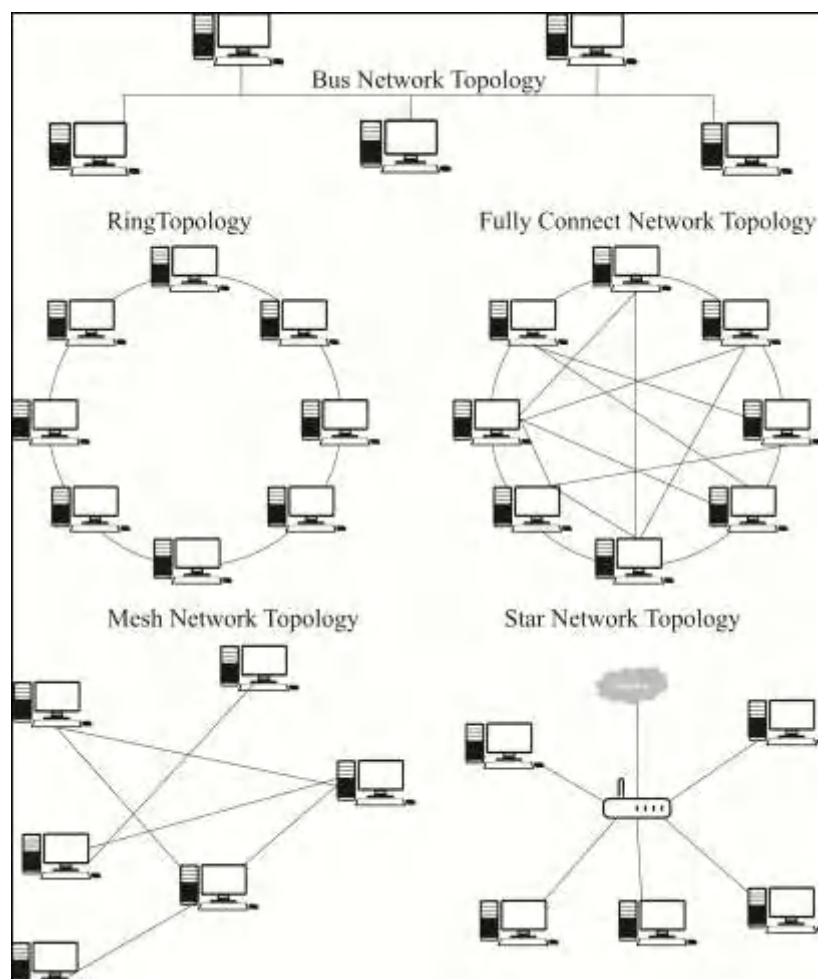


**Fig 1.9 Network Topology**

## Different Types of Topologies

Following are the different classes of network based on the topological structure.

- Bus Network
- Star Network
- Ring Network
- Mesh Network
- Tree Network

**Bus Network:** in a bus network, all nodes are connected to one line known as bus. it is conjointly referred as a time-shared bus. The bus permits just one pair of nodes to establish communication at a time. This property restricts the total number of nodes connected to form a reliable bus network. However, several protocols were developed for a bus to form communication more efficient and reliable. CSMA/CD and Token bus protocols are sensible examples. The structure of a bus network is shown in Figure 1.10.

Advantage of a bus network is its ability to connect any number of nodes without extensive hardware. Nodes can also be removed from the bus simply. It's straightforward to maintain the bus network.

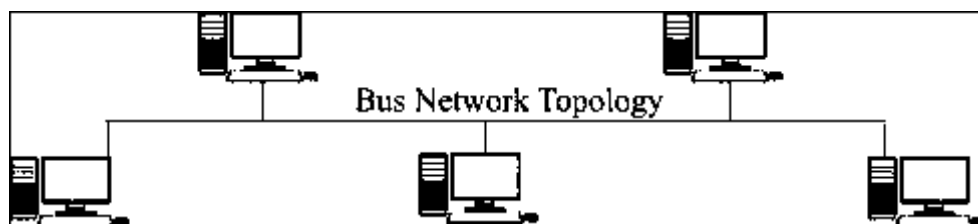


**Fig. 1.10 Bus network**

**Star Network:** In a star network, each node is connected by means of a dedicated point-to-point(P2P) channel to a central node called server that will act as a switch. The central server will provide the connectivity for all pair of nodes willing to communicate with each other. But, if the central server fails, the whole network will also fail. The transmission media may be a twisted pair, coaxial cable or optical fibre. Structure of a star network is shown in Figure 1.11.

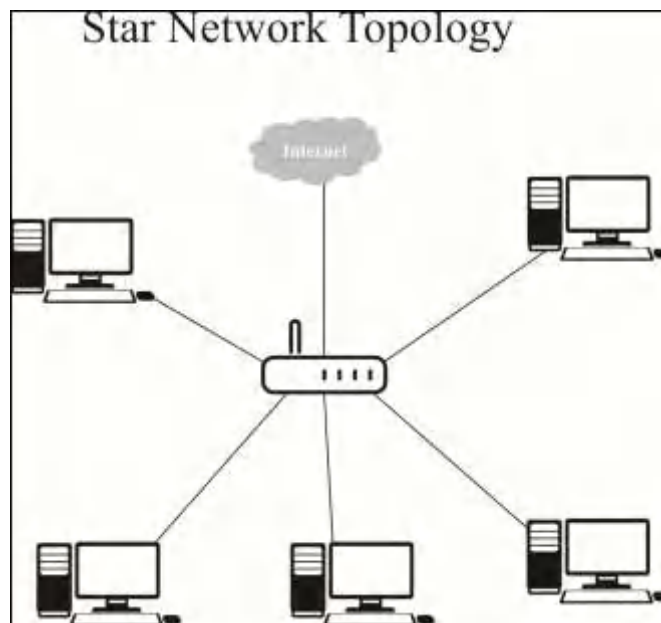


**Fig. 1.11 Star network**

Some of the advantages of star network are:

- Easy implementation
- Centralized control
- Simple access protocols

The main disadvantage of star network is that they suffer from the problem of central node failure. They also require long cable length; each new device requires an exclusive cable. Campus PBXs are often implemented using star network topology.

**Ring Network:** Nodes in a ring network are connected in the type of a closed loop. One communication channel is commonly implemented to provide the connectivity. Data from the sending node circulates round the ring till it reaches the destination. a ring will be unidirectional or bi-directional. In a unidirectional ring, data moves in one direction solely. In a bi-directional ring, data can move in both directions, but moves in one direction at a time. Single node failure may paralyse the transmission of information to a set of nodes in a unidirectional ring, but messages will be sent to nodes in either side of the affected node. .

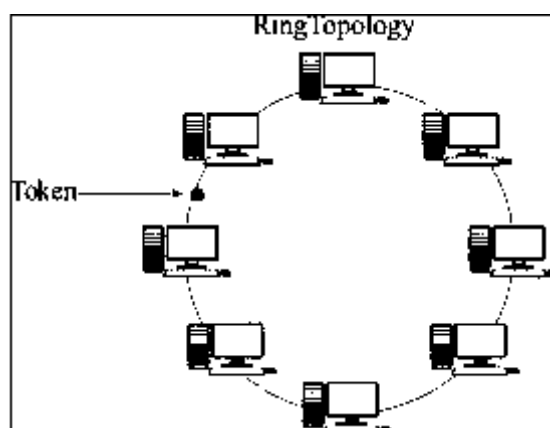


**Fig. 1.12 Ring network**

Ring network with a method called token passing (Token Ring) was proposed by IBM and approved by IEEE as one of the standards for LAN. Advantages of a ring network are its short cable length, suitability for optical fibre implementation and its flexibility to include new nodes which is also called as Network expansion. Disadvantages of ring networks include the failure of entire network in the presence of a single node failure, difficulty in diagnosing faults and its non-adaptability to structural changes.

**Mesh Network:** In a mesh network, each pair of nodes is connected by means of an exclusive point-to-point link. Each node requires a separate interface to connect with the other device. Mesh networks are seldom constructed in practice. They are useful in situations, where one node or station needs to frequently send messages to all other nodes. Otherwise, a considerable amount of network bandwidth got wasted. The advantages of mesh network are excessive amount of bandwidth and inherent fault-tolerance. The structure of a mesh network is shown in Figure 1.13.

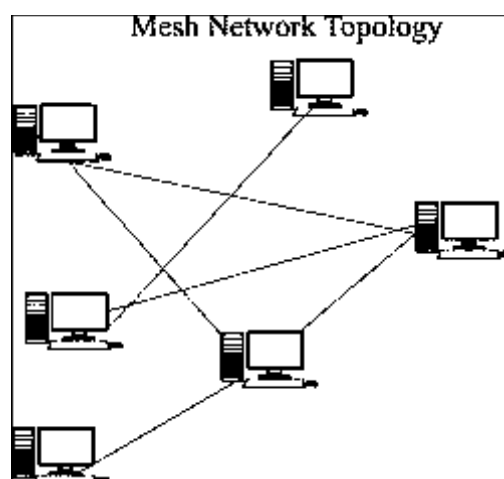


**Fig. 1.13 Mesh network**

**Tree Network:** A tree network is another form of bus network. Several nodes are connected into a hierarchical form as shown in diagram

The root node may be a powerful server or a mainframe computer often called a head-end. Tree networks are suitable for organizations, where head offices need to communicate with regional offices and regional offices need to communicate with remote offices. Advantages of a tree network are its ease of expansion, identification and isolation of faulty nodes whereas its disadvantage is that, it also suffers from the problem of the network being highly dependent on the root node.

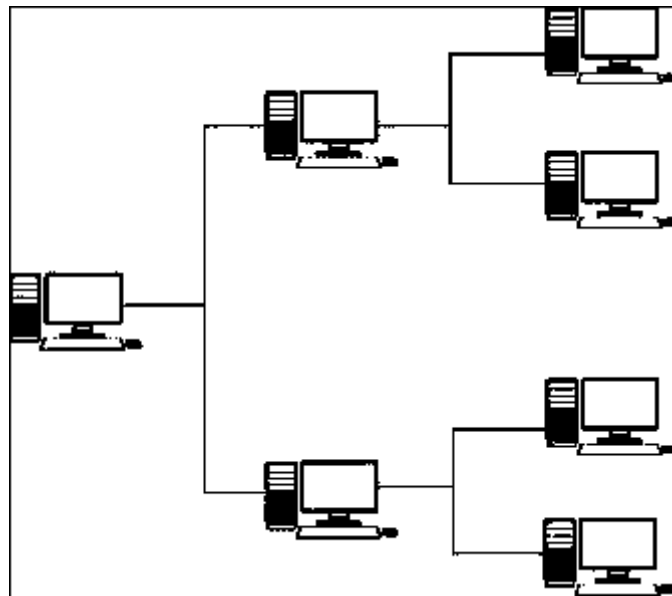


**Fig. 1.14 Tree network**

#### **Check your progress 4**

1. Which is not a network topology?

- a. Bus topology
- b. Star topology
- c. Brush topology
- d. Ring topology

---

## 1.6 Connecting Media: Wired and Wireless and their characteristics

---

Nowadays, internet users wish to enjoy accessing the internet even when they are away from their home and office place. Wireless modem is a modem that sends or receives network signals without the use of cable connections. It can access the internet without using any wired connectors or cable. Because of being wireless, it is much faster, reliable and is less expensive.

Today, many companies are into manufacturing wireless modems. These are designed as per the user requirements and are not expensive. Some of the famous and common wireless modems which are available include connect cards, USB sticks, Wi-Fi devices and wireless routers.

### Connect Cards

It is the starting series of wireless modems which first appeared in two versions:

- PC data cards
- Connect cards

Such wireless modems are very small and compact. They are used to provide internet facility in laptops, personal computers or routers.

### USB Sticks

It is a type of stick that is connected with the wireless modem and gives good internet speed in desktops and laptops. The size of USB stick is same as the size of a pen drive. The stick fits correctly into the USB port available in either the desktop or laptop. These sticks are not plug\_and\_play because it requires certain installation drivers in order to work. Such type of USB sticks are very easy to carry and can work anywhere.

### Mobile Hotspots

Wireless modems serve as a portable internet hotspots. Internet products such as Novatel's Mi-Fi routers get a wireless broadband network and move the signal to a particular range in the same way as Wi-Fi hotspots. It will make the respective devices with Wi-Fi features along with wireless broadband network that can be of much use to the user who use Computers, Smartphone and tablets while moving here and there.

## Wireless Routers

These are routers which are especially designed for home users as there is no need of a transmission cable. This router works without the internet cable and can receive or send the signal faster than a normal router.

### Check your progress 5

1. Wireless modem requires \_\_\_\_\_.
  - a. Cable
  - b. A Sockets
  - c. Some Wires
  - d. No Wires

---

## 1.7 Introduction to NIDs and their specifications

---

NIDs are an efficient method of providing operational and capital savings to service providers. A NID is installed at the customer premise and provides a demarcation point between the service provider and customer's network. Network Interface Devices allow end-to-end Operations, Administration and Maintenance (OAM) functionality for the service provider.

While the operational savings of NIDs are often shown with their features and capabilities for remote troubleshooting, easy installation and service Level Agreement (SLA) monitoring to reduce SLA penalties, it's necessary for service providers to be aware of the additional revenue streams and services that can be achieved once employing a NID at the demarcation purpose.

An NID may also be known as a network interface unit (NIU), telephone network interface (TNI), system network interface (SNI), or telephone network box

### Check your progress 6

1. What do you understand by the term NID?
  - a. Network Interface Design
  - b. Network Identification Design
  - c. Node Interface Design
  - d. None of these

---

## 1.8 Let Us Sum Up

---

In this unit we have learnt that networking involves arrangement of 2 or more computers that are connected across the world with the help of web and networking. It is studied that a workstation model is a basic arrangement where system comprises of workstations which are high end personal computers spread across the building or campus and are joined or connected through high speed LAN

It is found that a computer network is a group of interconnected computers which may be classified as per wide variety of characteristics. It is noted that a personal area network (PAN) is a computer network used for communication among computer devices close to one person.

The Metropolitan Area Network is a network that connects two or more Local Area Networks or Campus Area Networks together but does not extend beyond the boundaries of the immediate town/city. It is noted that CAN network may be considered as MAN network which in general is limited to smaller area as compared to typical MAN.

---

## 1.9 Answers for Check Your Progress

---

### Check your progress 1

Answers: (1-d)

### Check your progress 2

Answers: (1-d)



**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-c)

**Check your progress 5**

**Answers:** (1-d)

**Check your progress 6**

**Answers:** (1-a)

---

## 1.10 Glossary

---

1. **Network** - It is a relationship between the Sender and the Receiver.
2. **Computer network** - It is an interconnection of two or more computer systems located at either same or different places.
3. **Networking** - It is a connection between two or more computers.
4. **Wireless Modem** - It is a modem that sends or receives network signals without the use of cable connections.

---

## 1.11 Assignment

---

Define LAN and WAN.

---

## 1.12 Activities

---

Can a Wireless modem be taken anywhere to connect to the internet? Study and comment

---

## 1.13 Case Study

---

Study the network topology of your college.

---

## 1.14 Further Readings

---

1. Basic of Internet by Er. Nishit Mathur
2. Internet and the World by Ahmed Ansari

---

## UNIT 2: NETWORK INTERFACE DEVICES

---

### Unit Structure

#### 2.0 Learning Objectives

#### 2.1 Introduction

#### 2.2 Network Adaptor Cards (both wired and wireless)

#### 2.3 Hubs

#### 2.4 Switches

#### 2.5 Routers

#### 2.6 Access Points (Wireless)

#### 2.7 Repeaters

#### 2.8 Let Us Sum Up

#### 2.9 Answer for Check Your Progress

#### 2.10 Glossary

#### 2.11 Assignment

#### 2.12 Activities

#### 2.13 Case Study

#### 2.14 Further Readings

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About Network Adaptor Cards
- About Hubs and Switches
- About Routers

---

### 2.1 Introduction

---

Networks are built by adding a network interface card (NIC) or other network adapter to computer and then connecting that adapter to the medium--a wire or radio frequency--over which the data flows. Depending on network

topology, there may also be a central hub or router to which each of the computers connects. If the hub also routes data between the local network and another network, it is then called a router.

---

## 2.2 Network Adaptor Cards (both wired and wireless)

---

In order to connect to a network, a computer must be equipped with a device called a network card. A network card, or a network adapter, also called a network interface card, or NIC, permits a computer to attach to the exterior. If you buy a computer from one of those popular stores or big companies on the web, most of their computers have a network card tested and ready. If you go to a store that sells or manufactures computers, you will ask them to install or make sure that the computer has a network card.

### Types of Network Adapters

A network adapter is a unit of computer hardware. Several types of hardware adapters exist:

- Many new computers contain integrated (built in) wireless network adapter chips
- A USB network adapter plus into a standard USB port to enable computer network connections (typically Wi-Fi or Ethernet)
- A wireless game adapter (sometimes called a "media adapter") connects to an Xbox or Play station game console or other home entertainment product, providing a bridge to Wi-Fi wireless capability.
- On older PCs, a PCI adapter (often called a NIC) was a type of add-in card installed inside a desktop personal computer. A variant PCI adapter called "PC Card" (also known as PCMCIA cards) inserted into the side of a notebook computer to provide similar capability.

### Wired Network Cards: External

We have mentioned that a network card could also be used or installed externally. This can be done using USB. Before using it, you can purchase it from a computer store or a web store as shown in fig 2.1.



**Fig 2.1 USB drive**

### **Wireless Network Cards**

Depending on your network or budget of customers, instead of using wired network cards, you can use wireless ones. Most laptops already have a built-in wireless card so you may not have to acquire one. Many new desktop computers now have built-in wireless capability as shown in fig 2.2.



**Fig 2.2 Wireless Network Card**

Overall, the physical installation of a wireless network card follows similar rules as that of a wired NIC. They usually come with simple to follow instructions but it may be a good idea to install the wireless network adapters once installing the wireless router. Also, it may be a good plan to purchase the network cards and the wireless router from the same manufacturer.

Most desktop computers come without a wireless network card. If you buy a computer from a store and if you wish to use wireless networking, you'll buy a wireless network card separately. As stated already, a wireless network card isn't particularly tough to install.

Besides the wireless network cards that can be installed within the computer, you'll use external cards. These are installed using a USB port known as USB adapter as shown in fig 2.3.



**Fig 2.3 USB Adapter**

These adapters, like most USB objects, are easy to connect and use. Like any other hardware parts, when you connect these, the computer detects them and helps you to get them ready for use.

Unlike desktop computers, nowadays mostly laptops come equipped with a wireless network card. This means that, after purchasing or acquiring a laptop, you should simply check whether it has a wireless adapter. Therefore, check its documentation properly.

### **Check your progress 1**

1. Network cards can be \_\_\_\_\_.
  - a. Wired
  - b. Wireless
  - c. Both of these
  - d. None of these

---

## **2.3 Hubs**

---

A hub is a rectangular box that is used as the central object on which the computers and other devices are connected. To make this possible, a hub is equipped with small holes called as ports. Fig 2.4 shows such type of hub:



**Fig 2.4 Hub**

It comes with 4 ports, depending on its type and is equipped with 4, 5, 12, or more ports. Fig 2.5 shows hub with 8 ports:



**Fig 2.5 8 Bit Hub**

When configuring 8 bit hub, you need to attach an RJ-45 cable from the network card of a computer to one port of the hub. In most cases for a home-based or a small business network, you will not need (or should not use) a hub.

Hubs are the simplest way to connect 2 or more computers, servers and peripherals to form a simple network. A hub receives signals from each machine through wired connections, and then broadcasts them to all the other connected machines. So if computer A sends out a signal, Computers B, C and D can all receive it, even though the signal was meant only for computer D.

Hubs are of 2 types:

- **Active Hub:** they are smarter than the passive hubs. They not only provide the path for the data signals in fact they regenerate, concentrate and strengthen the signals before sending them to their destinations. Active hubs also are termed as 'repeaters'.
- **Passive Hub:** they're more like point contact for the wires to inbuilt the physical network. They have nothing to do with modifying the signals.

**Check your progress 2**

1. Which is known as repeaters?
  - a. Active Hub
  - b. Passive Hub
  - c. Both of these
  - d. None of these

---

## 2.4 Switches

---

A switch is a network device that selects a path or circuit for sending a unit of data to its next destination. A switch may additionally include the function of the router, a device or program that can determine the route and specifically what adjacent network point the data should be sent to. In general, a switch could be a simpler and quicker mechanism than a router, which needs knowledge about the network and the way to work out the route.

A switch is effectively a higher-performance alternative to a hub. People tend to benefit from a switch over a hub if their home network has four or a lot of computers, or if they want to use their home network for applications that generate significant amounts of network traffic, like multiplayer games or heavy music file sharing. Technically speaking, hubs operate using a broadcast model and switches operate using a virtual circuit model.

Switches are capable of determining the destination of each individual traffic element (such as an LAN frame) and selectively forwarding data to the one computer that actually needs it. By generating less network traffic in delivering messages, a switch performs higher than a hub on busy networks.

When a signal enters a port of the switch, the switch looks at the destination address of the frame and internally establishes a logical connection with the port connected to the destination node. Other ports on the switch have no part within the connection. The result's that each port on the switch corresponds to an individual collision domain, and network congestion is avoided. Thus, if a 10-Mbps Ethernet switch has 10 ports, every port effectively gets the complete bandwidth of 10 Mbps-to the frame, the switch's port seems to provide a dedicated connection to the destination node. {Ethernet|local area network|LAN} switches



are capable of building multiple internal logical connections at the same time, while routers usually process packets on a first-come, first-served.

There are 2 main types of switches. Layer-2 switches operate at the data-link layer of the OSI model and are based on bridging technologies. They establish logical connections between ports based on mac addresses. Use layer-2 switches for segmenting your existing network into smaller collision domains to improve performance. Layer-3 switches operate at the layer 3 of the OSI model and are based on routing technologies. They establish logical connections between ports based on network addresses. Use these for connecting different networks into an internetwork. Layer-3 switches are typically known as routing switches or multilayer switches.

### **Check your progress 3**

1. Which of the following operates at data link layer of the OSI model?
  - a. Layer-3 switch
  - b. Layer-2 switch
  - c. Hub
  - d. None of these

---

## **2.5 Routers**

---

Routers are network layer devices and are notably known as Layer- 3 devices of the OSI Model. They process logical addressing information within the Network header of a packet like ip Addresses. Router is used to form larger complex networks by complex traffic routing. It's the ability to connect dissimilar LANs on the same protocol. It additionally has the ability to limit the flow of broadcasts. A router primarily comprises of a hardware device or a system of the computer that has more than one network interface and routing software.



**Fig 2.6 Router**

When a router receives the data, it determines the destination address by reading the header of the packet. Once the address is determined, it searches in its routing table to get know how to reach the destination so forwards the packet to the higher hop on the route. The hop may be the final destination or another router.

Routing tables play a very pivotal role in letting the router makes a decision. So a routing table has to be compelled to be updated and complete. The 2 ways through which a router will receive information are:

- **Static Routing:** In static routing, the routing information is fed into the routing tables manually. It doesn't solely become a time-taking task but gets prone to errors as well. The manual change is additionally needed just in case of statically configured routers when change within the topology of the network or within the layout takes place. So static routing is feasible for tinniest environments with minimum of one or two routers.
- **Dynamic Routing:** For larger environment dynamic routing proves to be the practical solution. The process involves use of peculiar routing protocols to hold communication. The purpose of these protocols is to enable the routers to transfer information about to other routers, so the other routers can build their own routing tables.

### Check your progress 4

1. Which of the following is true about routers?
  - a. Routers operates on network layer
  - b. Routing table is maintained b 2 ways
  - c. Routers have the ability to limit the flow of broadcasts.
  - d. All of these

---

## 2.6 Access Points (Wireless)

---

In a wireless local area network, an access point may be a station that transmits and receives data. An access point connects users to different users among the network and can also serve as the point of interconnection between the WLAN and a fixed wire network. Every access point can serve multiple users within a defined network area; as people move beyond the range of 1 access point, they're automatically handed over to the next one. A small WLAN might solely need a single access point; the number required will increase as a function of the number of network users and therefore the physical size of the network.

A wireless access point is largely a hub with no wires that uses radio signals to try to its talking. Each one will handle some number of clients, usually about 30, very similar to a hub. They come available} in lots of sizes and shapes and have many different feature sets: those you'd use at home are completely unsuited to be used in a campus environment (and vice versa).



Fig 2.7 Wireless Access Point

Wireless access points (APs or WAPs) are the special-purpose communication devices on wireless local area networks (WLANs). Access points act as a central transmitter and receiver of wireless radio signals. Mainstream wireless APs support Wi-Fi and are most commonly used to support public Internet hotspots and other business networks where larger buildings and spaces need wireless coverage.

### Check your progress 5

1. What are WAPs?
  - a. Software used for routing.
  - b. Layers used in communication
  - c. Communication device on WLAN
  - d. None of these

---

## 2.7 Repeaters

---

Repeaters are network device used to regenerate or replicate a signal. Repeaters are employed in transmission systems to regenerate analog or digital signals distorted by transmission loss. Analog repeaters frequently will solely amplify the signal while digital repeaters will reconstruct a signal to near its original quality.

In a data network, a repeater will relay messages between sub networks that use different protocols or cable types. Hubs will operate as repeaters by relaying messages to all connected computers. A repeater cannot do the intelligent routing performed by bridges and routers.

In a wireless communications system, a repeater consists of a radio receiver, an amplifier, a transmitter, an isolator, and 2 antennas. The transmitter produces a signal on a frequency that differs from the received signal. This so-called frequency offset is critical to prevent the strong transmitted signal from disabling the receiver. The isolator provides additional protection in this respect. A repeater, once strategically located on top of a high building or a mountain, will greatly enhance the performance of a wireless network by permitting communications over distances much greater than would be possible without it.

### Check your progress 6

1. Repeater comprises of \_\_\_\_\_?
  - a. Amplifier
  - b. Transmitter
  - c. Isolator
  - d. All of these

---

## 2.8 Let Us Sum Up

---

In this unit we have learnt that networks are built by adding network interface card to the computer and connecting adapter to the medium which can be wire or radio frequency over which the data flows. A network card allows a computer to attach to exterior. If you buy a computer from one of those popular stores or big companies on the web, most of their computers have a network card tested and ready.

Unlike desktop computers, mostly laptops are equipped with wireless network card. A hub is a rectangular box that is used as central object on which the computers and other devices are connected. A switch is a network device that selects a path or circuit for sending a unit of data to its next destination and include the function of router, a device or program that can determine the route and specifically what adjacent network point the data should be sent to.

Routers are network layer devices and are notably known as Layer- 3 devices of the OSI Model which process logical addressing information in Network header of a packet like IP Addresses. Repeaters are network device used to regenerate or replicate signals which are used in transmission systems to regenerate analog or digital signals distorted by transmission loss.

---

## 2.9 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –c)

**Check your progress 2**

**Answers:** (1 -a)

**Check your progress 3**

**Answers:** (1 –b)

**Check your progress 4**

**Answers:** (1 -d)

**Check your progress 5**

**Answers:** (1 –c)

**Check your progress 6**

**Answers:** (1 –d)

---

## 2.10 Glossary

---

1. **Network** - It is an arrangement of network interface card or network adapter to computer.
2. **Network card** - It is a type of adapter which allow computer to attach externally.
3. **Hub** - Rectangular box on which computers and other devices are connected and carry many ports.
4. **Switch** - A network device which sends unit of data to next destination and has the function of router.
5. **Router** - A network layer devices which process logical addressing information in Network header of packet.

---

## 2.11 Assignment

---

Explain the purpose of Wireless Routers?

---

## **2.12 Activities**

---

Explain the features of Network Switch?

---

## **2.13 Case Study**

---

Compile the information about the type of network used in University.

---

## **2.14 Further Readings**

---

1. Basic of Internet by Er. Nishit Mathur
2. Internet and the World by Ahmed Ansari

---

## **Block Summary**

---

In this block, you will understand about Networking and knowledge on various types of networks. The block gives an idea on architecture and distribution of various network topologies with study about various topology features. The examples related to concept of network card, network adapter and network interface card are also discussed.

In this block, you will understand about the basic of working of switches and routers with their networking features. The concept related to wireless local area network and different OSI layers are also detailed. You will be demonstrated practically about the working of wireless modem.



---

## Block Assignment

---

### Short Answer Questions

1. If some packets arrive at router A and didn't find its destination in the router table, what will be action of router?
2. What is the name of the technology that is used to connect devices without physical connection?
3. What is a network?
4. If one switch is server and other switches attached are client, will the server switch vlans will propagate to client switches?

### Long Answer Questions

1. Switch A have 45 vlans and revision number 10 and switch B have 19 vlans and revision number 109, which switch vlan will propagate to which switch?
2. If Switch A is in server mode and switch B is in transparent mode, will the switch B will update its vlan database when switch A changes its own?
3. Describe Routing and ways to maintain routing table?

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                          | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| <b>Presentation Quality</b>                    | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Language and Style</b>                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Illustration used (Diagram, tables etc)</b> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Conceptual Clarity</b>                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Check your progress Quest</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| <b>Feed back to CYP Question</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# OBJECT ORIENTED CONCEPTS

PGDCA-102

## BLOCK 1: INTRODUCTION TO PROGRAMMING WITH JAVA



**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**

# OBJECT ORIENTED CONCEPTS

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Tejashree Bhosale

Mr. Kiran Shinde

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)





## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



## **OBJECT ORIENTED CONCEPTS**

### **Contents**

---

#### **BLOCK 1: INTRODUCTION TO PROGRAMMING WITH JAVA**

##### **UNIT 1 PROBLEM SOLVING WITH COMPUTERS**

Need of programming, Algorithms, Flowcharts, Pseudo-Code, Programming Languages Overview, and Overview of Object Oriented Concepts, Skills check, Exercises

##### **UNIT 2 INTRODUCTION TO JAVA**

History, Overview of JVM & Bytecode, Java versions (J2me etc.), Java class library, JDK, Your First Java Application, Skills check, Exercises

##### **UNIT 2 BEGINNING WITH JAVA PROGRAMMING**

Java Keywords, Comments in Java, Variables and Assignments, Strings and Characters, Arithmetic Operators and Expressions, Type Conversion in Expressions, Type Conversion, Skills check, Exercises

---

#### **BLOCK 2: JAVA CONTROL STATEMENTS, OPERATORS AND CLASSES**

##### **UNIT 1 JAVA CONTROL STATEMENTS**

The if statement, The if-else statement, Blocks of Code, The for statement, Increment & Decrement operators, Backslash codes, Relational and Boolean logical operators, Ternary operators, Skills check, Exercises

##### **UNIT 2 MORE ABOUT CONTROL STATEMENTS AND OPERATORS**

Nested if statement, Variations of the for loop, The while loop, The do loop, Nested loops, The break statement, The continue statement, The switch statement, The bitwise operators, Skills check, Exercises



### **UNIT 3**

### **CREATING CLASSES**

The General form of a class, Creating Simple Classes, Adding Constructors, Constructor Overloading, The this keyword, Instance variables and methods, Static variables and methods, Local variables and its scope, Method overloading, Argument Passing, Wrapper Classes, System Class, Garbage Collection, Skills check, Exercises

## **BLOCK 3:**

### **INHERITANCE, INTERFACE, PACKAGES AND EXCEPTIONS IN JAVA**

### **UNIT 1**

### **INHERITANCE**

Overview of Re-usability, Subclasses, Inheritance and Variables, Method Overriding, Inheritance and Methods, Inheritance and Constructors, Class Modifiers, Variable Modifiers, Constructor Modifiers, Method Modifiers, Object and Class classes, Skills Check, Exercises

### **UNIT 2**

### **INTERFACES AND PACKAGES**

Interfaces, Interface References, Interface Inheritance, The instanceof Operator, Packages, The import statement, Access control and Packages, Skills Check, Exercises

### **UNIT 3**

### **EXCEPTIONS**

Overview of Exceptions, Exception Handling, Catch Block and searches, The throw statement, Exception and Error classes, The throws clause, Custom exceptions, Skills check, Exercises

## **BLOCK 4:**

### **JAVA CLASS LIBRARY, FILE HANDLING AND GUI**

### **UNIT 1**

### **INTRODUCTION TO JAVA CLASS LIBRARY**

Classes of java.util package, Classes of java.net package, Classes of java.lang package, Overview of Collection Framework, Skills check, Exercises



## **UNIT 2**

### **FILE HANDLING IN JAVA**

Overview of File Handling, File Class in Java, Using Character Stream Classes, Using Byte Stream Classes, Using Scanner and Console Classes in Java, Skills check, Exercises

## **UNIT 3**

### **BUILDING GRAPHICAL USER INTERFACE (SWING)**

Building applications using classes related to Graphical Interface, Creating Layouts in GUI, Using Events in GUI applications, Skills check, Exercises



Dr. Babasaheb  
Ambedkar  
Open University

**PGDCA-102**

# **OBJECT ORIENTED CONCEPTS**

---

## **BLOCK 1: INTRODUCTION TO PROGRAMMING WITH JAVA**

---

### **UNIT 1**

|                                |    |
|--------------------------------|----|
| PROBLEM SOLVING WITH COMPUTERS | 03 |
|--------------------------------|----|

### **UNIT 2**

|                      |    |
|----------------------|----|
| INTRODUCTION TO JAVA | 22 |
|----------------------|----|

### **UNIT 3**

|                                 |    |
|---------------------------------|----|
| BEGINNING WITH JAVA PROGRAMMING | 42 |
|---------------------------------|----|

---

# BLOCK 1: INTRODUCTION TO PROGRAMMING WITH JAVA

---

## Block Introduction

Programming languages make it easier to design and implement programming ideas, by ensuring that you don't have to learn binary, which is the series of on/off or 1/0 commands the computer uses to calculate everything. The JVM is an abstract instead of actual machine or processor. It specifies certain instructions set, register sets, stack, garbage heap as well as required method. The Java program is a set of instructions which a computer understand.

In this block we will learn and study about concept of programming with basic idea about writing instructions sets. The knowledge about algorithm with its systematic procedures will necessary concept of flowchart will be detailed for use to student. The detailed about Java history and evolution along with pseudo code and byte codes are well explained that will normally be used in any programming language.

After completing this block, students will be able to know more about Java programming and necessary codes involved with basic of operators and variables which can more often be applied anywhere in the program. After this block, students will get knowledge about various types of Java version present as of today.

## Block Objective

**After learning this block, you will be able to understand:**

- Basic about Java programming
- Concept of writing an algorithm
- Idea about Flowchart
- Understanding about pseudo code
- Knowledge about Java virtual machine
- Explanation about Java 2 Platform editions
- Introduction to class library in Java
- Idea about Java Comments

Introduction to  
Programming  
with Java

- Types of variables in Java
- Knowledge about Java Expressions

## **Block Structure**

**Unit 1: Problem Solving with Computers**

**Unit 2: Introduction to Java**

**Unit 3: Beginning with Java Programming**

---

# **UNIT 1: PROBLEM SOLVING WITH COMPUTERS**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Need of Programming**
- 1.3 Algorithms**
- 1.4 Flowcharts**
- 1.5 Pseudo-Code**
- 1.6 Programming Languages Overview**
- 1.7 Overview of Object Oriented Concepts**
- 1.8 Skills Check**
- 1.9 Exercises**
- 1.10 Let Us Sum Up**
- 1.11 Answers for Check Your Progress**
- 1.12 Glossary**
- 1.13 Assignment**
- 1.14 Activities**
- 1.15 Case Study**
- 1.16 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Study the need of programming
- Features of Algorithms
- Basic of Flowcharts
- Know about Pseudo-Code
- Overview of Programming Languages



- Overview of Object Oriented Concepts

---

## 1.1 Introduction

---

Programming is writing of instructions sets which will guide the computer how to do certain work. Doing work relates to reading list of names from a file which could be alphabetical and writing it back again to the file. As seen, it could be much more complex as it involves displaying of graphical user interface which could mainly mean for games and other game's logic. Since, it is analysed that Java is relatively a current object-oriented programming language which has nowadays achieved more popularity and is easy to apply. It is a general-purpose language that can be used for many types of programming tasks.

---

## 1.2 Need of Programming

---

With programming languages, the commands break into generalised and simplified form. The comparison of various programming languages involves physical tasks which normally not require certain natural talent or skill which could be like playing outdoor games or doing painting or singing. Simply, a programming needs care and art of writing certain particular scripts called as codes.

Nowadays, computers do not really think as these are mathematical machines which can perform calculations which can be required by certain programming languages to be used in various software programmes.

### Check your progress 1

1. Programming language involves:

- |             |             |
|-------------|-------------|
| a. computer | c. software |
| b. codes.   | d. all      |

---

## 1.3 Algorithms

---

An algorithm is a series of definite number of steps which is shown in particular order so as to generate the solution for required problems. In computing, algorithms are essential because they serve as the systematic procedures that computers require. A good algorithm is like using the right tool in the workshop. It does the job with the right amount of effort.

The needs for algorithms can be:

- Input
- Generates an output
- Unambiguous
- Generality
- Correctness
- Finiteness
- Efficiency

### Advantages of Algorithms:

There are following advantages for using an algorithm:

**Efficiency:** In many problems, process such as sorting is most commonly appears in computing. A good speed algorithm should be applied to solve such problems that should consider the time and cost involved.

**Abstraction:** The algorithms should carry certain concept involved in solving problems which purify into easy ones. Using concept, a problem can be solved by breaking into simpler levels.

**Reusability:** As seen many commonly known algorithms are generalizations of difficult ones and many complicated problems can be purify into simpler ones, an effective way of solving many simpler problems will allow us to find many difficult problems.

It is studied that an algorithm is a commonly defined computational process that carries set of instructions which contains some value or may be set of values in shape of input and generates set of values as output. The algorithm accepts data, manipulate the data and gives the desired values as output as shown in fig 1.1.



Fig 1.1 Algorithm process

There are certain properties of an algorithm.

1. **Algorithm should be exact** - It is seen that an algorithm should be specific and particularly be explained in such a way that there remains no doubt.
2. **Algorithm should terminate** - The scope of algorithm concerns with solution of certain problem that can be solved on execution. So there should be finite number of steps involved in an algorithm.
3. **Algorithm should be effective** - It is seen that an algorithm should be straight and clear that results in correct output of a problem.
4. **Algorithm should be general** - Normally an algorithm is able to solve every part of problem. It should stress on whats, and not hows, apart from details for program version.

### Check your progress 2

1. The comment in an Algorithm begins with\_\_\_\_\_.
  - a. /\*
  - b. /
  - c. \*/
  - d. //

---

## 1.4 Flowcharts

---

A Flowchart is a graphical representation of an algorithm or its process. In this, every step in the process is shown by symbol that gives short description of the process. The symbols in the flow chart are connected together with arrows which show the flow of process in the particular direction. It typically describes the flow of data in a process which shows operations/steps in shape of pictures format that is convenient to understand in textual format.

A flowchart shows the process of operations that works in sequence in order to solve a particular problem. To understand the process, the flowchart gives you the way out step by step. Since it is the pictorial or graphical representation of a process, the idea behind flow chart is to communicate how a process should work.

Basically, flowchart analyses design and manage a process or program in various fields. Fig 1.2 shows the generic flowchart.

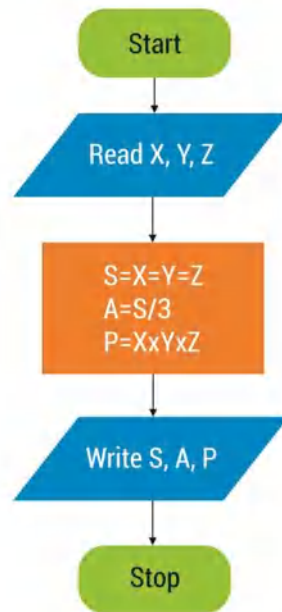


**Fig 1.2 Flowcharts**

It is studied that flowcharts are normally drawn in the beginning of making a computer solution. It serves as a communication tool among the programmers and people. With this, the programming of a particular problem is done that is useful in understanding complicated logic. Once the flowchart is drawn, it becomes easy for programmer to design a program in any high level language.

#### **Advantages:**

- It is a good way of communicating the logic of a system.
- It is helpful in analysing problems in an effective way.
- It serves as effective program documentation that is required for various jobs.
- It is a kind of blueprint of a particular program which is helpful in development phase.
- It is effective tool for debugging a process.
- It helps the programmer to show efforts efficiently on particular part of program.

### Flowchart Symbols:

To draw a flowchart of a particular program or process, you need some symbols to represent it. Some of the standard symbols that are required for drawing flowchart are:

**Terminator:** It is an oval shape symbol that shows start or end process.



**Process:** It is a rectangular shaped symbol which shows normal/generic process flow step.



**Decision:** This is of diamond shaped symbol which shows a branch in the process flow. It is applied when you need to have a decision which is in shape of Yes/No question or True/False.



**Connector:** This is of circular shaped symbol which shows a jump in the process flow.



**Data:** This is represented as parallelogram shape symbol which shows data input or output (I/O) for a process.



**Delay:** This is like D shaped symbols which shows a delay or wait in particular process for input to other process.



**Arrow:** The arrow shows direction of control flow in a process. Arrow coming from one symbol and entering into another symbol will shows the passing of control to the symbol.

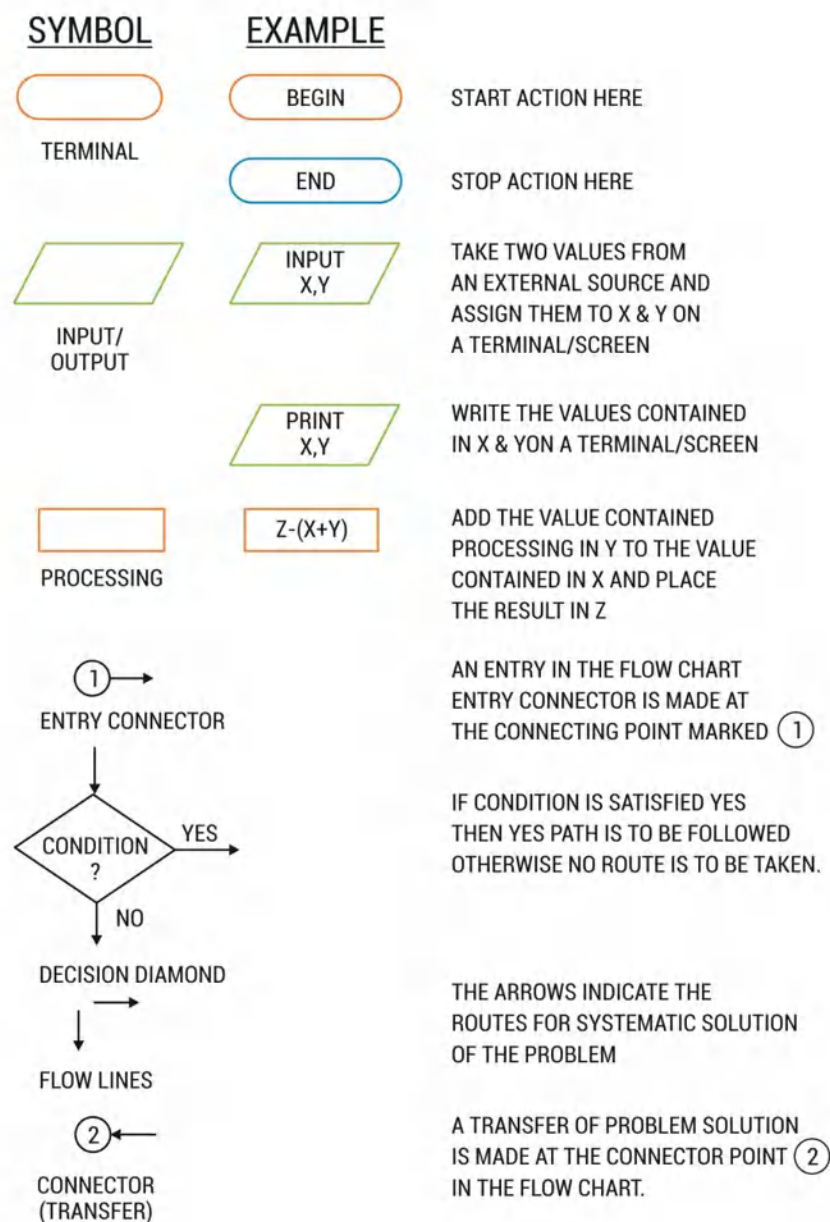


**Fig 1.3 Flowchart Symbol**

### Check your progress 3

1. In a flowchart, process can be represented by:
  - a. circular shaped object
  - b. rectangular shaped object
  - c. diamond shaped object
  - d. none of above

---

## 1.5 Pseudo-Code

---

Pseudo-code is a tool which is used for planning program logic. Pseudo-code is an artificial and informal language that helps programmers develop algorithms. Pseudo-code is a "text-based" detail (algorithmic) design tool.

To repeat an instruction, often it is required that all data should be processed in continuation in loop. Loop basically states for as a repetition or iteration of control structure. These codes are made up of following basic logic structures which are used for writing any computer program. These are:

1. Sequence
2. Selection (IF...THEN...ELSE or IF...THEN)
3. Iteration (DO...WHILE or REPEAT...UNTIL)

### Sequence

It is a logic applied for performing instructions that will work one after another in a particular series. So, in sequence logic, pseudo-code instructions are developed in order or sequence where they are used. Logically, the flow of pseudo-code is from top to bottom. Figure 1.4 shows an example of sequence logic structure.

## Selection

11



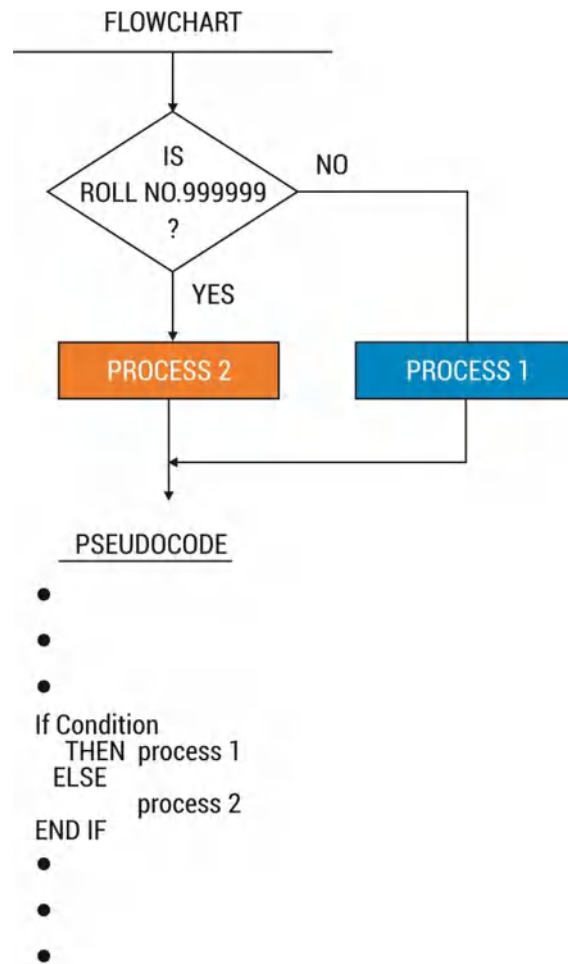


**Fig 1.5 pseudo-code for If-Then-Else structure**

In fig 1.5, the IF...THEN structure shows that when the condition is true, then process 1 will work and if this condition is not true, then the process will skip. As seen, the process 1 and process 2 can be one or more processes.

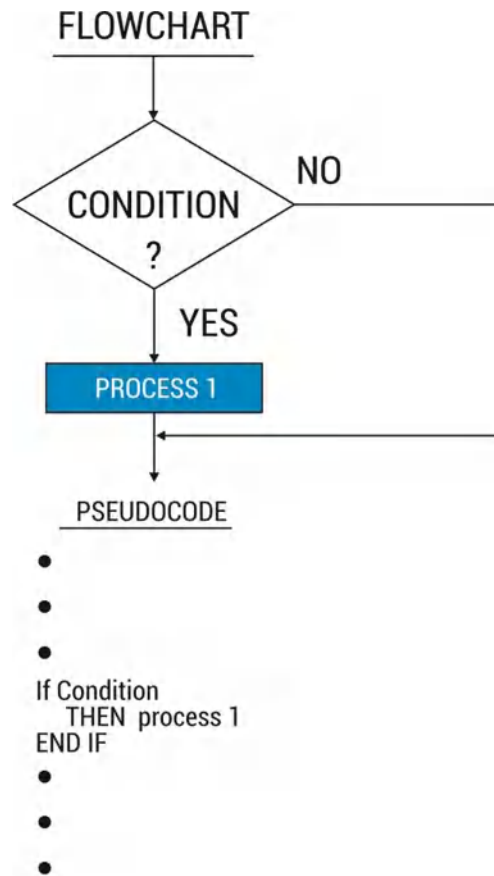


Fig 1.6 pseudo-code for If-Then selection structure

### Check your progress 4

1. The purpose of Pseudo-code is to:

- |                           |                       |
|---------------------------|-----------------------|
| a. plan for program logic | c. prepare a software |
| b. design logic program   | d. all of above       |

## 1.6 Programming Languages Overview

It is a type of computer generated language which is used by language programmers to develop applications, scripts or set of instructions so that a computer can carry out. Out of all the most basic language is machine language or a low level language which uses binary numbers 1 and 0 which a computer can easily understand and run without the use of translator or interpreter. There are many high-level languages such as Basic, C and Java which are simple and are like English which requires a compiler or an interpreter to change high-level code into the machine code. Because of this conversion, these languages are slower.

There are varieties of programming languages which are also called as computer language.

### Check your progress 5

1. In programming language, the basic language is:

- |                       |                      |
|-----------------------|----------------------|
| a. machine language   | c. both a and b      |
| b. low level language | d. none of the above |

---

## 1.7 Overview of Object Oriented Concepts

---

Object-oriented technology is very wide and extensive. The users of computer systems found the effects of such technology as increasingly easy-to-use software applications and operating systems which are flexible that is catered by several industries such as banking, telecommunications and television. In case of software technology, such object-oriented technology will cover object-oriented management of projects, computer hardware and computer aided software engineering, among others. There are certain concepts that laid the Object-oriented technology to spread its scope which can be:

### Concept: Object behaviour

In earlier programming, we have:

- Data which is inactive
- Functions which will handle any sort of data

It is seen that an object:

- Carries both data and methods which will handle every data
- Active and not passive which does things
- Charges for its own data
- Exploits data to other objects

### Concept: Object State

It is seen that in an object state:

- Both data and methods helps to control data
- Data has definite state of object

- Shows relationships among several objects

### **Concept: Object Classes**

It found that in an object class:

- Every object shows a class
- Object has field or variables
- Field is shown by class
- Object carries methods
- Class represents certainty of such methods
- Class works as template or cookie cutter

### **Concept: Classes as Abstract Data Types**

In an Abstract Data Type:

- Data is shown by object or thing
- Operations are done on data

### **Concept: Classes form a hierarchy**

In case of classes as hierarchy:

- Classes presents as tree like structure
- Root class is known as object
- Every class excluding object is a superclass
- Carries old object
- To define a class, you need to specify as superclass
- If superclass is not defined, it means object
- Every class has one or more subclasses

### **Concept: Objects from Superclasses**

A class can be called as fields or methods

Objects of such class contains fields and methods

Object contains:

- Fields which are in class's superclasses
- Methods which are in class's superclasses

It is not a detailed for an object.

**Concept: Created Objects**

It is seen that an n integer will performs:

- Declaration of n integer variable
- Allocation space keeping values for n

**Concept: Variables to hold subclass objects**

If B is a subclass of A, then:

- A objects will allocate to A variables
- B objects will allocate to B variables

B objects will allocate to A variable, but A objects cannot be allocated to B variable.

All B objects is A but not all A object is B

So bVariable = (B) aObject;

**Concept: Constructors make objects**

In this:

- Every class has a constructor which generates an object
- Keyword new applies to call constructor as secretary = new Employee ( )
- Writing of a constructor if allowed

Java gives default constructor having no arguments by setting up of new object to 0 and if this works out, then a constructor cannot be written, so syntax for writing constructors will remain similar for writing methods.

**Check your progress 6**

1. The object oriented technology includes:
  - a. object oriented management of projects
  - b. computer hardware
  - c. computer aided software engineering
  - d. all of above

---

## 1.8 Skills Check

---

Programming skills will allow someone to test proficiency level that exists in different programming languages. There are certain programming languages such as C, Java and .Net which allows various skills to test. To check your skills you need to work on certain small programmes that can be evaluated and test with related softwares. To know about better skills the need for good programming skills are essential.

### Check your progress 7

1. Programming skills can be checked:

- |                           |                           |
|---------------------------|---------------------------|
| a. by writing a programme | c. by working on software |
| b. by testing a program   | d. all of these           |

---

## 1.9 Exercises

---

### Example 1:

Write an algorithm and flowchart for computing the average number of the default 4 numbers.

Algorithm:

Start-

-----

Input A,B,C,D

$AVERAGE = (A + B + C + D) / 4$

Print AVERAGE

-----

End

CLS

CLEAR

INPUT a, b, c, d

$avr = (a + b + c + d) / 4$

```
PRINT avr  
END
```

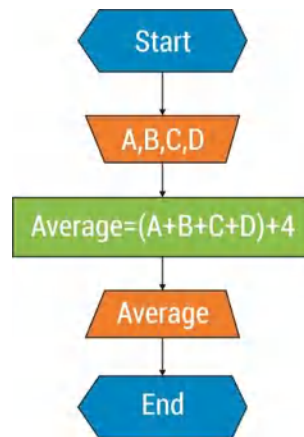


Fig 1.7 Example Flowchart

### Check your progress 8

1. What will be output of the program?

```
int x = 1, y = 6  
while (y--)  
{  
    x++;  
}  
System.out.println("x="+x"y="+y)
```

a. x = 1 y = 0  
b. x = -7 y = 6  
c. x = 3 y = -4  
d. compilation error

---

## 1.10 Let Us Sum Up

---

In this unit we have learnt that a programming is a sort of writing instructions sets that guides computer how to do certain work

It is studied that an algorithms are essential as they serve as systematic procedures which computers require.

It is seen that an algorithm carries certain concept which is involved in solving problems which purify into easy ones.

A Flowchart is a graphical representation of an algorithm or its process in which the diamond shaped symbol represents process flow.

It is found that pseudo-code is a tool which is used for planning program logic.

In case of software technology, such object-oriented technology will cover object-oriented management of projects, computer hardware and computer aided software engineering, among others.

---

## 1.11      **Answers for Check Your Progress**

---

**Check your progress 1**

**Answers:** (1-d)

**Check your progress 2**

**Answers:** (1-d)

**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-a)

**Check your progress 5**

**Answers:** (1-c)

**Check your progress 6**

**Answers:** (1-d)

**Check your progress 7**

**Answers:** (1-d)



**Check your progress 8**

**Answers:** (1-d)

---

## 1.12 Glossary

---

1. **Algorithm** - It is a series of fixed number of steps that are arranged in particular order.
2. **Programming** - It is writing of instructions sets that will guide computer how to do certain work.
3. **Efficiency** - It determines the speed at which an algorithm be able to solve problems.
4. **Abstraction** - It involves solving of certain program by breaking up into simpler levels.
5. **Pseudo-code** - It is a tool which is used for planning program logic.

---

## 1.13 Assignment

---

Write short note of necessity of programming?

---

## 1.14 Activities

---

Calculate the output of the following program?

```
int I = 0;

outer;

while(true)
{
    I++;

    Inner;

    For (int j = 0; j < 10; j++)
    {
        I += j;
    }
}
```

```
        If(j == 3)
            continue inner;
        break outer;
    }
    continue outer;
}
System.out.println(I);
```

---

## 1.15 Case Study

---

Compare and discuss the Java programming operators?

---

## 1.16 Further Readings

---

1. Learning Programming by Peter Norvig's
2. Approach programming with a more positive by P. Brian. Mackey

---

## **UNIT 2: INTRODUCTION TO JAVA**

---

### **Unit Structure**

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 History**
- 2.3 Overview of JVM and Bytecode**
- 2.4 Java Versions (J2me etc.)**
- 2.5 Java Class Library**
- 2.6 JDK**
- 2.7 Your First Java Application**
- 2.8 Skills Check**
- 2.9 Exercises**
- 2.10 Let Us Sum Up**
- 2.11 Answers for Check Your Progress**
- 2.12 Glossary**
- 2.13 Assignment**
- 2.14 Activities**
- 2.15 Case Study**
- 2.16 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Overview of JVM and Bytecode
- Basic of Java versions
- Study of Java class library
- Know-how of JDK
- First Java Application

---

## 2.1 Introduction

---

Java is a programming language created by James Gosling from Sun Microsystems in 1991. The first publicly available version of Java was released in 1995 as Java 1.0. This is commonly used programming language that has variety of uses and applications which makes them to use in desktop applications, Mobile Applications, Enterprise applications etc.

Java is:

- Class based and object oriented programming language
- Computing platform
- Fast, secure, and reliable
- Free
- General purpose
- Concurrent

The features of Java language are:

- It is a simple object oriented language and is quiet familiar.
- It is quiet protected and rough and tuff language.
- It is quiet portable and has varied architecture.
- It is quiet portable and has varied architecture.
- It works with high performance.
- It is interpreted, threaded and dynamic.

---

## 2.2 History

---

Java was initially framed by Sun Microsystems in the beginning of 1990s. The language was created with the idea to solve problem related to connectivity of several household machines together. As it was designed for special connectivity purpose only, so the particular project was failed as no one wants to use it. Initially the name of Java is OAK.

James Gosling worked initially on Java and was known as father of Java. The name OAK was renamed as Java in the year 1994. During the mid of year 1995 in month of May, Java was publicly released. Java was targeted for Internet

development. The release of applets was initially supported by big companies such as Netscape Communications. Some of the Java Versions are shown in table

**Versions of Java**

| Java Version | Release Date | Year |
|--------------|--------------|------|
| JDK 1.0      | January 21   | 1996 |
| JDK 1.1      | February 19  | 1997 |
| J2SE 1.2     | December 8   | 1998 |
| J2SE 1.3     | May 8        | 2000 |
| J2SE 1.4     | February 6   | 2002 |
| J2SE 5.0     | September 30 | 2004 |
| Java SE 6    | December 11  | 2006 |
| Java SE 7    | July 28      | 2011 |

### **Check your progress 1**

1. OAK was renamed as Java in year:

- a. 1995
- b. 1996
- c. 1997
- d. 1994

---

## **2.3 Overview of JVM and Bytecode**

---

### **JVM**

A JVM is called as Java virtual machine which interprets assembled Java binary code which is called as bytecode for computer's processor to do certain Java program instructions. The idea of inventing Java is only to allow application programs which possibly could work on any platform without the need of rewriting it or compiling again by programmer for individual platform. With the help of Java virtual machine, it is possible as it carries particular instruction lengths and related speciality of the platform.

Once the machines have been implemented for a particular platform, any type of Java program will be able to work on such base. Java virtual machine will either understand the bytecode per instruction at a time or by accumulating bytecode for actual processing with the help of just-in-time compiler.

It is studied that a Java virtual machine (JVM) as shown in fig 2.1 is a software implementation of computer which will carry out programs such as real machine. The Java virtual machine is designed particularly for particular operating system.

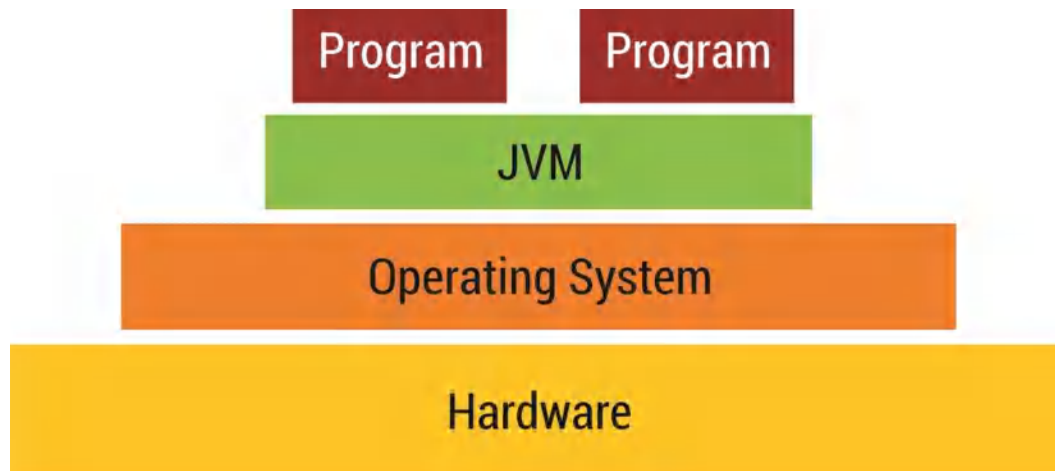


**Fig 2.1 Java Virtual Machine**

### **Bytecode**

It is a computer object code which is obtained by a program machine called as virtual machine instead of actual computer machine as the processor. Such type of machine will convert standard machine instruction into particular machine instruction or instructions which a computer processor is able to understand. The result of such compilation is bytecode. The bytecode is the result obtained after compilation of source code which is written in language that is able to handle by such machine. Many computer languages like C and C++, requires an individual compiler for every computer platform like for all computer operating system along with its hardware set of instructions.

It is seen that the best known language at present which uses bytecode and virtual machine approach is Java. Apart from this, LISP language applied in artificial intelligence applications was also an earlier language which was compiled by bytecode.

Instead of understanding single instruction at a particular time, Java bytecode will be able to recompile at every system platform with the use of just-

in-time compiler which will make the Java program to run faster. It is found that in Java, the bytecodes are present in binary file that carries .CLASS suffix.

### Check your progress 2

1. The full form of JVM is called as:

- |                         |                          |
|-------------------------|--------------------------|
| a. Java virtual machine | c. Java vertical machine |
| b. Java visual machine  | d. None of above         |

## 2.4 Java Versions (J2me etc.)

Java is formally called as Java 2 Platform which carries three editions:

- Java 2 Standard Edition (J2SE)
- Java 2 Enterprise Edition (J2EE)
- Java 2 Micro Edition (J2ME)

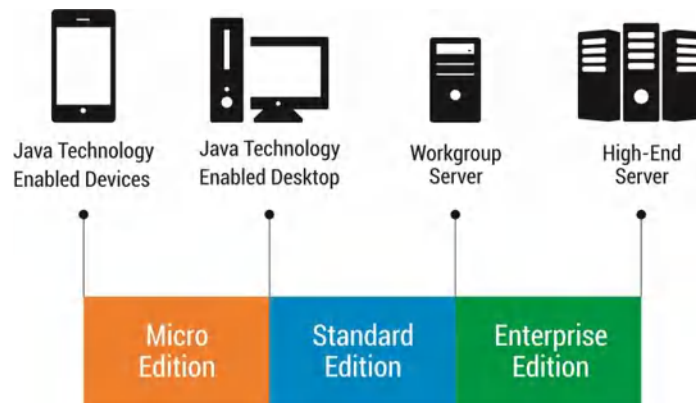


Fig 2.2 Java Editions

All these three editions will focus on different kinds of applications which run on different devices. It is found that:

- Desktop based applications were developed with the help of J2SE that carries necessary user interface classes.
- Server based applications was formed with the help of J2EE that stress more on component based programming as well as deployment.
- Handheld along with embedded devices are formed with the help of J2ME.

In the year 1995, the JDK 1.0 exists which was upgraded to JDK 1.1 and to Java 2 in the year 1999.

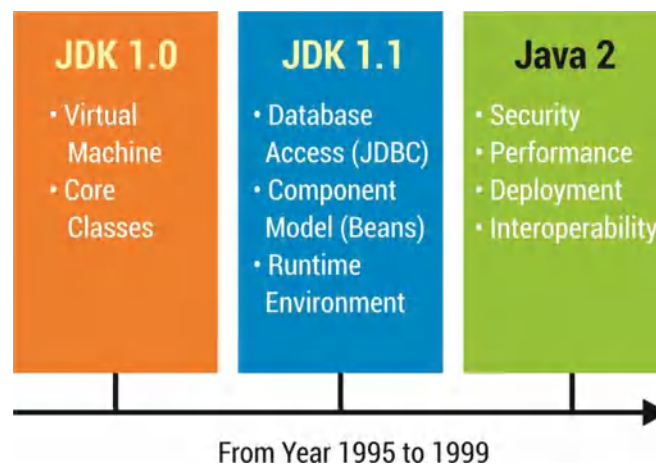


**Fig 2.3 JDK**

It is seen that there exists single Java platform with multiple profiles such as:

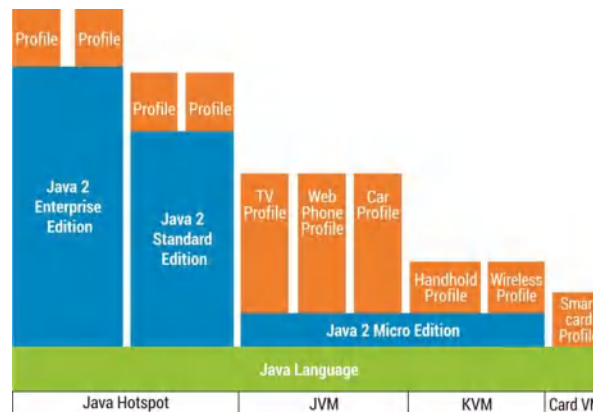


**Fig 2.4 Java Platforms**

## J2ME

The Java Platform Micro Edition which is also known as Java ME, is a platform which is designed for particular embedded systems. In this, the target devices such as industrial controls to mobile phones and set-top boxes are present. It is seen that Java ME was earlier called as Java 2 Platform which is Micro Edition as J2ME.

As studied, the edition of Java ME was initially invented by Sun Microsystems which was made advanced by Oracle Corporation who named it as Personal Java. Earlier, the different brands of Java ME have evolved in different JSRs. During the month of December 2006, Java ME source code was licensed under GNU which was released as phone ME.

## J2SE

Since Java is a dynamic programming languages used by computer programmers today, this language carries advance features with its current edition



on Java 2 Platform which is the standard edition called as J2SE. This edition is mainly used for writing applets and other applications.

The main advantage of J2SE edition is that it is used in development of certain Java applications that are utilised for single computers. The J2SE edition applets and several other applications allow such functions to run smoothly. In the absence of such applications, various transactions and several Internet interactions will not takes place? With this, the edition is of a great enabler of carrying web activity.

## J2EE

In order to lower the costs and fast track application design and development, Java 2 Platform Enterprise Edition called as J2EE shows a component based mechanism so as to construct, develop, assemble and deploy enterprise applications. Such platform uses multitier distributed application model. It is studied that application logic is framed into parts as per the function with certain application components which makes J2EE application to be kept on various machines according to the tier present in the multitier J2EE environment as per which the application belongs.

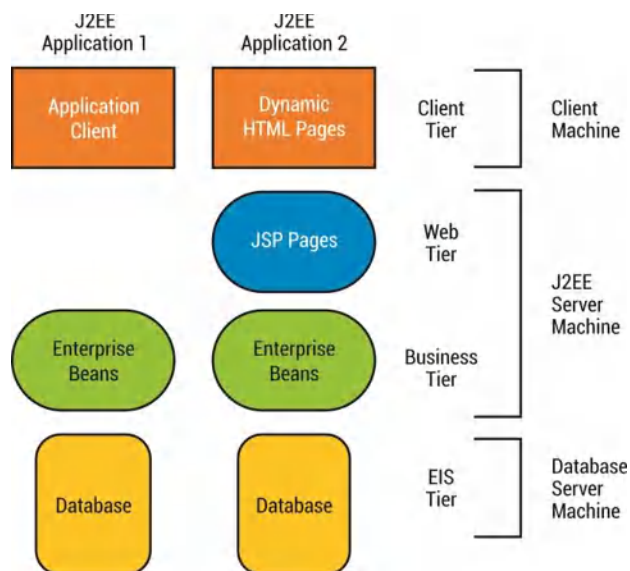


**Fig 2.5 J2EE Multitier Applications**

Figure 2.5 shows two multitier J2EE applications divided into the tiers described above. The parts shown are presented in J2EE Components as:

- Client-tier components run on the client machine.
- Web-tier components run on the J2EE server.
- Business-tier components run on the J2EE server.

- Enterprise information system (EIS)-tier software runs on the EIS server.

### Check your progress 3

1. Which is a multitier distributed application model?

- |         |                 |
|---------|-----------------|
| a. J2ME | c. J2EE         |
| b. J2SE | d. all of above |

---

## 2.5 Java Class Library

---

It is found that there are many classes available in the library in Java. A wide range of extensive library of pre-written classes which can be applied in certain programs are listed below which are grouped into Java 1.1 packages.

- |                            |                   |
|----------------------------|-------------------|
| • Java.applet              | java.awt          |
| • Java.awt.data transfer   | java.awt.event    |
| • Java.awt.image           | java.awt.peer     |
| • Java.beans               | java.io           |
| • Java.lang                | java.lang.reflect |
| • Java.math                | java.net          |
| • Java.rmi                 | java.rmi.dgc      |
| • Java.rmi.registry        | java.rmi.server   |
| • Java.security            | java.security.acl |
| • Java.security.interfaces | java.sql          |
| • Java.text                | java.util         |
| • Java.util.zip            |                   |

In this we find that every package defines a number of classes, interfaces, exceptions and errors. Also, the packages further gets splitted into sub-packages, as in case of java.lang package, which has sub-package as java.lang.reflect. It is seen that a class in sub package will not approach to a class located inside parent package. It is seen that java.net package having interfaces, classes as well as exceptions are shown below:

### Interfaces in java.net

- ContentHandlerFactory
- FileNameMap
- SocketImplFactory
- URLStreamHandlerFactory

### Classes in java.net

- |                     |                    |
|---------------------|--------------------|
| • ContentHandler    | DatagramPacket     |
| • DatagramSocket    | DatagramSocketImpl |
| • HttpURLConnection | InetAddress        |
| • MulticastSocket   | ServerSocket       |
| • Socket            | SocketImpl         |
| • URL               | URLConnection      |
| • URLEncoder        | URLStreamHandler   |

### Exceptions in java.net

- BindException
- ConnectException
- MalformedURLException
- NoRouteToHostException
- ProtocolException
- SocketException
- UnknownHostException
- UnknownServiceException

### Check your progress 4

1. In \_\_\_\_\_package, gc() method is present.

a. java.lang

c. java.awt

b. java.util

d. java.io

---

## 2.6 JDK

---

JDK also known as Java Development Kit is a program development environment which is basically employed for writing Java applets and applications. It comprises of runtime environment which remains on top of an operating system layer along with tools and programming which is required by the developer in order to compile, debug and run applets as well as applications that are written in Java language.

It is found that JDK carries JRE and Javac tools which will allow the development as well as execution of certain Java applications as well as applets. It carries lots of important useful tools like:

- **Applet viewer:** It runs and debugs Java applets without Web browser.
- **Javadoc:** It will extract comments in particular format from Java source files which will form an HTML file tree.
- **Jar:** It is a sort of archive tool which combines various files into single .jar archive file.
- **Jarsigner:** It is a Java application that will produce signatures for .jar files which will verify signatures of particular signed .jar files.
- **Javaws:** This is software that downloads and run Java applications from Web.
- **pack200:** It is another Java application which will transform .jar file to compressed pack200 file with the help of Java's gzip compressor.

### Check your progress 5

1. \_\_\_\_\_ will run and debug Java applets without the use of Web browser.

- |                  |              |
|------------------|--------------|
| a. Applet viewer | c. Jar       |
| b. Javadoc       | d. Jarsigner |

---

## 2.7 Your First Java Application

---

To write your First Java program, you have to first install Java Development Kit. After installation of JDK, you can create java program by using a class having main method as shown:

- Initially install JDK.

- Now set path of jdk/bin directory.
- Create java program
- Compile and run java program
- Finally the program will create first hello java example.

To create hello java program, write the code as:

```
Class simple{  
  
    Public static void main (String args []){  
  
        System.out.println("Hello Java");  
  
    }  
  
}
```

Save this file as Simple.java. You will find that when you run the above code, then the compiled output will show cannot open shared object file; No such file or directory.

To understand the first java program, you need to understand the meaning of class, public, static, void, main, String[], System.out.println().

- class: It is applied to declare a class in java.
- public: It is an access modifier that shows visibility to all.
- static: It declares certain statics method which is not required to form an object to invoke such method. It is done by JVM hence doesn't require an object to invoke further saves memory.
- void: It is a return type of method which doesn't returns any value.
- main: It shows startup of a program.
- String': It is applied for command line argument.
- System.out.printlnO: This command is applied for printing statement.

You can write a simple Java program in a notepad by selecting start menu---  
--> All Programs -----> Accessories -----> notepad as shown:

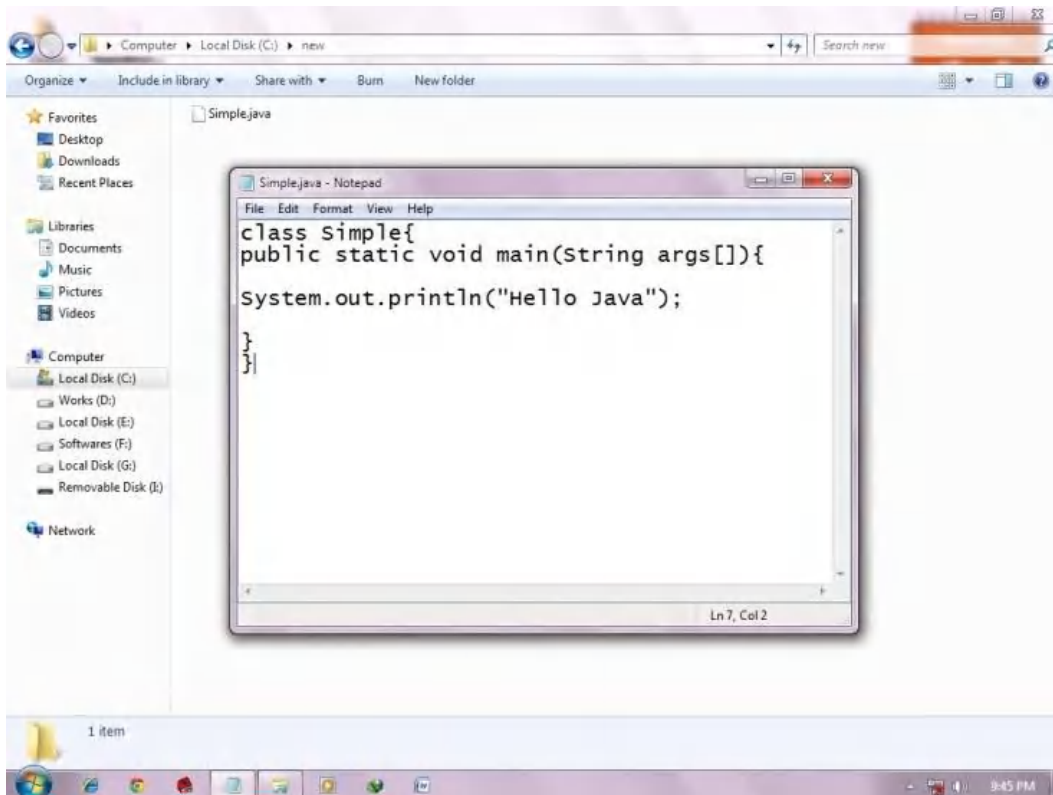


**Fig 2.6 Simple Java Program in Notepad**

Now in order to compile and run the above program, you need to open command prompt by clicking on start menu -----> Choosing All Programs -----> Selecting Accessories -----> command prompt.

When you compile a program, then the Java File converts Java code into bytecode as shown:

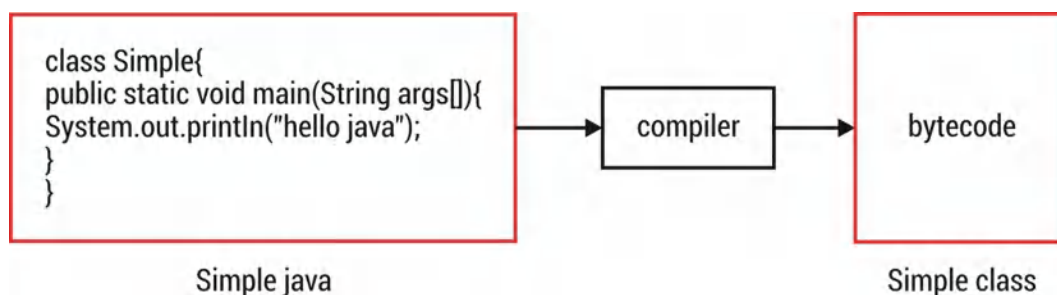


**Fig 2.7 Compilation Process**

Also after the compilation, the above program will run with a process as shown in flowchart below:

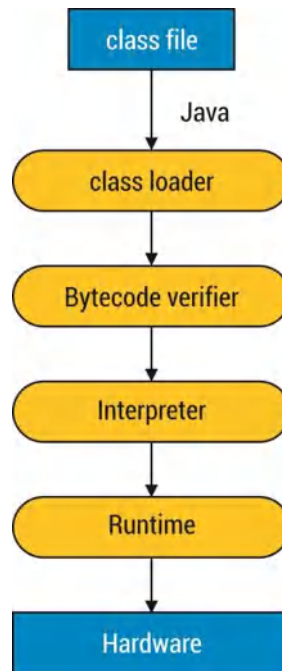


**Fig 2.8 Flowchart showing running process**

Figure 2.9 shows the compilation and running output of the program on notepad as shown below:

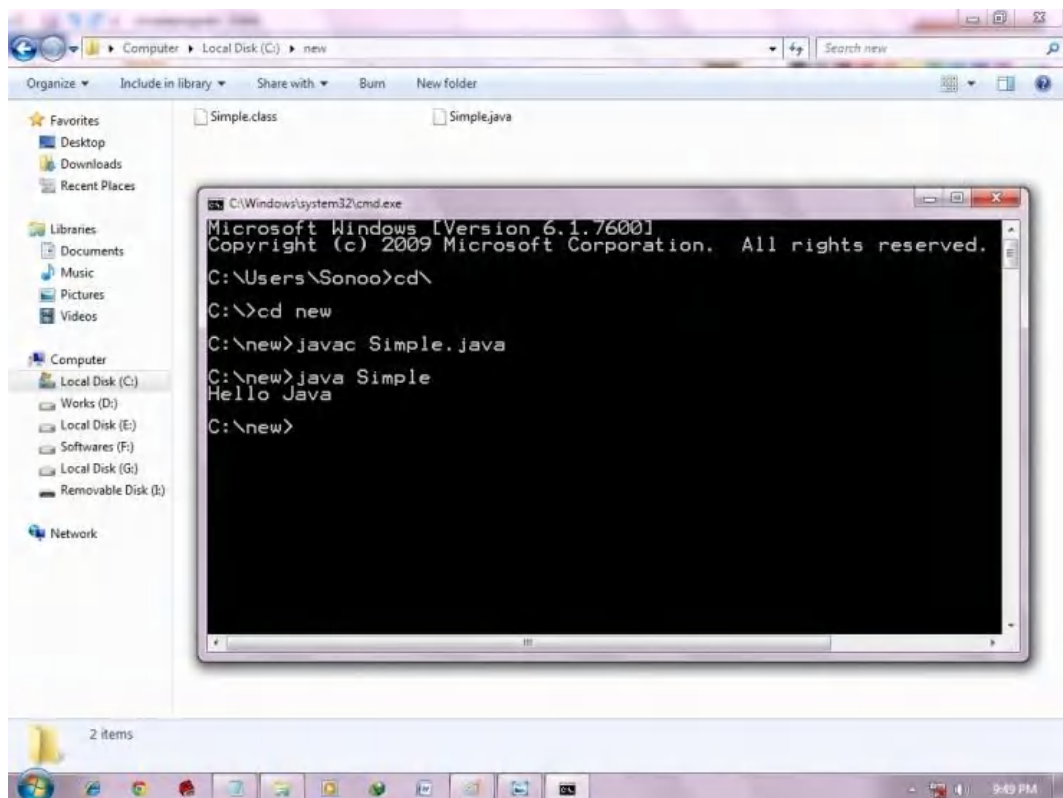


**Fig 2.9 Compilation of Java Program**

You need to have following information in order to compile and run program. You need to go to your current directory first; my current directory is c:\new and then writes here:

To compile:        `javac Simple.java`

To execute:        `java simple`

## Check your progress 6

1. Which is not associated with first Java program?

- a. class
- b. public
- c. static
- d. syntax

## 2.8 Skills check

There are certain skills involved in Java that can be worked out by considering the Java programs. If you want to develop and run any Java program, in such situation you have to first load JDK in your computer system that can be made available to you from

<http://www.oracle.com/technetwork/java/javase/downloads>.

After downloading you find that the initial screen will look as shown in fig 2.10. In the figure, the selected red box will allow you to download the kit.



Fig 2.10 Java Downloads



On selecting windows option and clicking on download your download will start that will ask you to save your file.

### Check your progress 7

1. To work on Java, you have to have:

- |             |                    |
|-------------|--------------------|
| a. computer | c. tool kit        |
| b. webcam   | d. development kit |

---

## 2.9 Exercises

---

### Example 1

Design a program in Java which will calculate the Fibonacci series of first 20 numbers as  $F(n)$ , where  $F(n)=F(n-1)+F(n-2)$  and  $F(1)=F(2)=1$ .

**Solution:**

```
public class Fibonacci {  
    public static void main (String args[]) {  
        int n = 3;           // the index n for F(n), starting from n=3  
        int fn;              // F(n) to be computed  
        int fnMinus1 = 1;    // F(n-1), hilt to F(2)  
        int fnMinus2 = 1;    // F(n-2), hilt to F(1)  
        int nMax = 20;       // maximum n, inclusive  
        int sum = fnMinus1 fnMinus2;  
        double average;  
        System.out.println("The first " nMax " Fibonacci numbers are:");  
        .....  
        while (n <= nMax) {  
            // Compute F(n), print it and add to sum  
            .....  
            //Adjust the index n and shift the numbers
```

```

        .....
    }

    // Compute and display the average (=sum/nMax)

    .....

}
}

```

On executing the above program, we see that first 20 fibonacci numbers are:

1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765

### Example 2

Write a Java program which will declare one float variable and other string variable which assigns numbers 5, 11 and shows values on screen.

Solution:

```

public class JavaExercises {
    public static void main(String[] args)
    {
        accessVariables();
    }

    static void accessVariables0{
        int x;
        float y;
        String s;
        x = 5;
        y= 11f;
        s = "Java programming";
        System.out.println(x);
        System.out.println(y);
        System.out.println(s);
    }
}

```

}

### Check your progress 8

1. Find the result obtained after executing this program.

```
int disp()
{
    int x,y=10,z=10;
    x=(y==z);
    System.out.println(x);
    return 0;
}
```

- |      |       |
|------|-------|
| a. 0 | c. -1 |
| b. 1 | d. 2  |

---

## 2.10 Let Us Sum Up

---

In this unit it is seen that Java is a programming language which was invented by James Gosling in 1991.

It was found that the first publicly available version of Java was released in 1995 as Java 1.0.

JVM is Java virtual machine that interprets assembled Java binary code known as bytecode for computer's processor which can do Java program instructions.

Java is called as Java 2 Platform having editions as Java 2 Standard Edition (J2SE), Java 2 Enterprise Edition (J2EE) and Java 2 Micro Edition (J2ME). It is noted that Java contains several classes of library that is pre-written.

---

## 2.11 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-d)

**Check your progress 2**

**Answers:** (1-a)

**Check your progress 3**

**Answers:** (1-c)

**Check your progress 4**

**Answers:** (1-a)

**Check your progress 5**

**Answers:** (1-a)

**Check your progress 6**

**Answers:** (1-d)

**Check your progress 7**

**Answers:** (1-d)

**Check your progress 8**

**Answers:** (1-b)

---

## 2.12 Glossary

---

1. **Applet** - It is an application which is a type of utility program that will do one or few simple functions.
2. **Version** - It is a sort of description or an account that differs from one point of view to another.
3. **JVM** - It is a program called as Java virtual machine that calculates assembled Java binary codes.
4. **Bytecode** - It is a computer object code which runs through a program known as virtual machine instead by actual computer machine.

5. **Class library** - It is a collection of prewritten classes or codes in shape of templates that can be specified and used by programmer during development of an application program.

---

## 2.13 Assignment

---

Write a program which will print "PASS" if int variable "mark" is more than or equal to 60 else it will print "FAIL".

---

## 2.14 Activities

---

Discuss the output obtained from this program.

```
int a=new int [10];  
int I,s=0;  
for(i=0;i<10;i++)  
{  
a[i]=I;  
s=s+a[i]+I;  
}  
System.out.println(s);
```

---

## 2.15 Case Study

---

Discuss about the Java program shown?

```
public class CheckOddEven {    // saved as "CheckOddEven.java"  
    public static void main(String[] args) {  
        int number = 49;        // set the value of number here!  
        System.out.println("The number is " + number);  
        if ( ..... ){  
            System.out.println(..... );  
        } else {  
            System.out.println( ..... );  
        }  
    }  
}
```

```
        }  
    }  
}
```

---

## 2.16 Further Readings

---

1. Rich Green, Open JDK 2011-11-25.
2. Fitzsimmons, Thomas, 2007 JAVA
3. Angel, Lillian, 2008 OpenJDK

---

## **UNIT 3: BEGINNING WITH JAVA PROGRAMMING**

---

### **Unit Structure**

- 3.0 Learning Objectives**
- 3.1 Introduction**
- 3.2 Java Keywords**
- 3.3 Comments in Java**
- 3.4 Variables and Assignments**
- 3.5 Strings and Characters**
- 3.6 Arithmetic Operators and Expressions**
- 3.7 Type Conversion in Expressions**
- 3.8 Type Conversion**
- 3.9 Skills check**
- 3.10 Exercises**
- 3.11 Let Us Sum Up**
- 3.12 Answers for Check Your Progress**
- 3.13 Glossary**
- 3.14 Assignment**
- 3.15 Activities**
- 3.16 Case Study**
- 3.17 Further Readings**

---

### **3.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Basic of Java Keywords
- Comments in Java
- Study about Variables and Assignments
- Strings and Characters

- Features of Arithmetic Operators and Expressions
- Types of Conversion in Expressions

---

## 3.1 Introduction

---

Java is a strong and simple object oriented programming language which is quiet similar to C++ language. It was invented by Sun Microsystems in year 1991. With the contribution of James Gosling, Patrick Naughton, Chris Warth, Ed Frank and Mike Sheridan, this language was intended to provide a platform-independent programming language.

The Java program is a set of instructions which a computer understand. The Java Compiler will translate programs from Java to computer language which functions as per computer to computer and with different operating systems. While performing, the Byte Code Program will translate Java into bytecode language.

---

## 3.2 Java Keywords

---

Java contains several different types of keywords and reserved words that are:

|                     |                 |
|---------------------|-----------------|
| Abstract Keyword    | Boolean Keyword |
| Break Keyword       | Byte Keyword    |
| Case Keyword        | Catch Keyword   |
| Char Keyword        | Class Keyword   |
| Continue Keyword    | Default Keyword |
| Do Keyword          | Double Keyword  |
| Else Keyword        | Extends Keyword |
| False Keyword       | Final Keyword   |
| Finally Keyword     | Float Keyword   |
| For Keyword         | If Keyword      |
| Implements Keyword  | Import Keyword  |
| Instance of Keyword | Int Keyword     |



|                   |                      |
|-------------------|----------------------|
| Interface Keyword | Long Keyword         |
| New Keyword       | Null Keyword         |
| Package Keyword   | Private Keyword      |
| Protected Keyword | Public Keyword       |
| Static Keyword    | Super Keyword        |
| Switch Keyword    | Synchronized Keyword |
| This Keyword      | Throw Keyword        |
| Throws Keyword    | Transient Keyword    |
| Try Keyword       | True Keyword         |
| Void Keyword      | While Keyword        |

### Check your progress 1

1. Which of the following keywords are used to define abstract class?
  - a. abst
  - b. abstract
  - c. Abstract
  - d. abstract class

---

## 3.3 Comments in Java

---

It is seen that comments serves as internal part of any program. With the help of comment, the programmer can read the code which will make them to understand better and can function as required.

### Type of Comments

Java consists of three language styles of comments such as:

- Single Line
- Multi-line
- Java doc

It is seen that comments in Java can be inserted anywhere in a program code where a white space is available. These comments are not included by Java compiler in final exe file. We can make as many Java comments as we want in our

program as per their usability. In java, the following styles of comments are entertained.

Java single line comments or slash-slash comments or end of line comments (//...)

Java multi-line or traditional comments (/\*...\*/)

Javadoc or Java documentation comments (/\*\*...\*/)

In Java, these single line and multi-line comments are commonly called as:

- documentation comments
- implementation comments

A documentation comment is related to the detail study of mathematics involved in case of class, field or method. It is analysed that a good documentation comments will make use of class and its methods without reading the source code.

An implementation comment will clarify the type of particular piece of code which works. Such type of comment is not commonly applied, but can be used as per need. In Java, the comment will appear on the right before declaration of class, interface or member with every line starting with "/\*".

### Java Single Line Comments

The single line comment in Java begins with two forward slashes (//) without white spaces that will go till the last line. Every line will contain additional two slashes, if the comment exceeds next line. These single line comments are useful for giving short explanations for variables, function declarations and expressions. The single line comment is shown as:

```
If (x < y)
{ // begin if block
x=y;
y = 0;
} // end if block
```

### Java Multi-Line Comments

The multi-line comments in Java will show text kept within slash-star (/\*) and star-slash (\*/). Here while applying such comments, there should be no white space between slash and star. It is required when comment text not fits in single line and needs spanning across lines. The multi-line comment is shown:

```
// CommentDemo.java
//Demonstrating multi-line comments

public class CommentDemo
{
    public static void main(String[] args)
    {
        for (int i = 0; i < 5; /* exits when i reaches to 5 */ i++)
        {
            System.out.println(i + " ");
        }
    }
}
```

If you perform above program, then the output obtained will be 0 1 2 3 4

In the above program, it is seen that in CommentDemo.java a comment is put inside loop header.

### Check your progress 2

1. Which is incorrect in case of Java comments?
  - a. Comments should be nested
  - b. Comments should have slash star
  - c. Comments should be surrounded by double quotes
  - d. All of these

---

## 3.4 Variables and Assignments

---

### Variables:

In a programming language, a variable is a place to keep the values that are used in a Java program. Before using such values, you need to declare it. Normally, declaring a variable is an initial part in any program. There are four types of variables available in Java programming language such as:

- **Instance Variables:** It is a type of Non-Static Fields where fields are declared without static keyword.
- **Class Variables:** It is a kind of Static Fields which can be declared with the help of static modifier that directs a compiler of its presence.
- **Local Variables:** It is a process where temporary state is stored inside the local variable which is visible to methods in which they are declared.
- **Parameters:** It is a main method which is public static void main(String[] args) where args variable serves as a parameter that are classified as variables not fields.

### Declare a Variable

Since Java is a strong programming language, so each variable in this should carry data types associated with it. The benefit of declaring a variable is that the following mentioned primitive data types can be used such as byte, short, int, long, float, double, char and boolean. Declaring a variable in Java requires a data type with variable name such as:

```
int numberOfDays;
```

Here, the variable "numberOfDays" was declared along with data type "int". In this the line gets ends with a semi-colon which tells the completion of declaration to the Java compiler. As seen, the declared variable, numberOfDays will only carry values which will be compatible with definition of data type.

### Initializing Variables

Initializing a variable means assigning value to a variable. If you are using a variable without assigning a value to it, then:

```
int numberOfDays;

// try and add 10 to the value of numberOfDays

numberOfDays = numberOfDays + 10;
```

In this we find that the compiler while execution will give an error as:

Variable numberOfDays might not have been initialized

Variables can be assigned in Java with the help of an assignment statement which carries similar pattern of equation as found in mathematics such as  $2 + 2 = 4$ . Here we see left side of an equation, right side of an equation and an equal sign ("=") in the middle. In order to assign the value to a variable, we find that left side is the name of variable while right side is the value of the variable as shown:

```
int numberOfDays; numberOfDays = 7;
```

We find that, numberOfDays was declared with data type of int with an initial value as 7. On adding 10 to value of numberOfDays to initialize, we see that:

```
int numberOfDays;  
numberOfDays=7;  
numberOfDays = numberOfDays + 10;  
System.out.println(numberOfDays);
```

Normally, initializing a variable can be done at the same time of its declaration as:

```
//declare variable and give it a value all in one statement int numberOfDays = 7
```

## Assignments

Normally, there are certain statements where you are adding something to a variable. So it is seen that in Java, the variables are assigned or given values with the help of assignment operators. As seen above, the variables are placed on left-hand side of assignment operator and values are placed on right-hand side of assignment operator. It is seen that an assignment operator will be calculated from right to left, hence  $x = y = z = 0$ ; will be assigned 0 to z, then z to y then y to x. Consider a variable shown:

```
i = i + 2;
```

Here assign i's value to new value which is i+2. You also write assignments using the += operator. Such operator can be applied for Arithmetic Operators.

### Check your progress 3

1. Which of these cannot be used for a variable name in Java?
  - a. identifier
  - b. keyword
  - c. identifier
  - d. keyword

## 3.5 Strings and Characters

### Strings

In Java, Strings serves as series of characters which are basically objects. Java platform will show a String class in order to form and control strings. Strings are common type of data present in the computer. Java contains certain basics such as chars, +, length () and substring ().

These are series of characters which are collected together like word "Hello". It creates a string in code with the help of chars in double quotes as:

```
String str = "Hello";
```

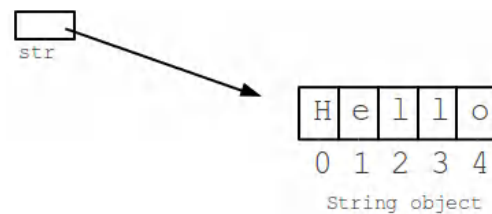


Fig 3.1 String Object

Fig 3.1 shows the string object in memory having individual chars H e l l o representing by index numbers 0, 1, 2.....

### Creating Strings:

One of the direct ways to form a string is to write as:

```
String greeting = "Hello world!";
```

On encountering a string literal in a code, the compiler will form a String object along with its value in "Hello world!". Apart from this object, you can create String objects with new keyword and constructor. It is seen that a String class carries 11 constructors which shows the initial value of string with various sources like an array of characters.

```
Public class StringDemo{  
    public static void main(String args0){  
        char[] helloArray = {'h' , 'e' , 'l' , 'l' , 'o' , '.'};  
        String helloString = new String(helloArray);  
        System.out.println( helloString );  
    }  
}
```

It will give an output as:

hello

### **String Length:**

The procedure used to achieve information about particular object is known as accessory method. The method used with strings is length () method that returns the number of characters present in the string object. After the following two lines of code have been executed, len equals 17:

```
public class StringDemo {  
    public static void main(String args[]) {  
        String palindrome = "Dot saw I was Tod";  
        int len = palindrome.length();  
        System.out.println("String Length is : " + len );  
    }  
}
```

This would produce the following result:

String Length is : 17

### **Check your progress 4**

1. Which among the following operators can be used to join character strings end-to-end in a string object?

|       |      |
|-------|------|
| a. +  | c. & |
| b. += | d.   |

---

## 3.6 Arithmetic Operators and Expressions

---

### Arithmetic Operator

In Java, the basic arithmetic operators are performed by certain operators as shown in table.

### Arithmetic Operators

| Operator | Meaning        | Example |
|----------|----------------|---------|
| +        | Addition       | 4 + 5   |
| -        | Subtraction    | 6 - 3   |
| *        | Multiplication | 3 * 4   |
| /        | Division       | 35 / 7  |
| %        | Modulus        | 34 % 5  |

In the table shown above, every operator carries two operands on either side of the operator. Here the subtraction operator is applied to negate a single operand, by multiplying operand by -1.

It is carefully assumed in case of division where on storing a division operation in integer, the output will be truncated to next lower whole number as int data type will not be able to carry out floating-point numbers. As an example, the expression 25/7 results in 3 if stored as an integer. In modulus division, % operator will form the remainder of division operation as 25% 7 results in 4 because 25 divided by 7 leaves a remainder of 4.

Consider a simple arithmetic example in Java, where there exists a Source File Weather.java

```
1: class Weather {  
2: public static void main(String[] arguments) {  
3: float fah = 86;  
4: System.out.println(fah + " degrees Fahrenheit is ...");  
5: // To convert Fahrenheit into Celsius  
6: // Begin by subtracting 32  
7: fah = fah - 32;
```



```
8: // Divide the answer by 9
9: fah = fah / 9;
10: // Multiply that answer by 5
11: fah = fah * 5;
12: System.out.println(fah + " degrees Celsius\n");
13:
14: float cel = 33;
15: System.out.println(cel + " degrees Celsius is ...");
16: // To convert Celsius into Fahrenheit
17: //Begin by multiplying it by 9
18: cel = cel * 9;
19: // Divide the answer by 5
20: cel = cel / 5;
21: // Add 32 to the answer
22: cel = cel + 32;
23: System.out.println(cel + " degrees Fahrenheit");
24: }
25: }
```

On running such Java program, we will find the output as:

86.0 degrees Fahrenheit is...

30.0 degrees Celsius

33.0 Degrees Celsius is...

91.4 degrees Fahrenheit

In Lines 3–12 of this application, you will see that temperature in Fahrenheit is converted to Celsius with the use of arithmetic operators as:

Line 3: The floating-point variable fah is created with a value of 86.

Line 4: The current value of fah is displayed.

Line 5: The comments are ignored by Java compiler.

Line 7: The fah is set to its current value minus 32.

Line 9: The fah is set to its current value divided by 9.

Line 11: The fah is set to its current value multiplied by 5.

Line 12: Since fah is converted to Celsius value, fah is shown again.

If we analyse the lines from 14–23, we see that temperatures in Celsius is converted to Fahrenheit with the help of `System.out.println()` in many statements. Such statement method is used in application to show strings and related information to particular output device. Such method takes a single argument in its parentheses which is a string.

### Expressions

Expressions in Java are a type of statement which shows a value. It is made up of variables, constants and method returned values along with operators. It contains certain side-effects in shape of actions that carries out at the time of finding an expression. The common mathematical expressions are shown with source code:

```
int x = 3;
```

```
int y = x;
```

```
int z = x * y;
```

The above three statements are considered as expressions if they shows values which are given to variables. If we consider the first expression, we see that the literal 3 is assigned to variable x. The second expression will assigns value of variable x to variable y with multiplication operator \* for multiplying x and y integers resulting and storing in z integer.

### Check your progress 5

1. The decrement operator "--" decreases the value of variable by \_\_\_\_.

a. 1

c. 3

b. 2

d. 4

### 3.7 Type Conversion in Expressions

It is seen in Java programming, various operation having different types of data can be applied in a single expression. For this consider an example shown:

```
char c;  
int i;  
float f;  
c = '1';  
i = 1;  
f = 1.1f;  
f = c + i - f;
```

As studied, Java make use of certain types of variable that can be in single expression as it contains a particular set of change over rules that can solve type differences.

It is seen that when a variable of type char, byte or short is applied in an expression, then its value will be directly changed to int on calculating particular expression. Such type of conversion rules in java is called as integral promotion which will only be in effect at the time of evaluation of an expression. In this the variable memory size will remain same.

## Check your progress 6

1. What will be the output of the following program?

```
int i;

float f;

i = 10;

f = 10.5f;

f = f - i;
```

---

## 3.8 Type Conversion

---

Type conversion is basically changing from given data type to specific data type. In Java, the type conversion is done directly when the type of expression present on right side of an assignment operator is endorsed to type of variable located on left of an assignment operator.

```
// 64 bit long integer  
  
Long myLongInteger;  
  
// 32 bit standard integer  
  
int myInteger;  
  
myLongInteger = myInteger;
```

The type conversion in Java can be:

- Implicit
- Explicit

An implicit conversion is such that in which the value of one type of an expression is changed to value of another type of expression without the use of particular directive from the programmer, while an Explicit conversions is basically carried off with the help of casting where the name of type to which a value is converted be kept in parentheses in front of value.

### Check your progress 7

1. Which among the following information is correct in case of automatic type conversion in Java?
  - a. Source type is bigger than destination type.
  - b. Destination type is bigger than source type.
  - c. Destination type is same as source type.
  - d. None of above

---

## 3.9 Skills check

---

In order to check your skills in Java you need to practice it. Since Java is a strong and simple object oriented programming language having similar commands as C++ has set of instructions that can be easily understood by computers. You can write programs in Java if you have sufficient knowledge about its operators, strings etc.

In this comments made the programmers to easily read the code that can make them understand better and can function as required. You need to master the variables in Java such as Instance Variables, Class Variables, Local Variables and Parameters that are used to apply in small programs where calculations are involved.

### Check your progress 8

1. Which is not a data type?

a. bits

c. short

b. byte

d. int

---

## 3.10 Exercises

---

### Example 1:

Design a Java program which will make the user to input two integer values and shows the result as operations of such integer?

To solve this using Java program as designed below:

```
import java.util.Scanner;

public class
{
    public static void main(String[] args)
    {
        caculateValues0;
    }

    static void caculateValues0 {
        int a,b;
```

```
int resultaxresults,resultm;  
float resultd;  
Scanner sc=new Scanner(System.in);  
System.out.print("Enter a:");  
a=sc.nextInt();  
System.out.print("Enter b:");  
b=sc.nextInt();  
resulta=a+b;  
results=a-b;  
resultm=a*b;  
resultd=(float)a/b;  
System.outprintln("The result of adding is "+resulta);  
System.outprintln("The result of subtracting is "-Fresults);  
System.outprintln("The result of multiplying is "-Fresultm);  
System.out.println("The result of dividing is "+resultd);  
    }  
}
```

In the above program, if we:

Enter value a:20

Enter value b:5

The result of adding is 25.

The result of subtracting is 15;

The result of multiplying is 100.

The result of dividing is 4.

---

## 3.11 Let Us Sum Up

---

In this unit we have learnt that Java is a strong and simple object oriented programming language that is similar to C++.

It is seen that Java program contains a set of instructions that can be easily understood by computer.

Comments in Java will allow the programmer to read the code that will be make them understand better.

As studied, variable is a place where values are kept for use by Java program the includes Instance Variables, Class Variables, Local Variables and Parameters

The benefit of declaring a variable is that the mentioned primitive data types can be used such as byte, short, int, long, float, double, char and boolean.

In Java, Strings serves as series of characters which are basically objects. Java platform will show a String class in order to form and control strings.

Expressions in Java are a type of statement which shows a value. It is made up of variables, constants and method returned values along with operators.

It is seen in Java programming, various operation having different types of data can be applied in a single expression

---

## 3.12 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-b)

**Check your progress 2**

**Answers:** (1-a)

**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-a)

**Check your progress 5**

**Answers:** (1-a)

**Check your progress 6**

**Answers:** (1-a)

**Check your progress 7**

**Answers:** (1-b)

**Check your progress 8**

**Answers:** (1-a)

---

### 3.13 Glossary

---

1. **Comments** - These are document containing java codes and program logic.
2. **Statement** - It is an independent unit in a program which is similar as english language.
3. **Block** - It is type of compound statement which is surrounded by curly braces { }.
4. **White Spaces** - These are blank, tab and newline spaces in java which are normally blank.
5. **Variable** - It is the name of storage location which will stores value of definite data type.
6. **Identifier** - It is a series of characters which is used to name a variable.

---

### 3.14 Assignment

---

Compare the types of variable present in Java and discuss.

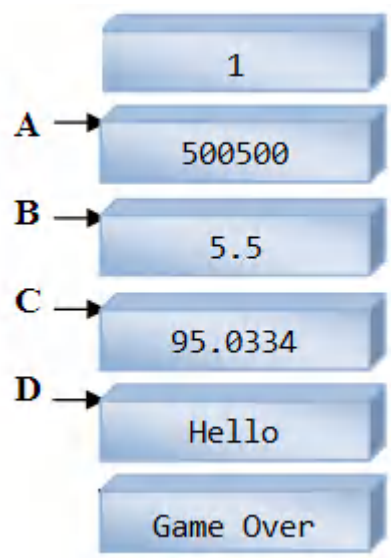


---

### 3.15 Activities

---

In a variable diagram shown, which alphabet shows the string variable?



---

### 3.16 Case Study

---

In a year having month from 1-12 and days from 1-31, what will be the boolean expression that will return true for dates that appears before September 8, 2015.

---

### 3.17 Further Readings

---

1. Learning Programming by Peter Norvig's
2. Approach programming with a more positive by P.Brian.Mackey

---

## **Block Summary**

---

In this block we will get sufficient knowledge about basic of Java and its history. The origination of Java along with its different versions is well explained. The three types of Java 2 Platform such as Java 2 Standard Edition (J2SE), Java 2 Enterprise Edition (J2EE) and Java 2 Micro Edition (J2ME) are illustrated with diagrams. The block detailed with sample exercises on Java program explained with different set of instructions made the students to practice more by varied concepts. The idea about Comments and different variables such as Instance Variables, Class Variables, Local Variables and Parameters are well explained.

After this block study, student can implement basic description about Java and its necessary tools that are used to design first Java program and subsequent Java versions. It is important for readers to read this block as basic of Java program design style are explained with which a student can write and design a program of its own.

---

## Block Assignment

---

### Short Answer Questions

1. What do you mean by programming?
2. What are Java Comments?
3. When was Java invented?
4. What do you mean by Java binary code?
5. What are Java Expressions?

### Long Answer Questions

1. What is the benefit of declaring a variable in a Java Program?
2. Explain Flowchart with examples?
3. What is the function of class library in Java?

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# RELATIONAL DATABASE MANAGEMENT

PGDCA 103



**BLOCK 1:  
INTRODUCTION TO  
DATABASE MANAGEMENT  
SYSTEM AND CONCEPTUAL  
MODELLING**

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# RELATIONAL DATABASE MANAGEMENT

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Ms. Katherine Abraham

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'





## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



## **RELATIONAL DATABASE MANAGEMENT**

### **Contents**

---

#### **BLOCK 1: INTRODUCTION TO DATABASE MANAGEMENT SYSTEM AND CONCEPTUAL MODELLING**

##### **UNIT 1 INTRODUCTION TO DATABASE SYSTEM**

Basic Concepts of Data and Database, Database management system, Data Administrator and its functions, Advantages of DBMS over file systems, Applications of DBMS, Three-tier architecture of DBMS, Data Models, Components of DBMS, Overview of Languages of DBMS (DDL, DML, DCL)

##### **UNIT 2 CONCEPTUAL MODELLING**

Introduction, Basic Concept of Entity-Relationship Diagram, Relationship concept and its types (unary, binary and ternary), Cardinalities (one-one, one-many, many-one and many-many), Overview of Extended ER model, Case Study of Extended ER model

---

#### **BLOCK 2: DATABASE INTEGRITY AND NORMALIZATION**

##### **UNIT 1 DATABASE INTEGRITY CONCEPTS**

Introduction, Domain Integrity constraints, Entity Integrity constraints, Referential Integrity constraints

##### **UNIT 2 NORMALIZATION**

Introduction, Basic Concept of designing, Need of normalization, Decomposition (lossy and lossless), Functional Dependency, Full Functional Dependency, Armstrong Axioms of Functional Dependencies, Normalization Rules (1NF, 2NF, 3NF and BCNF), Examples based on normalization



**BLOCK 3:    RELATIONAL ALGEBRA AND QUERY LANGUAGE**

**UNIT 1        RELATIONAL ALGEBRA**

Introduction, Basic concepts of relational algebra, symbols of relational algebra, operations on relational algebra, Examples based on relational algebra

**UNIT 2        QUERY LANGUAGE**

Introduction, Introduction to Structured Query Language, Data Definition Language (DDL), Data Manipulation Statement (DML), Data types in MySQL, Operators (Arithmetic, Comparison, Logical), SQL Functions, Group function (AVG, MAX, MIN, SUM and COUNT), COMMIT and ROLLBACK

**BLOCK 4:    DATA RETRIEVAL SQL STATEMENT AND TYPES OF DATABASE SYSTEM**

**UNIT 1        DATA RETRIEVAL SQL STATEMENT**

Introduction, Single table query without condition, Single table query with condition, Group by clause, Orders by Clause, Self Join Natural Join and Sub query, Examples based on SQL concepts

**UNIT 2        TYPES OF DATABASE SYSTEMS**

Introduction, Centralized Database System, Parallel Database Systems, Distributed Database Systems, Client-Server Database System



Dr. Babasaheb  
Ambedkar  
Open University

**PGDCA 103**

# **RELATIONAL DATABASE MANAGEMENT**

---

## **BLOCK 1: INTRODUCTION TO DATABASE MANAGEMENT SYSTEM AND CONCEPTUAL MODELLING**

---

### **UNIT 1**

|                                 |    |
|---------------------------------|----|
| INTRODUCTION TO DATABASE SYSTEM | 03 |
|---------------------------------|----|

### **UNIT 2**

|                      |    |
|----------------------|----|
| CONCEPTUAL MODELLING | 30 |
|----------------------|----|

---

# **BLOCK 1: INTRODUCTION TO DATABASE MANAGEMENT SYSTEM AND CONCEPTUAL MODELLING**

---

## **Block Introduction**

Database is a collection of files or records in electronic form, which is accumulating in addition that, are investigated by computer. Database involves single or multiple computer files to programme data in a tremendously programmed format. Database Management System, is a software program that that allows the formation as well as management of databases. A database is a prearranged arrangement of data essentials designed for the trouble-free collection of information.

In this block, we will learn and study about basic of database concepts and idea about conceptualisation. We will cover the topics related to Database domain, model, architecture and languages. The student will be given with the knowledge about different rules and criteria's of ER model.

The block will focus on a basic understanding about DBMS languages and ER concepts that will give the knowledge about cardinalities. The students or programmers will get benefit while reading this block as it gives shortcuts and related examples that will clear all doubts.

## **Block Objective**

**After learning this block, you will be able to understand:**

- Basic Concepts of Database
- Generalisation of Database management system
- Data Administrator and its functions
- About Three-tier architecture of DBMS
- Basic of Data Models
- Overview of Languages of DBMS (DDL, DML, DCL)

Introduction to  
Database  
Management System  
and Conceptual  
Modelling

- Concept of Entity-Relationship Diagram
- About Relationship concepts and its types
- Idea about Cardinalities

## **Block Structure**

**Unit 1: Introduction to Database System**

**Unit 2: Conceptual Modelling**



---

# **UNIT 1: INTRODUCTION TO DATABASE SYSTEM**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Basic Concepts of Data and Database**
- 1.3 Database Management System**
- 1.4 Data Administrator and Its Functions**
- 1.5 Advantages Of DBMS Over File Systems**
- 1.6 Applications Of DBMS**
- 1.7 Three-Tier Architecture Of DBMS**
- 1.8 Data Models**
- 1.9 Components Of DBMS**
- 1.10 Overview Of Languages Of DBMS (DDL, DML, DCL)**
- 1.11 Let Us Sum Up**
- 1.12 Answers For Check Your Progress**
- 1.13 Glossary**
- 1.14 Assignment**
- 1.15 Activities**
- 1.16 Case Study**
- 1.17 Further Readings**

---

## 1.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About File compilation in Database.
- Knowledge about Database Understanding of Metadata and Data dictionary.
- Know how about DBMS.
- Basic of database scheme.
- Overview of Data Independence.

---

## 1.1 Introduction

---

Database is a compilation of files or records in electronic form, which is stored on furthermore can be investigate by computer. A database real meaning is involving one or multiple computer files to encode data in an extremely prearranged format. Database Management System (DBM), is a software program that allows the formation as well as management of databases. A database is a prearranged arrangement of data essentials designed for the trouble-free collection of information. Unlike a set of flat files, a database surrounds both data as well as structural information that are used in taking out data from the files where data lives in.

In actuality, the database systems are more or less works as Relational Database Management System (RDBMS) as they have unique smart features that can store the required data across multiple tables.

A database management system (DBMS) is the grouping of computer software programs used for producing, organizing, retrieving, analysing as well as sorting information in computer based databases. This type of software is informally called as database software.

---

## 1.2 Basic Concepts of Data and Database

---

A database is presently the name given to a collection of data. The data is prearranged in several etiquette so that the information enclosed inside the database can be easily recovered. Some of the easy databases that you know are things such as phone books or rolodexes. At the same time, as data processing has turned out to be more complicated, there are certain methods for collecting,

storing as well as retrieving information. Databases have turned out to be the foundation stone for an irresistible amount of computing background in survival.

Databases over and over again fall into one of two broad categories. The initial category encompasses specific purpose and limited databases. In the academic world, these surround data to carry out a relatively incomplete role simply in a meticulous project. The database possibly will be planned to offer the researcher with an exacting set of data, however have no particular function or role at the ending of the project.

The next category encompasses general purpose, resource databases. A good illustration of a resource database is regional archaeological sites as well as monuments records (SMRs) that can be national monuments records. Such databases are not projected as precise. Resource databases more often than not challenge to be inclusive within their area of conversation that are maintained as well as updated, plus are prepared as per the interested parties. Such databases challenge to be inclusive in order to contain unexpected enquiries as well as research, but these cover a broad array of data which in turn needs a complex data structure, or technique of storing the information.

The CISP database is anticipated to be a source database moreover as a result has a complex data structure. Such data structure, on the other hand, provides huge power as well as flexibility equally for the retrieval as well as for the handling of the data, other than the future development of the database to contain other information in addition to materials.

## **Data**

It is a peculiar form of information that are arranged or organised in special ways. Basically, every software carries data and programs. It is found that a program is a mixture of instructions that helps in carry out data. You can arrange data in numerous forms in shape of numeric or text and can be explained on paper by way of bits and bytes kept in memory.

## **Metadata**

Metadata also a form of data dictionary which tells about a particular data. With the help of database it can be explained through program data independence. Metadata keeps the below mentioned information about every data element inside. These are:

- Name
- Type

- Range of values
- Source
- Access authorization

The above listed data an element applies with all related application programs so that if any alteration in data structure required, then affected programs will further be obtained. It will handle all database operation along with data integrity and correctness. Such type of data is mostly utilised by developers in order to construct programs, queries, controls and procedures so that the data can be controlled and carry out certain programs.

### **Data dictionary**

The file system in database will describes basic arrangement of a database. So data dictionary carries certain list of related files which is present inside database. These will contain number of records with details about names and types of every field. It is seen that mostly some DBMS makes such files hidden from users so that it should not be damaged or destroyed. It is many times seen that data dictionary will not contain real data from database, but only enter such information in book so as to handle it. If the data dictionary is not present, then DBMS will not be able to work with data which is present inside database.

### **Check your progress 1**

1. Database is:

- |                          |                              |
|--------------------------|------------------------------|
| a. Collection of files   | c. Collection of information |
| b. Collection of records | d. All of above              |

2. Database Management System is:

- |                         |                         |
|-------------------------|-------------------------|
| a. Software Program     | c. Collection of Fields |
| b. Collection of Record | d. None of above        |

3. Open Database Connectivity is:

- |             |                  |
|-------------|------------------|
| a. Driver   | c. Fields        |
| b. Software | d. None of above |

---

## 1.3 Database Management System

---

Database Management System or DBMS in diminutive refers to the know-how of storing as well as retrieving users' data with highest efficiency all along with suitable security method.

A DBMS creates it probable for end users to create, read, update as well as delete data in a database. The DBMS fundamentally provides an interface involving the database as well as end users or application programs, that guarantee that the data is again and again organized plus remains easily available.

The DBMS administers 3 important things:

- the data
- the database engine
- the database schema

These above 3 foundational elements help provide concurrency, security, data integrity as well as uniform administration measures. Characteristically, database administration responsibilities maintained by the DBMS consists of:

- change management
- performance monitoring/tuning
- backup and recovery

Many database management systems are in addition answerable for mechanical rollbacks, restarts along with recovery as well as the logging also auditing of activity.

The DBMS is for the most part useful for providing a federal view of data that can be admittance by numerous users, as of multiple locations, in a guarded behaviour. A DBMS can bind what data the end user will observe, as well as how to facilitate he end user viewing the data, as long as numerous views of a single database representation. End users as well as software programs are free from having to know where the data is physically situated and otherwise on what type of storage space media it resides inform the reason that the DBMS handles all requests.

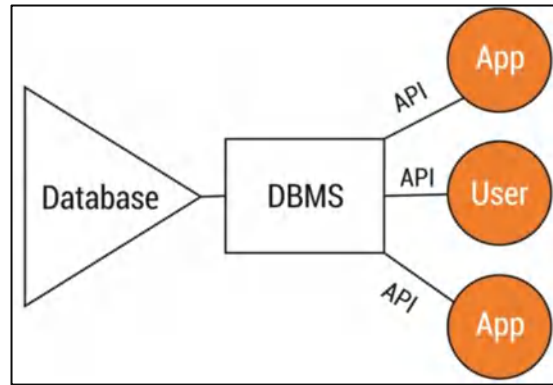


**Fig 1.1 DBMS system**

The DBMS can present equally logical as well as physical data independence. With this aim, it saves the users as well as applications from needing to be familiar with where data is stored regarding changes to the physical structure of data in terms of storage and hardware. As long as programs make use of the application programming interface (API) designed for the database with the aim to show by DBMS, developers that won't require modifications programs as changes have also been added to the database.

By means of relational DBMSs (RDBMSs), such API is SQL, which is a basic programming language that defines, protect as well as access data in a RDBMS. With DBMS, it is easy to store as well as manage data that comes with advantages, but also fixed cost. One of the major advantages of using a DBMS is that it gives permission to end users as well as application programmer's the right to use as well as allow the use of same data along with managing data integrity. The data has a superior protection plus maintained when it can be shared using a DBMS as a substitute for producing new iterations of the equivalent data stored in new files for each original application. The DBMS makes accessible a central store of data with the intention that it can accessible by multiple users in controlled approach.

A DBMS can in addition make available a lot of views of a particular database schema. A view describes what data the user sees in addition to how that user observes the data. The DBMS offers a level of concepts and wiced between the conceptual schema that defines the rational structure of the database as well as the physical schema that explain the files, indexes in addition to other physical mechanisms used by the database. As soon as a DBMS is used, systems can be customized a large amount further easily when business requirements change. New categories of data can be added to the database without disrupting the existing system and applications can be insulated from how data is structured and stored.

## Database Schema

A database schema is the structural arrangement to facilitate the rational view of the whole database. It illustrates how the data is predetermined and furthermore how the relation is stuck among them that are correlated. It positions mutually each and every restriction that is to be functional on the data. A database scheme portrays its entities as well as relationship that exist among all. Further, it controls a vivid detail of the database that is able to correspond by way of scheme illustration as shown in fig 1.2. It is the database designer who sketches the representation to help out programmers to be responsive of databases in accumulation to work constructively.

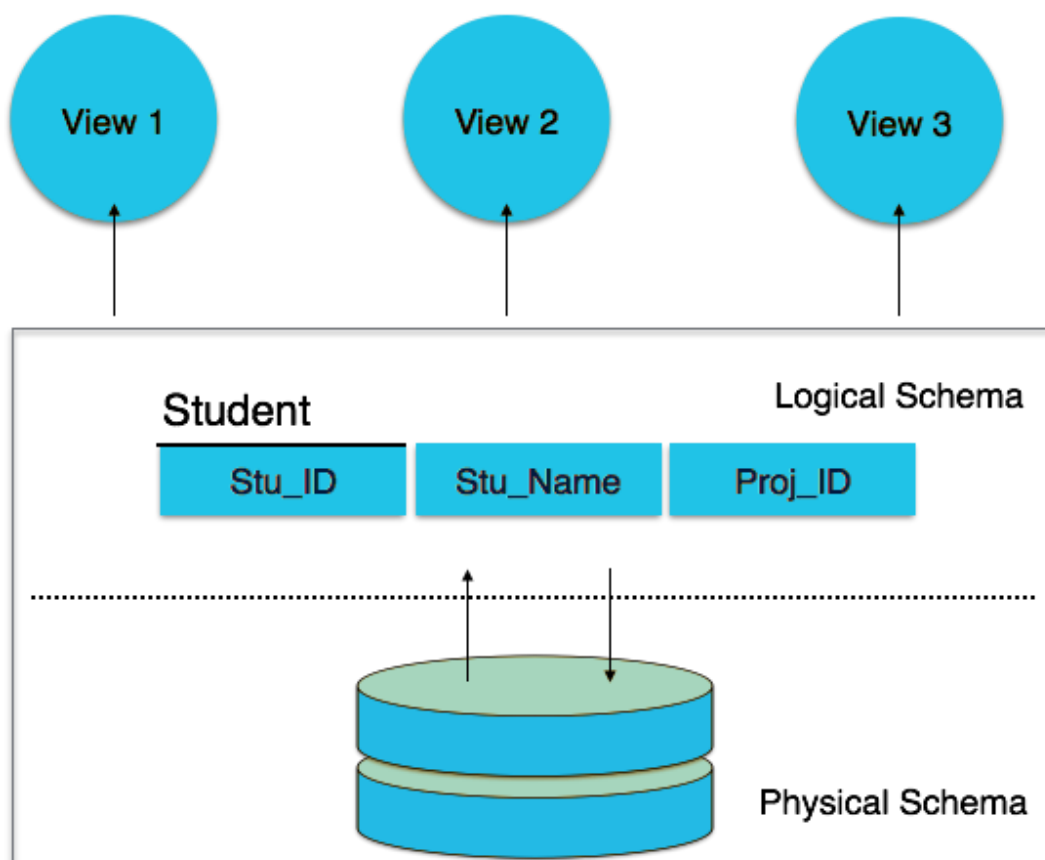


Fig 1.2 Database schema

A database schema can be separated roughly into two groups -

**Physical Database Schema** - Such type of schema relates to the definite storage space of data as well as forms of storage space like files, indices, etc. It explains how the data be determine with the purpose of storage in secondary part.

**Logical Database Schema** - Such type of scheme illustrates search and every rational constraint with the intent of pre-requisites to be practical on the data stored. It shows tables, views, as well as integrity limitations.

## Database Instance

It is imperative with the idea of differentiation that occurs with two terms separately. Database representation is the structure of database. It is designed rapidly as if the database doesn't survive at all. It is observed that as soon as the database is carried out, after that it turns out to be hard to prepare any alteration in it. A database representation does not manage every data or information.

A database graphic is a situation of a performing database with data at numerous prearranged times. It contains a snap of the database. Database incidents have a predisposition to modify by way of time. A DBMS formulates crystal-clear as a result that every incidence is stated to be in applicable situation, by consistently sub sequencing each and every validation, constraints, as well as situation to facilitate the database designers to work.

## Data Independence

A database system in general controls a bunch of data in accumulation to users' data. For illustration, it stores data in relation to data, recognized as metadata, to locate as well as to get back data without difficulty. It is somewhat tricky to change or bring up to date a position of metadata as it stores it in the database. Although since a DBMS is spread out, it desires to transform over time to please the needs of the users. If the whole data is dependent, it would roll out to be multifaceted yet boring.

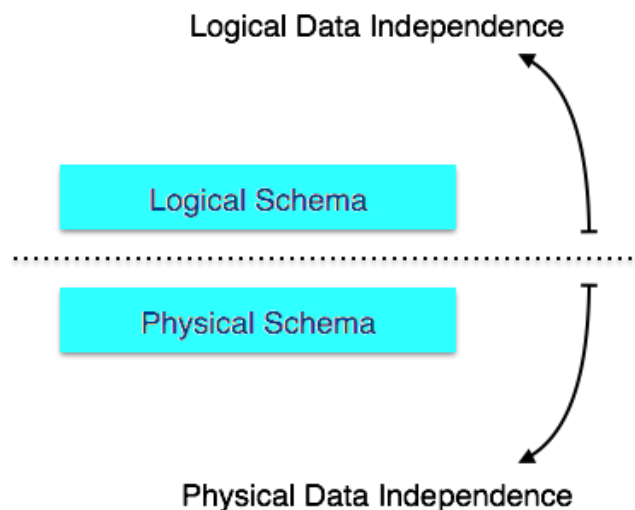


Fig 1.3 Data Independence

Metadata itself go subsequent to a layered structural design, as a result when we amend data at a particular layer, it does not have an outcome on the data at supplementary stage. This data is self-determining, but never the less it is mapped to each one.



## Logical Data Independence

Logical data independence relates to a type database which shows details about how the data will be handled or arranged inside the system. If you happen to analyse a relation that was created in database by considering several constraints and at the same time practice on such relation, then you will find that it is direct which shows particular data storage on a disk.

If we do several amendments on table arrangement, it should not revolutionize the data which is available in the disk.

## Physical Data Independence

Each and every representation in Physical Data Independence is rational; furthermore the real data is accumulated in a bit that is created on the disk. Such type of data independence is authorized to mutate the physical data so as not to cover the collision of representation or logical data.

For instance, in a situation, if we hope to transform or improve the storage system itself, we have to revolutionize the hard-disks by means of SSD, further which should not give several contacts on logical data or diagram.

### Check your progress 2

1. The database schema is written in
  - a. HLL
  - b. DML
  - c. DDL
  - d. DCL
2. A logical schema
  - a. is the entire database.
  - b. is a standard way of organizing information into accessible parts.
  - c. describes how data is actually stored on disk.
  - d. both a. and c.

3. One of the advantage of database management is
- a. data depends on programs.
  - b. data dismissal increases.
  - c. data is incorporated in addition to admittance by multiple programs.
  - d. none of the above.

---

## 1.4 Data Administrator and Its Functions

---

A data administration also recognized as database administration manager, data architect as well as information centre manager, is a far above the ground level function which is in charge for on the whole management of data assets in an group. In order to carry out its duties, the data administrator should recognize a high-quality agreement of system analysis as well as programming. These are the functions of a data administrator:

1. Ideate about data policies, process and standards.
2. Development of organization's IT view, project model, cost/profit model, recommendations of database impression with supervision map.
3. Data divergence motion.
4. Data investigation which covers the aim as well as description of model data needs, industry rules, functioning needs and saving commercial data glossary.
5. Inside promotional impression.
6. Administering the data storage area.

### Check your progress 3

1. \_\_\_is a person that is responsible for data, metadata, and policies related to data usage.
- |                           |                     |
|---------------------------|---------------------|
| a. Data administrator     | c. Database steward |
| b. Database administrator | d. Both a and b     |

---

## 1.5 Advantages of DBMS Over File Systems

---

A database is designed for data storage space as well as data recovery. As soon as you place data into a database table, the database table is indexed in different ways. As a result of that it depends on a number of columns you searched by which data at all times survive as quickly as possible.

Individually, the database does such activity by loading as much data into RAM as feasible also called as buffer cache. This cache seizes all the data that has been newly used inside the database; hence it can be established faster than by looking for it on the disk. As the data in the cache turns out to be idle, it is noticeable to be glowing so that if new data desires to be loaded into cache the database can create room for it.

Database servers in addition hold up numerous people reading as well as writing the data at the similar time. Databases agree for locks to be engaged on precise rows, data pages, or entire tables depending on the need at the time, as a result of this, one person can perform a transaction beside the database where as other users can't update those records, pages, or tables until the initial transaction has finished.

The database approach offers a number of potential advantages as compared to traditional file processing systems.

1. **Program-Data Independence** - The parting of data descriptions on or after the application programs uses is referred as Data independence. By means of such database approach, data descriptions are accumulated in a central location referred to as repository. Such quality of database systems permit an organization's data to alter with no altering the application programs so as to process the data.
2. **Data Redundancy and Inconsistency** - During File-processing, files having dissimilar formats as well as application programs might generate by changed programmers. Likewise dissimilar programs could be on paper in numerous programming languages. The identical information positioned at dissimilar files cause laying-off as well as inconsistency and thus higher storage as well as admittance costs.
3. **Difficulty in accessing Data** - In conventional file arrangement, the data or information is collected and stored in various files. It is easy storing data in files, as you can take the data back any time where there is a need on

accepting the prior approval and on request of application program. This is a tire some procedure.

4. **Data isolation** - On account of data there is a wide complexity of files, which are conceivably in asymmetric formats. It is adverse to formulate new application programs to accumulate the appropriate data.
5. **Combined access** - There continues no central administration of declaration in conventional file organization. As an effect, the concurrent admittance of data proximate many users to use it.
6. **Safety Problems** - In reality there continues no main administration of data in conventional file arrangement wherefore, safety, compulsion is not alleviated in File-processing approximation.
7. **Integrity Problem** - The declaration values acquired in the database should satisfy certain categories of consistency constraints. For prototype, the ledger of bank assets feasibly will never fall down lower than an affirmed charge. These constraints are essential in the approach nearby accumulating together appropriate code in the distinct application programs. On the other hand, when authentic constraints are accumulated, it endures to alter the programs to set up into sequence. The difficulty is combined when prohibitions approach numerous data elements from distinct files.
8. **Advanced Data assigning** - A database is arranged as an assorted corporate resource. Legal inward as well as exterior users are categorized with approval to develop application of the database. Additionally, every user is allowed with one or additional user observation to create potential use. A consumer witness is a feasible reputation of numerous divisions of the database that is essential for an employer.
9. **Improved Creativeness of Application Progress** - One of the most important benefits of database is to move in line as it is an important aspect which considerably lowers the amount as well as time of establishing new business submissions.

Conveniently there will be two important points where data base gathering can frequently be urbanized into a lot supplementary fast as compared to conventional file applications.

- a) Contemplating that the database as well as the associated data contains support applications subsequently planned and exercised the programmer can essence on the catalogued functions compulsory for

the fresh application, with no worry about file arrangement or low-level facilitation facts.

- b) The data base management approximation approach is an allotment of high-level productivity gadgets like forms plus logs generators along with high-level dialects to facilitate computerize operations of the database concept as well as facilitation.

### Check your progress 4

1. Which among the following is an advantage of database management?
- a. In this, data is dependent on programs.
  - b. In this, data redundancy increases.
  - c. In this, data is integrated and can be accessed by multiple programs.
  - d. none of the above.

---

## 1.6 Applications of DBMS

---

Database management systems is applicable on developing and forming a database, pile it up with data or information and forms different ways to reserve all along by way of modifying the particular information not including the technological aspects of data storage as well as reclamation. Apart from this, some of the characteristics of database management systems consist of:

- User admission as well as security administration systems presenting appropriate data contact to numerous users at the same time protecting perceptive data.
- Data encouragement to formulate definite and consistent accessibility of data.
- The access logs makes smoother for database manager to justify the use of database that is being applied.
- The rules enforcement will gives surety about prescribed data that is kept in every field. This is stated as date fields which are adjusted for particular data having dates in specific range.

- In DBMS, formulae like counting, averaging as well as summing up makes arithmetical examination of simple data.
- Performance monitoring as well as optimization tools that shows permit to users to adjust database settings that relates to speed along with efficiency.

Numerous web applications depends on DBMS that will vary from search engines as well as article, directories to social networks network like Facebook along with Twitter.

### Check your progress 5

1. The purpose of DBMS is to:

- |                    |                   |
|--------------------|-------------------|
| a. Keep record     | c. Display record |
| b. Maintain record | d. all            |

---

## 1.7 Three-tier architecture of DBMS

---

The plan of a DBMS applies only to its architecture. This could be centralized or decentralized as well as hierarchical. It is found that the DBMS architecture is single tier or multi-tier. With N-tier architecture, it is seen that the DBMS is divided into complete system that relates to independent N modules, which is independently altered, changed or further replaced.

In single tier DBMS architecture, with single entity, the user can simply sit on database as well as can use such database. If the user makes any alteration, then this will simply alter the DBMS directly. For this, it will not carry any handy tools to be used by end users. The creator or designers of database will simply wish to apply single-tier architecture.

In case of 2-tier DBMS, the application will pass through DBMS in order to access straight. Programmers if applying 2-tier architecture will simply access DBMS through its application. We see that the application in 2 tier DBMS is a wholly self-governing of database in relation to process, plan and indoctrination.

### 3-tier Architecture

In case of 3-tier DBMS architecture, the difference among other tier arrangements is different from each other. This applies on the convolution of users by means of opinion as to how they apply data that is available in the database.

The 3 tier architecture is for the most part used architecture used in forming DBMS. Fig 1.4 shows this arrangement.

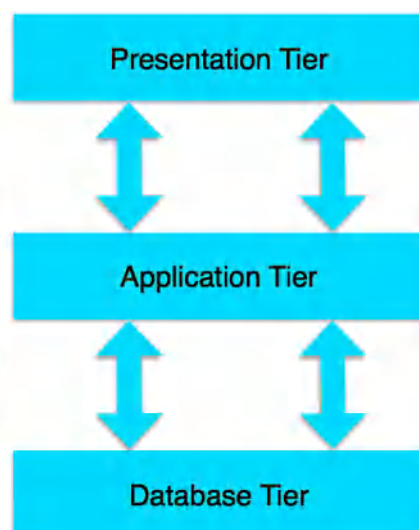


Fig 1.4 3-tier architecture

- **Database (Data) Tier** - In this arrangement, the database is available all along with its query processing languages.
- **Application (Middle) Tier** - In this type of arrangement, the database is available in application servers with programs to work with the database. This application is good for user as it explains a conceptual analysis of database. End-users are ignorant of some continuation of database away from the application.
- **User (Presentation) Tier** - In this type of arrangement, we see that the operator uses the database without having any knowledge about existence of database further than this layer. In this many views of database are shown with the help of applications.
- **Multiple-tier** - In this type of database architecture, you can modify or edit, as roughly each and every component is autonomous. Furthermore, it can be transformed separately.

### Check your progress 6

1. Presentation tier is used by the:

- |                |                    |
|----------------|--------------------|
| a. Programmers | c. Software people |
| b. End users   | d. All             |

---

## 1.8 Data Models

---

A Database model defines the logical design of data. The data model is an actual description of analysing real world objects along with several measures in relation to description. It is a concept which facilitates on required, intrinsic characteristic of an association as well as overlooks the unplanned properties. A data model approximates the arrangement itself. It should assign the foundational conceptions as well as emblems that consideration allocates database designers as well as end users unambiguously additionally correctly to broadcast their comprehension of the organizational data.

Data Model can be described as an accomplished coalition of considerations for accounting as well as skimming data, connections between data, as well as confinements on the data in an organization. A data model encompasses of three elements:

- A compositional featured, consisting relevantly a set of controls according to which databases can be developed.
- A manipulative measure, describing the categories of performance that are assigned on the data.
- Feasibly a set of collectivity rules, which promises that the data is correct.

The approach of a data model continues to approximate data also to construct the data perspicuous. There acquire been ample data models endeavoured in the composition. The model explicates the connections between contradictory elements of the data. In the history of database design, three models have been in use.

- Hierarchical Model
- Network Model
- Relational Model
- Entity-Relationship Model

### The hierarchical model

The data is arranged in a hierarchy with the help of downward tree. In such models, pointers are applied to steer among stored data. Such model was initially a DBMS model.



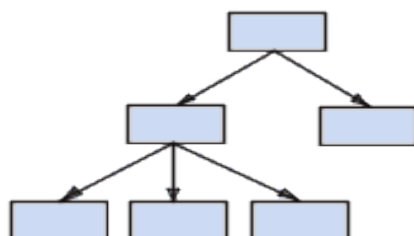


**Fig 1.5 hierarchical structure**

We see that in such type of model, every entity contains a single parent having many children. As seen, the top of hierarchy carries single entity known as root.

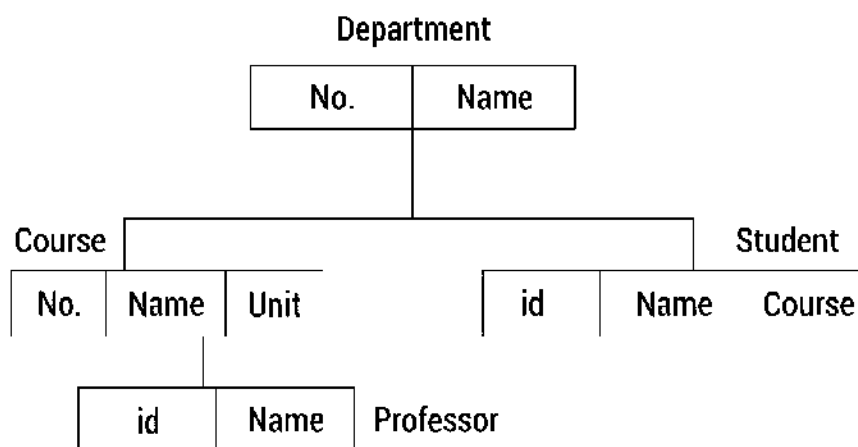


**Fig 1.6 hierarchical model**

### Network model

Just like the hierarchical model, the network model also utilizes pointers in the way as the data is kept. On the other hand, it does not automatically make use of the downward tree structure.

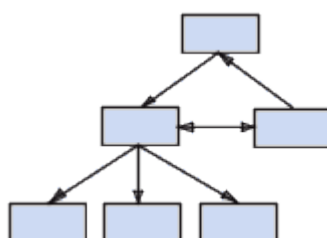


**Fig 1.7 network structure**

In the network model, entities are organised in a graph, in which some entities can be accessed through several path

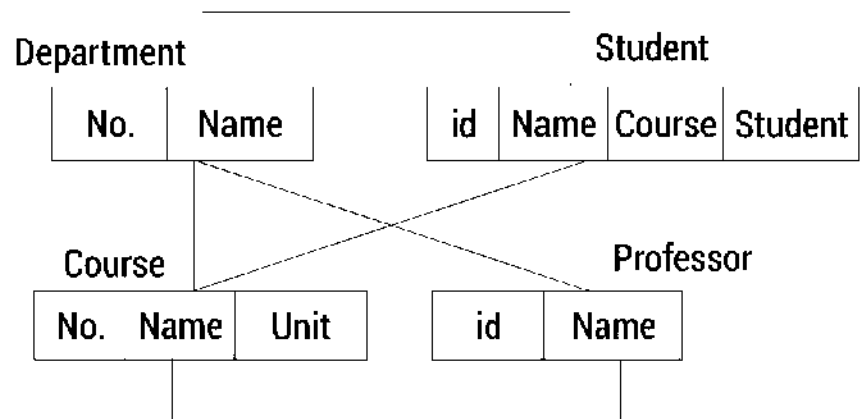


Fig 1.8 network model

### Relational model

In this model, data is kept in 2 dimensional tables having rows as well as columns. In this, the data is calculated depending upon the relational theory of mathematics.

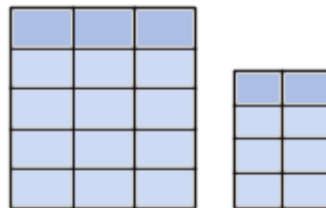


Fig 1.9 relational model

Further it is seen that, in such model, the data is gathered in 2-dimensional tables as shown above known as relations. It is observed that these tables or relations somewhat relate to each other. This arrangement is quiet famous as it is more scientific model as compared to rest. It is based on first-order predicate logic and defines a table as an n-ary relation.

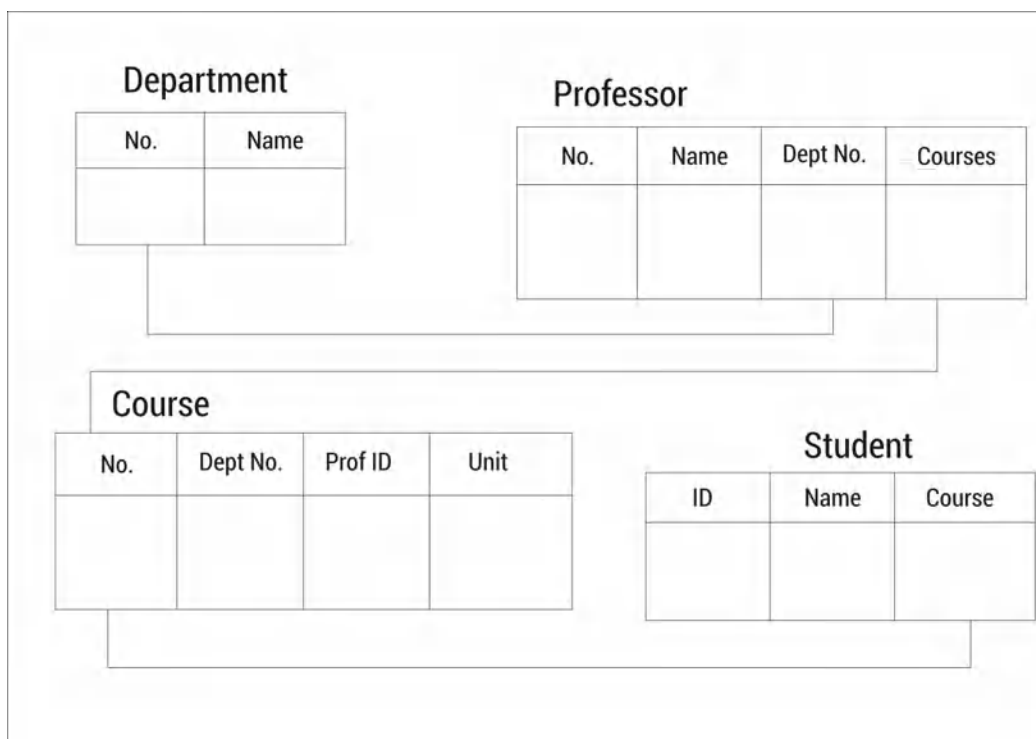


Fig 1.10 relational model

### Entity-Relationship Model

Entity-Relationship (ER) Model is a type of database model structure that is based on conception of real world things as well as relations surrounded by them. At the same time, the preparation of the actual world situation hooked on the database model, ER Model generates entity set, relationship set, common characteristic as well as constraints. It is the most excellent model applied for theoretical plan of a database. It is based on –

- Entities and their attributes.
- Relationships among entities.

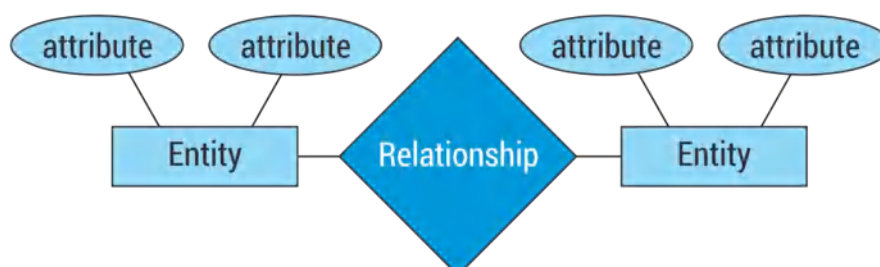


Fig 1.11 ER model

## Entity

We can say that the unit of ER Model is an actual entity example that carries properties called attributes. Every characteristic is a definite by set of principles called domain. For instance, while considering school database, a student is considered as an entity having various characteristics such as name, age, class, etc.

## Relationship

The rational relationship amongst entities is called correlation. Relationships are mapped by means of entities portraying different behaviour. Mapping cardinalities characterize the extent of relationship among two entities. In this, the mapping cardinalities are:

- one to one
- one to many
- many to one
- many to many

### Check your progress 7

1. Hierarchical model is also called

- |                   |                        |
|-------------------|------------------------|
| a. Tree structure | c. Normalize Structure |
| b. Flex Structure | d. Table Structure     |

2. Which of the following is record based logical model?

- |                          |                  |
|--------------------------|------------------|
| a. Network Model         | c. E-R Model     |
| b. Object oriented model | d. None of these |

---

## 1.9 Components of DBMS

---

Databases are composed of related tables, which in turn are composed of fields and records.

### Field

Every record contains specific fields to store data. The field contains user information such as name, address, city, phone numbers etc. Normally the fields are defined by:

#### Field name

#### Data type

- Character: Details about customer telephone numbers and pin codes
- Numeric: it carries numbers that can be calculated using certain mathematical operators
- Date: it will show numeric dates which can be calculated mathematically.
- Logical: It state True or False or can also stress on Yes or No

#### Field size

- It shows the capacity of data storage

### Record

It is a data in shape of values which is for related fields that is for single entity. It refers to particular person, product, company etc.

### Table

It shows details of particular records. It shows the examples as employee table, product table, customer, and orders tables.

The table contains rows and column, so the records can be seen in terms of row and fields can be seen in terms of columns. We see that database is a mixture of rows and columns.

### Relationships

In the table, the relationship exists in three types:

- One-to-One
- One-to-Many

- Many-to-Many

Among the above, the famous relationship in relational database is One-to-Many and Many-to-Many.

To see for One-to-Many relationship, we see that the presence of Customer table and Orders table having single customer that can do multiple orders.

In case of One-to-Many relationships, we find that it contains two tables, “one” table and “many” table.

Normally, it is found that an example of Many-to-Many relationship relates to Orders table and Products table.

If we see Many-to-Many relationship, we find that it will have three tables: two “one” tables and one “one” table.

### Key Fields

Two fields are related to each other if they have common fields. It is seen that a key field in “one” table of One-to-Many relationship has to be the main key. If we see for the similar key in “many” table of One-to-Many relationship such type of key will be narrated as foreign key.

### Check your progress 8

1. Database is composed of:

- |            |                 |
|------------|-----------------|
| a. Table   | c. Fields       |
| b. Records | d. All of above |

2. Data type consists of:

- |              |                 |
|--------------|-----------------|
| a. Character | d. Logic        |
| b. Numeric   | e. All of above |
| c. Date      |                 |

3. Units of data within a database are generally called:

- |            |                  |
|------------|------------------|
| a. Driver  | c. Fields        |
| b. Records | d. None of above |

---

## 1.10 Overview of Languages of DBMS (DDL, DML, DCL)

---

A DBMS is a software pack that does various tasks which will cover provision of facilities to allow the user to contact as well as change information in the database. The database is an intermediate link between the physical database, computer and the operating system and the users. To provide the various facilities to different types of users, a DBMS normally provides one or more specialized programming languages called database languages.

Database languages come in different forms: -

1. Data Description Language (DDL)
2. Data Manipulation Language (DML)
3. Data Control Language (DCL)

### DML

DML is abbreviation of Data Manipulation Language. It is used to retrieve, store, modify, delete, insert and update data in database. Some examples:

- SELECT - recover data from database
- INSERT - introduce data in table
- UPDATE - bring up to date the present data in a table
- DELETE - erase all account from a table
- MERGE - insert or update operation
- CALL - bring PL/SQL or Java subprogram
- EXPLAIN PLAN - give details about access path to data
- LOCK TABLE - manage concurrency

### DDL

DDL is short form of Data Definition Language. It is applied to produce as well as change the arrangement of database substance in database. Some examples:

- CREATE - produce objects in the database
- ALTER - convert the arrangement of database
- DROP - obliterate materials from database
- TRUNCATE - acquire away each as well as every descriptions from table
- COMMENT - accumulate acknowledgment to data documentation
- RENAME - give another identity to the material

## DCL

DCL continues compaction of Data Control Language. It is exercised to develop reverences, approvals, as well as referential completeness as well it is utilized to administer approach to database by guarding it. Some examples:

- GRANT - gives user's access privileges to database
- REVOKE - withdraw access privileges given with the GRANT command

### Check your progress 9

1. DDL is:

- |                           |                             |
|---------------------------|-----------------------------|
| a. Dynamic Data Language  | c. Data Definition Language |
| b. Detailed Data Language | d. Data Derivation Language |

---

## 1.11 Let Us Sum Up

---

A database involves one or multiple computer files to encode data in an extremely prearranged format. A Database Management System is a software program that allows the formation as well as management of databases. Database is a compilation of files or else records in electronic form, which can be searched by computer. DBMS is a type of software programming that is used to arrange and frame all information and data in a database.

It serves as a mixture of different software programs that are used to arrange, sort, set and organise computer-based databases. Open Database Connectivity (ODBC) is a driver that allows the database to integrate with other databases. Some of the common relational database management systems are



Microsoft Access, File maker, Microsoft SQL Server, MySQL, Oracle etc. Databases are composed of related tables, which in turn are composed of fields and records.

We see that a field is specific area that is present inside the record which will keep the particular data. A record is the collection of values for all the fields pertaining to one entity. A table is a collection of related records. Units of data within a database are generally called records.

Further it is seen that a DBMS is a software package which performs many tasks which will cover provision of facilities to allow the user to contact as well as can alter any information present inside database.

---

## 1.12 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-d), (2-a), (3-a)

**Check your progress 2**

**Answers:** (1-c), (2-a), (3-c)

**Check your progress 3**

**Answers:** (1-a)

**Check your progress 4**

**Answers:** (1-c)

**Check your progress 5**

**Answers:** (1-d)

**Check your progress 6**

**Answers:** (1-d)

**Check your progress 7**

**Answers:** (1-a), (2-a)

**Check your progress 8**

**Answers:** (1-d), (2-e), (3-b)

**Check your progress 9**

**Answers:** (1-c)

---

## 1.13 Glossary

---

1. **Hardware** - It contains computer components.
2. **Software** - It is a program that contains operating system, utilities, files and applications programs from data stored in files.
3. **Procedures** - The instructions and rules that govern design and use of software component.
4. **Data** - The collection of facts.

---

## 1.14 Assignment

---

Explain the Components of DBMS?

---

## 1.15 Activities

---

Write features of Data Administrator and its functions?

---

## 1.16 Case Study

---

Compare DDL, DML and DCL?

---

## 1.17 Further Readings

---

1. Asher, H.B. (1984). Causal modelling. Sage University Paper series on quantitative applications in the social sciences, 07-003. Newbury Park, CA: Sage Publications.

2. Creswell, J.W. (1994). Research design: Qualitative and quantitative approaches. Thousand Oaks, CA: Sage Publications.
3. Kerlinger, F.N. (1979). Behavioural research: A conceptual approach. New York: Holt, Rinehart & Winston.

---

## UNIT 2: CONCEPTUAL MODELLING

---

### Unit Structure

#### 2.0 Learning Objectives

#### 2.1 Introduction

#### 2.2 Basic concept of Entity-Relationship Diagram

##### 2.2.1 Symbols of entities

##### 2.2.2 Types of entities

##### 2.2.3 Types of attributes.

#### 2.3 Relationship concepts and its types

#### 2.4 Cardinalities

#### 2.5 Overview of Extended ER model

#### 2.6 Case Study of Extended ER model

#### 2.7 Let Us Sum Up

#### 2.8 Answers for Check Your Progress

#### 2.9 Glossary

#### 2.10 Assignment

#### 2.11 Activities

#### 2.12 Case Study

#### 2.13 Further Readings

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- The concept of Conceptual Modelling
- Knowledge about Entity-Relationship Diagram
- Basic of different types of entities Idea about Single and Multi value attribute

## 2.1 Introduction

The conceptual modelling is separated into three decades that is 1970s, 1980s as well as 1990s. In 1970s the database design was extremely significant as per Peter Chen's paper "Entity-Relationship Model that is Unified View of Data." This is highlighted in the area of data modelling as well as database design. Readily available discussions endeavour to expand high level data definition languages designed for defining conceptual schemas that can be Conceptual Schema Language (CSL). In 1980s, there sprang up a number of approaches to expand Chen's Entity Relationship Model. In fact information systems as well as the design of IS were considered attractive subjects. Next, in the start of 1990s, there were many queries such as schema integration, schema transformation as well as quality measures meant for conceptual schemas in region of database design. However this moment in time is also predisposed by object-oriented modelling methods in addition to languages in software engineering.

A conceptual model is a model prepared of the work of art of concepts, which are old to help people be acquainted with understanding or reproduce a subject that shows the model. A number of models comprising of physical objects that facilitate it together by making it work like the object it represents. A conceptual model recognizes the elevated relationships involving the unlike entities. Features of conceptual data model are:

- Vital entities as well as relations which is present.
- Having no particular attribute.
- Contains no main key.

The figure 2.1 is an example of a conceptual model.

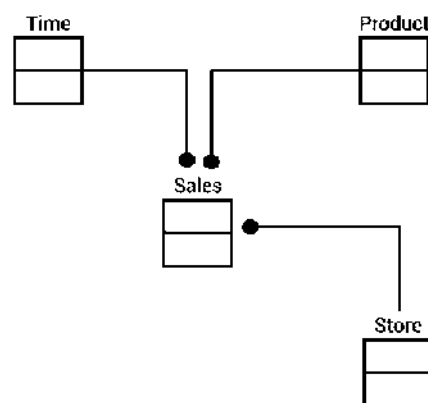


Fig 2.1 conceptual model

In fig 2.1 it is seen that the only information shown is by means of the conceptual modelling entities that explains the data as well as the relationships flanked by individual entities. No other information is exposed through the conceptual model.

## 2.2 Basic Concept of Entity-Relationship Diagram

An entity-relationship diagram, or ERD, is a type of chart which optically signifies the correlation involving database entities. The chart is an organizational data storage needs that carries three major mechanisms:

- Entities
- Attributes
- Relationships

This model chart will describe about the hypothetical dream of a database. This type of chart model will be comfortable in the region of real-world entities in addition to relationships surrounded by them. At the review stage, ER model is well thought-out as good alternative for designing databases.

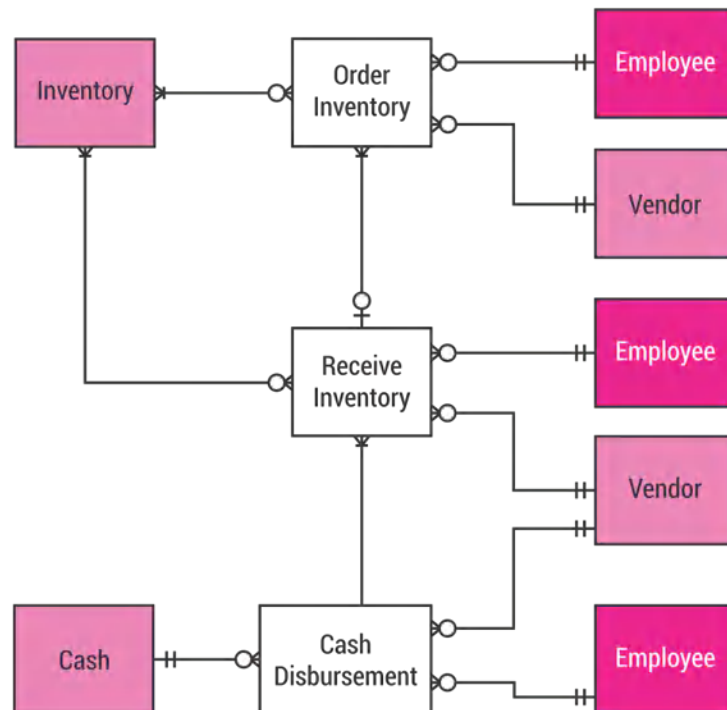


Fig 2.2 ERD model example

Entity-Relationship model works on the phenomena of actual-world entities as well as relationships that appear among them. At similar instance, preparing such actual-world condition in database models will lead to formation of entity set, relationship set, general attributes as well as constraints. It was established for theoretical design of database which relies on:

- Entities and their attributes.
- Relationships among entities.

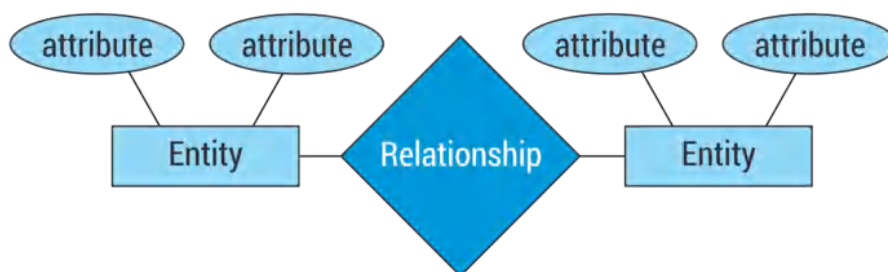


Fig 2.3 ER model

### 2.2.1 Symbols of entities

It seems that an entity is an actual-world entity that can be either animate or inanimate, but is identifiable. For instance, if we consider a school database, we see that students, teachers, classes as well as courses serves as entities. It is seen that every such entities carries various attributes or properties that give them their identity.

Moreover, it is found that an entity set is a mixture of similar types of entities. An entity set may perhaps hold entities with the characteristic sharing comparable values. For illustration, a set of students may comprise of each and every students of a school, similarly a set of Teachers will contain carryall teachers of school with all faculties. The Entity sets need not be disjoint.

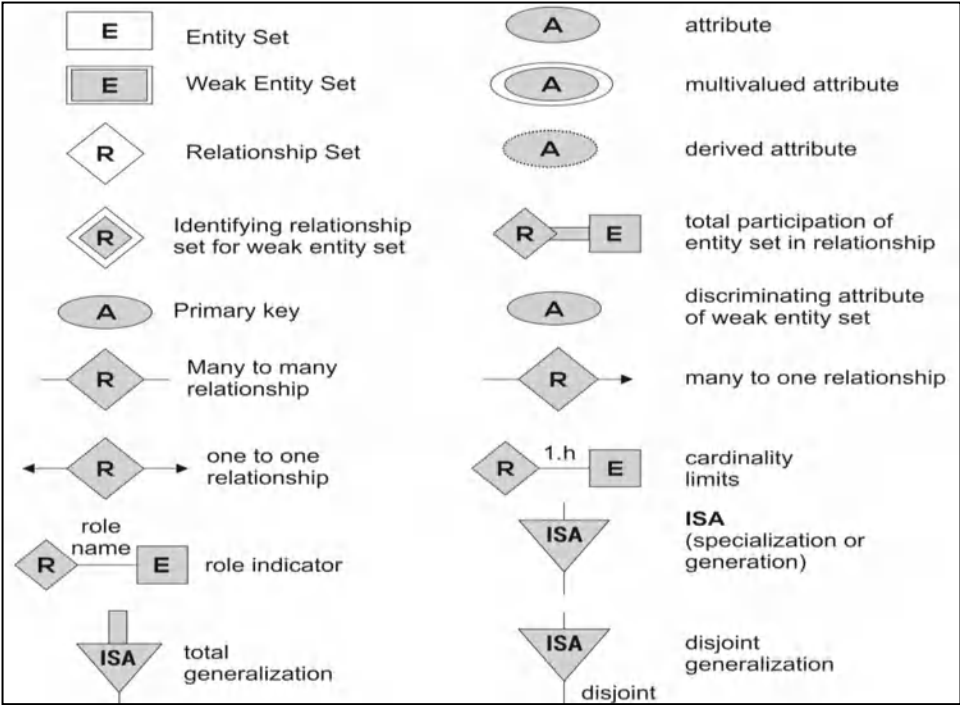


Fig 2.4 ER model symbol

2.2.2 Types of entities

Entity Relationship (ER) model carries various types of entities.

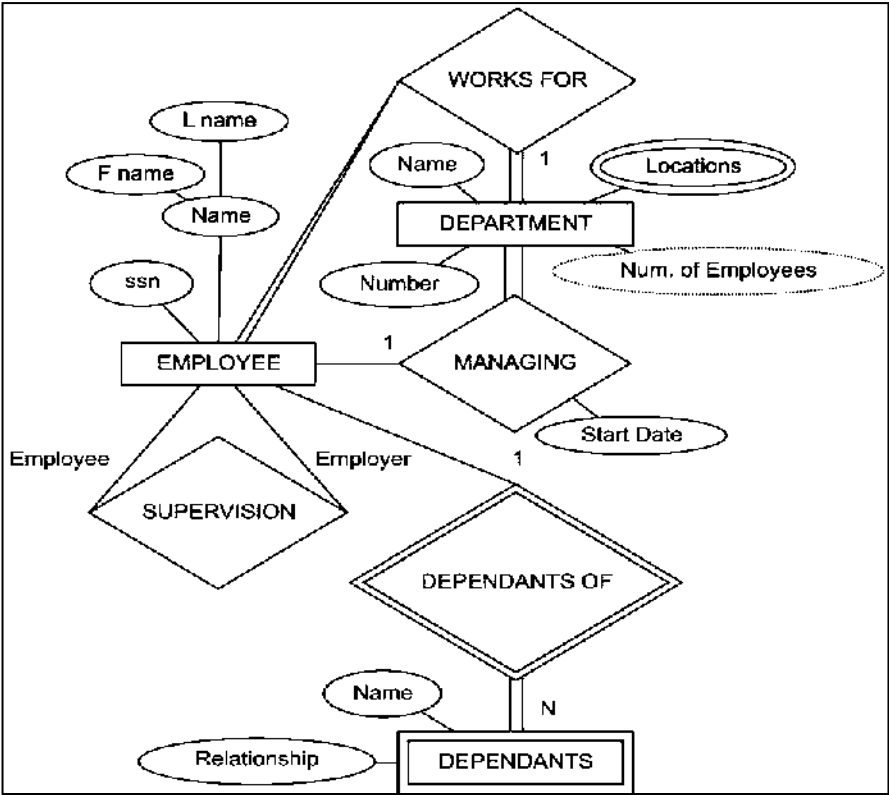


Fig 2.5 Entity in ER model



The continuation of an entity possibly will depend on the continuation of one or more previous entities, like existence dependent. It is that an entity whose continuation is not bound on every previous entity is called as not existence dependent entity. Entities which are marked as per their characteristics are:

- Strong Entities
- Weak Entities

### Strong Entity and Weak Entity

The entity set which does not encompass enough attributes to figure a primary key is termed as Weak entity set. The entity set which carries a primary key is named as Strong entity set.

It is seen that a weak entity is reality dependent. It has a survival of a weak entity that depends on the continuation of a classifying entity set. In this the partial key is applied to make out previous attributes of a weak entity set. The primary key of a weak entity set is created by primary key of recognizing entity set as well as the discriminator of weak entity set. It is studied that the presence of weak entity is shown by double rectangle in the ER diagram as shown in fig 2.6.

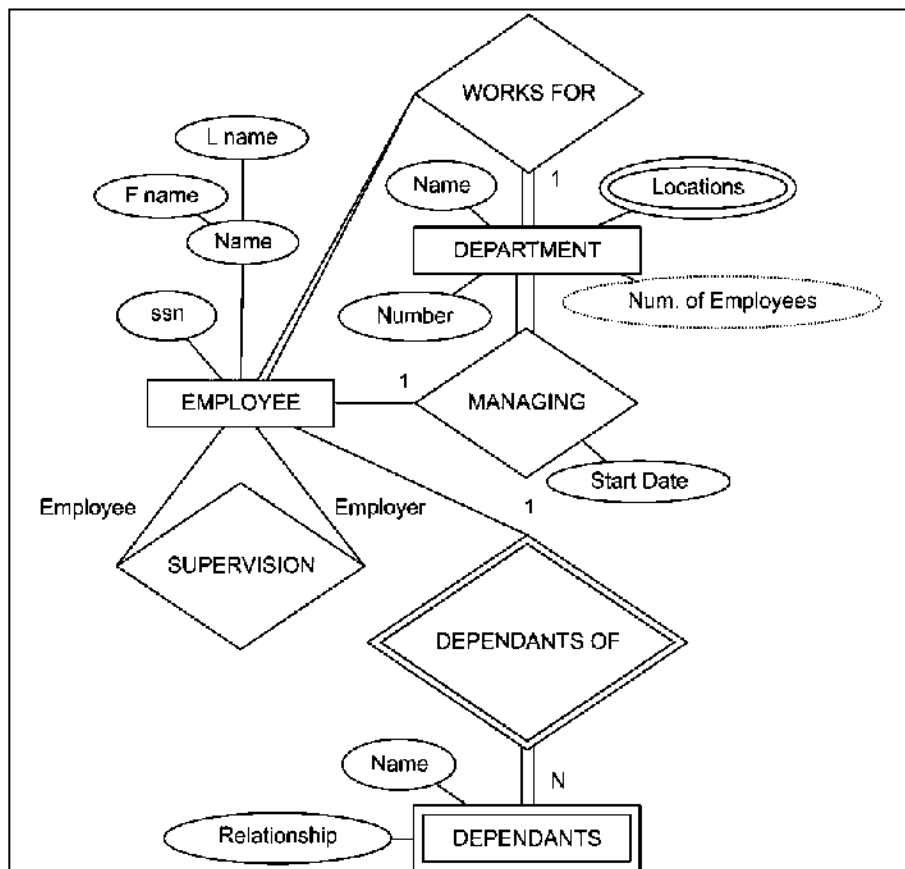


Fig 2.6 Entity in ER model

From the diagram 2.6, it is clear that the underlined is the discriminator of weak entity set that carries dashed line.

Further it is analysed that the relationship that exists among weak entity as well as strong entity set is termed as Identifying Relationship. For this consider diagram 1.8, which is related to a loan payment plan that identifies the relationship for payment entity. In this, a weak entity set is characterized by double outlined box in addition to equivalent categorized relation by a double outlined diamond shaped symbol.

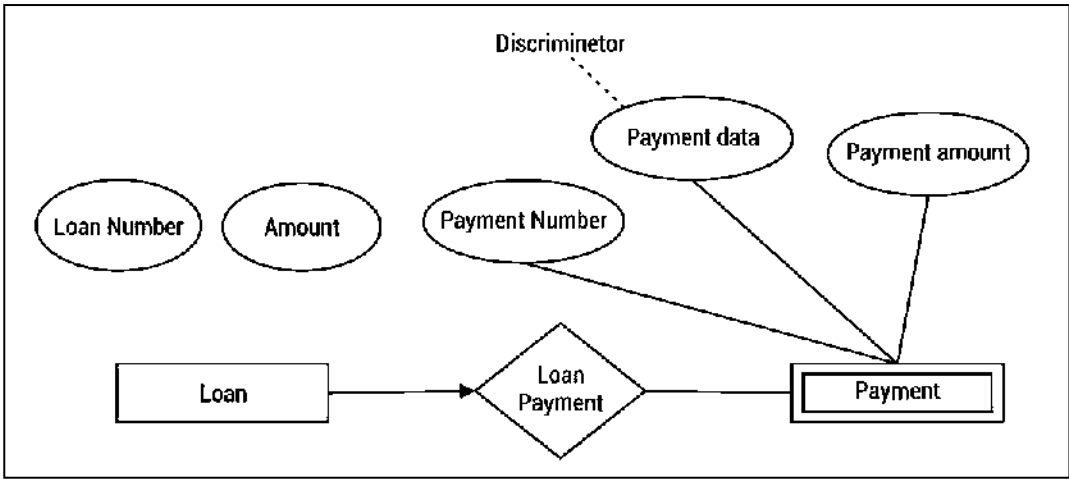


Fig 2.7 Example

In fig 2.7, we see that the double lines shows the complete participation of weak entity in strong entity place, which says that each payment be required to be connected by means of loan payment to various accounts. The arrow beginning from loan-payment in the direction of loan indicates shows the presence of single payment for each loan. In this, the discriminator of weak entity bunch is underlined with the help of dashed lines instead of bold line. The distinction between weak and strong entity is shown:

| Strong Entity Set                | Weak Entity Set                                                         |
|----------------------------------|-------------------------------------------------------------------------|
| It has its own primary Key       | It does not save sufficient attributes to form a primary key on its own |
| It is represented by a rectangle | It is represented by a double rectangle                                 |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>It contains a primary key represented by an underline</p> <p>The member of strong entity set is called as dominant entity set</p> <p>The Primary Key is one of its attributes which uniquely Identifies its member</p> <p>The relationship between two strong entity set is represent by a diamond symbol</p> <p>The line connecting strong entity set with the relationship is single</p> <p>Total participation in the relationship may or may not exist</p> | <p>It contains a Partial Key or discriminator represented by a dashed underline</p> <p>The member of weak entity set is called as subordinate entity set</p> <p>The Primary Key of weak entity set is a combination of partial Key and Primary Key of the strong entity set</p> <p>The relationship between one strong and a weak entity set is represented by a double diamond sign it is known as identifying relationship</p> <p>The line connecting weak entity set with the identifying relationship is double</p> <p>Total Participation in the identifying relationship always exists</p> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

### 2.2.3 Types of attributes

Entities are characterized by way of their properties that are known as attributes. Every attributes contains certain values.

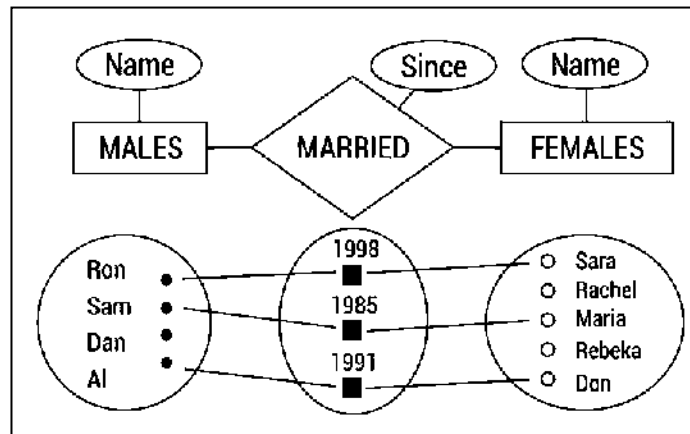


Fig 2.8 Arrangement of Attribute

From the figure, we found that a student entity having name, class as well as age serves as attributes. In such situation, there exists a domain or range of values to facilitate the attributes. In such case, the student's name cannot be a numeric value, but have to have an alphabetic; also the student's age should not be a negative number.

There are certain types of Attributes that exists in ER model:

**Simple attribute:** Simple attributes contain atomic values that cannot be separated more. In an example, a student's mobile number is an atomic value of 10 digits.

**Composite attribute:** Composite attributes is another type of attribute that are prepared from more than one simple attribute. In an example, if a student is writing his name, then the first name and last name constitute a composite attribute.

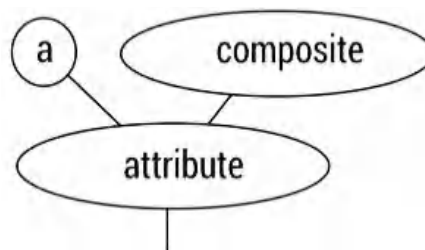


Fig 2.9 Composite attribute

**Derived attribute:** These are the attributes which do not present in physical database; however their values are copied from different attributes that are available in the database. In an example, an average salary in an organisation should not be kept directly in database, as an alternative, it should be derived.



Fig 2.10 Derived attribute

**Single-value attribute:** The Single value attributes carries only one value. It can be a Social Security Number.

**Multi-value attribute:** Multi-value attribute is a type of attribute that carries more than one value. It can be a person that has more than one mobile number, email address, etc.

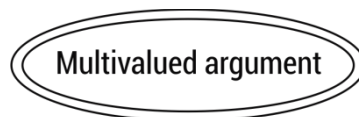


Fig 2.11 Multi value attributes

Such type of attribute can be:

- simple single-valued attributes
- simple multi-valued attributes
- composite single-valued attributes
- composite multi-valued attributes

### Check your progress 1

1. Which of the following are the properties of entities?
  - a. Groups
  - b. Table
  - c. Attributes
  - d. Switchboards
2. E-R model uses this symbol to represent weak entity set?
  - a. Dotted rectangle.
  - b. Diamond
  - c. Doubly outlined rectangle

- d. None of these
3. Conceptual design
- a. Is a documentation technique.
  - b. Needs data volume and processing frequencies to determine the size of the database.
  - c. Involves modelling independent of the DBMS.
  - d. Is designing the relational model.

---

## 2.3 Relationship concepts and its types

---

The level or degree of relationship is basically the number of entity types which takes part in the relationship. Since from the ER model, there are three on the whole common relationships that occurring ER modelling:

- Binary
- Unary
- Ternary

### Binary Relationship

This type of relationship occurs when two entities take part shows the familiar relationship degree. For Example:



Fig 2.12 Binary relationship

### Unary Relationship

This type of relationship exists when both participants in relationship behaves in similar manner. Consider fig 2.13.

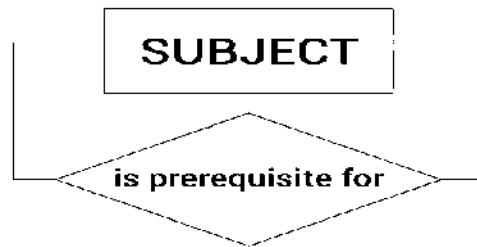


**Fig 2.13 Unary relationship**

In this, the subject gets prerequisites for different subjects.

### **Ternary Relationship**

This type of relationship appears when three entities take part in a common relationship. Consider example in fig 2.14.

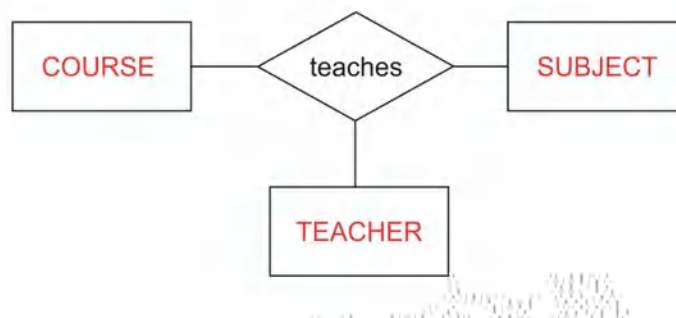


**Fig 2.14 Ternary relationship**

In the example above, it is seen that the University may further require the evidence that which teachers taught what particular subjects in which courses.

### **Check your progress 2**

1. Which relationship exists when the maintenance is between two entities?
  - a. Unary
  - b. Binary
  - c. Ternary
  - d. Quaternary

## 2.4 Cardinalities

The cardinality can be seen as a relationship that exists for a number of occurrences of entity B so as to associate with entity A. There is least cardinality as well as highest cardinality for each relationship; though an indefinite greatest cardinality is given away as N. Cardinality limits are more often than not derived from the organisations, policies or external constraints.

Consider a case at the college where every Teacher is able to teach an indefinite utmost number of subjects as lengthy as his/her weekly hours carry out that will not go beyond 24. Teachers possibly will teach 0 subjects but since they are occupied in non- teaching projects, so the cardinality limits for Teacher can be (0, N) as shown in fig 2.15.

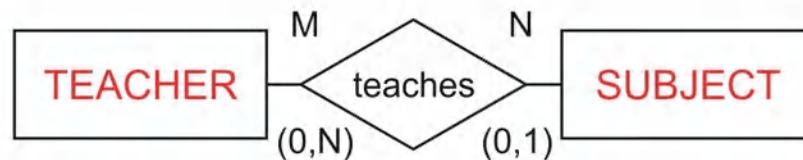


Fig 2.15 Cardinality limits for Teacher

As per the college policies, each Subject to be taught by only single teacher, further possibly, Subjects might not have allotted to a teacher, so in this case the cardinality limits for SUBJECT will be (0,1).

The connectivity of a relationship is its classification. It may be a one to one (1:1), one to many (1:M) or many to many (M:N) relationship. Relationships connectivity is represented by a 1, M or N next to the related entity.

### One to one (1:1)

In this, a Principal manages one Department, where every Department is managed by single Principal Teacher as described in fig 2.16



Fig 2.16 1:1 Cardinality of Principal Teacher



### One to many (1:M)

In this, a Subject can be given many times, but each contribution belongs to one Subject as shown in fig 2.17 of one to many relationships

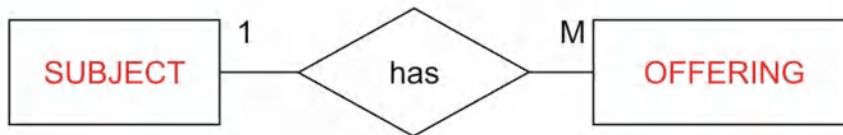


Fig 2.17 One to many relationships

### Many to many (M:N)

In this case, a Teacher can teach several different Subjects, but every Subject may be taught by several Teachers. This is explained in fig 2.18 in many to many relationships:



Fig 2.18 Many to many relationship

### Many-to-one

An entity in A is related to at most one entity in B, but an entity in B is related to any number of entities in A.

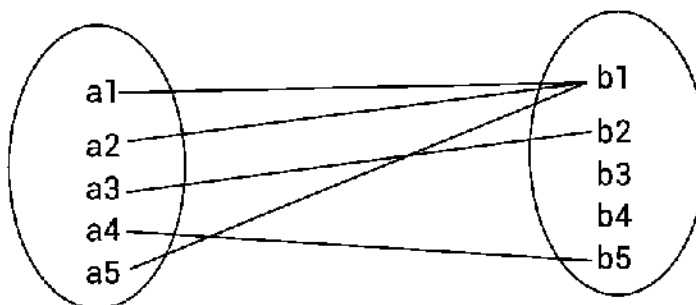


Fig 2.19 many to one relationship

### Check your progress 3

1. Which cardinality is explained from, “in a School, a teacher can teach many different Subjects, but every subject can be taught by many teachers?”

- |                 |                |
|-----------------|----------------|
| a. Many to many | c. Many to one |
| b. One to many  | d. One to one  |

---

## 2.5 Overview of Extended ER model

---

It is the process of assigning sub grouping inside an entity set and is commonly narrate as specialization. In such model, we apply IS for showing relationship having specialization. Is a type of relationship referred as super class-subclass relationship.

Example: Person IS an Employer

Person IS an Owner

Employee IS A General Manager

**Generalization:** In such extended model, the design process appears in bottom up manner. In this numerous entity sets are manufactured in a higher level entity set depending upon its features.

Example: Consumer as well as occupier entities can be conceptualized into a higher amplitude entity Person.

**Accredit inheritance:** Accredit of higher degree entity set are chosen by submerge level entity set.

**Aggregation:** Aggregation is a consideration where association sets are practising as higher degree entity sets. An association set is confined by entity set from inside because that alike entity sets conducted to endowment of associations.

**Specialization:** It is a mechanism which will explain about group of subclasses related to an entity type known as superclass. It explains based on special features that are available in entities in case of superclass. It contains varied specific variations of similar entity type with respect to certain special features as shown in fig 2.20. It is studied that subclass contains its own:

- Local attributes
- Particular relationship types

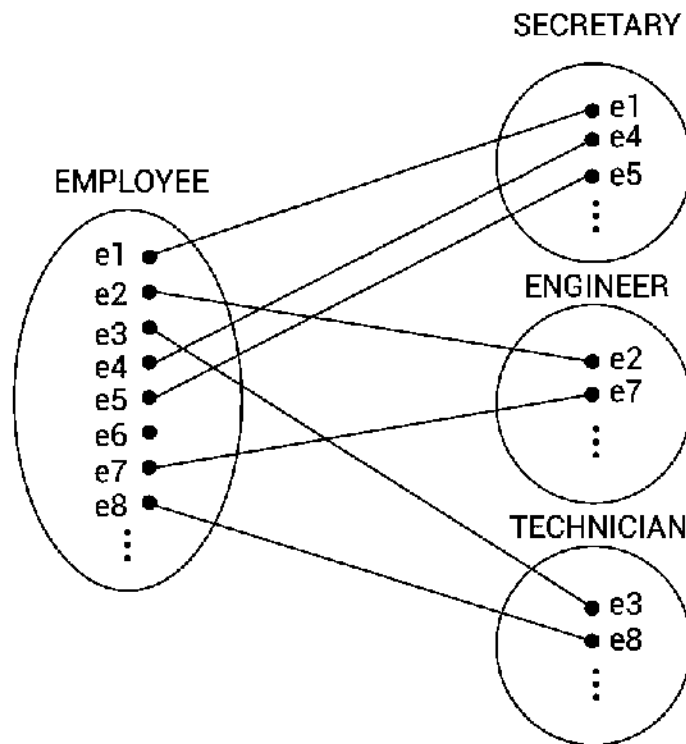


Fig 2.20 Instances of Specialization

#### Check your progress 4

1. In an Extended ER Model, relationship is part of:

- |                   |            |
|-------------------|------------|
| a. contribution   | c. concept |
| b. specialization | d. all     |

## 2.6 Case Study of Extended ER model

Enhanced Entity Relationship (EER) Model is high level data model which offers extension of original Entity Relationship (ER) model. EER Models contains several design details. The EER Modelling comes out as a result for modelling highly compound databases. Consider an Enhanced Entity Relationship model of the Dream Home as shown in fig 2.21

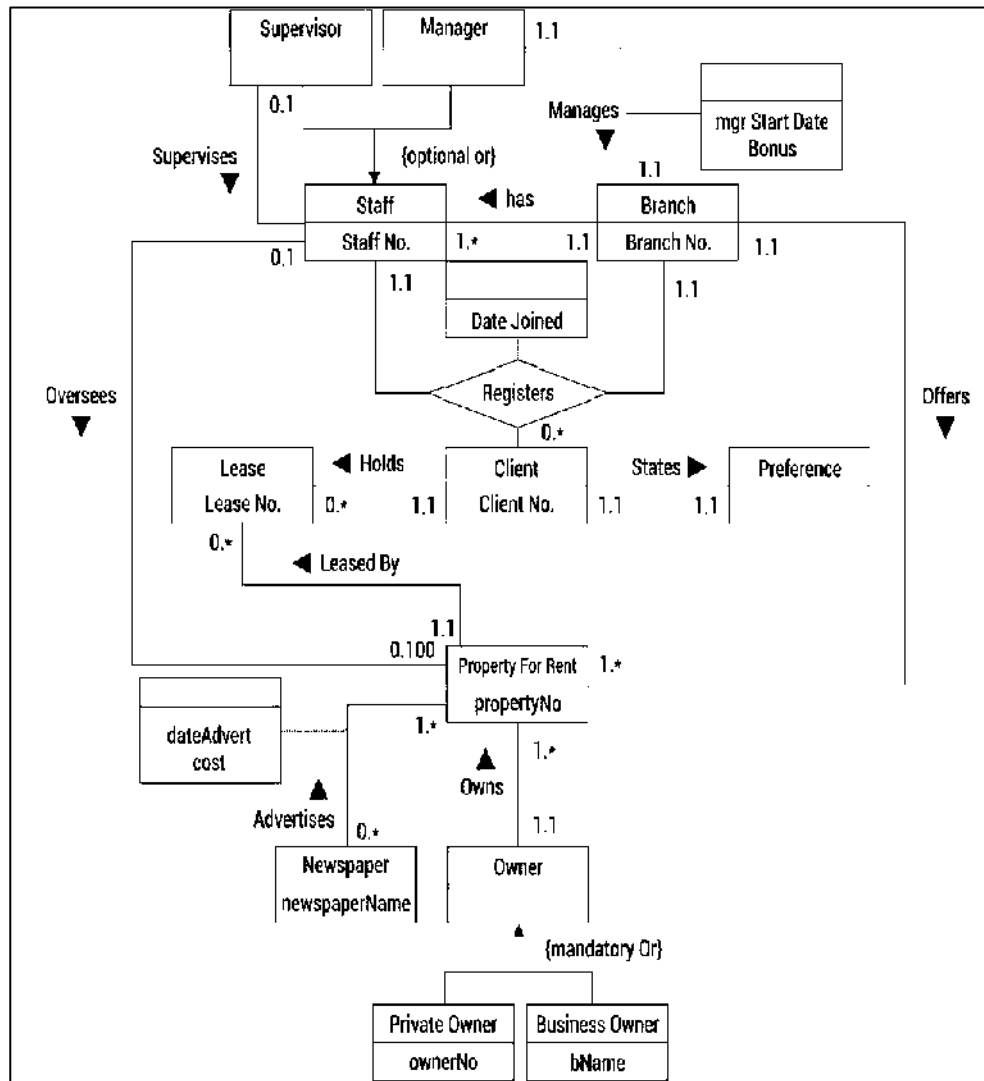


Fig 2.21Enhanced Entity Relationship model of Dream Home

### Check your progress 5

1. EER Modelling, comes with:

- a. low data model
- b. high data model
- c. complex data model
- d. all

## 2.7 Let Us Sum Up

In the above unit we have learned that a conceptual model is a model which is prepared from work of art of concepts, which are old to help people be acquainted with understanding or reproduce a subject that shows the model.

It is studied that an Entity Relationship model is such type of model which works on the phenomena of actual-world entities as well as relationships that appear among them. Similarly preparing a real world condition in database models will led to formation of entity set, relationship set, general attributes as well as constraints. In case of a conceptual model, the modelling is done base on the work of art of concepts. The Entity-relationship diagram is a chart which signifies correlation involving database entities. The entity set is collection of similar types of entities which carries a primary key is named as strong entity set

---

## 2.8 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-c), (2-c), (3-c)

**Check your progress 2**

**Answers:** (1-b)

**Check your progress 3**

**Answers:** (1-a)

**Check your progress 4**

**Answers:** (1-b)

**Check your progress 5**

**Answers:** (1-b)

---

## 2.9 Glossary

---

1. **Data Modelling** - A software engineering turns out to be more and more vital.
2. **Entity Relationship model** - It is a concept of real world entities and relationships among them.
3. **Entity set** - It is a collection of same types of entities.

---

## 2.10 Assignment

---

Is data modeling required for DBMS?

---

## 2.11 Activities

---

Is a unary and binary model having common features?

---

## 2.12 Case Study

---

Is one-one, one-many, many-one and many-many cardinalities are same?

---

## 2.13 Further Readings

---

1. Asher, H.B. (1984). Causal modelling. Sage University Paper series on quantitative applications in the social sciences, 07-003. Newbury Park, CA: Sage Publications.
2. Creswell, J.W. (1994). Research design: Qualitative and quantitative approaches. Thousand Oaks, CA: Sage Publications.
3. Kerlinger, F.N. (1979). Behavioural research: A conceptual approach. New York: Holt, Rinehart & Winston.

---

## **Block Summary**

---

The students have given the basic of database concepts and idea about conceptualisation. The user was given a description on topics related to Database domain, model, architecture and languages. The model and origination of rules and criteria's of ER model was also detailed. The block focuses on basic understanding of DBMS languages and ER concepts that will give the knowledge about cardinalities. The students or programmers will get benefit while reading this block as it gives shortcuts and related examples that will clear all doubts.

After reading this block the student will gather information on Database domain, model, architecture and languages with practice on certain rules and criteria's of ER model. The basic about advanced ER modelling is explained with conceptualisation of DBMS languages and ER concepts followed by different cardinalities.

---

## Block Assignment

---

### Short Answer Questions

1. What is Database Management System?
2. What is Entity-Relationship model?
3. What is an Entity set?
4. What is a hierarchical and network model?
5. Explain Cardinalities?

### Long Answer Questions

1. What is the basic concept of Entity-Relationship Diagram?
2. Write detail about Three-tier architecture of DBMS?
3. What are the advantages of DBMS over file systems?



**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                              | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|----------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| <b>Presentation Quality</b>                        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Language and Style</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Illustration used<br/>(Diagram, tables etc)</b> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Conceptual Clarity</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Check your progress<br/>Quest</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Feed back to CYP<br/>Question</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# FUNDAMENTALS OF OPERATING SYSTEM

PGDCA 104

## BLOCK 1: INTRODUCTION TO OPERATING SYSTEMS

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# FUNDAMENTALS OF OPERATING SYSTEM

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!





## FUNDAMENTALS OF OPERATING SYSTEM

### Contents

---

#### **BLOCK 1: INTRODUCTION TO OPERATING SYSTEMS**

##### **UNIT 1 BASICS OF OS**

Definition and Function of operating systems, Evolution of operating system, Operating system structure-monolithic layered, virtual machine and Client server

##### **UNIT 2 TYPES OF OPERATING SYSTEM**

Different types of operating system-real time systems, multi-user System, distributed system

##### **UNIT 3 BATCH OPERATING SYSTEM**

Introduction to basic terms and batch processing system: Jobs, Processes files, command interpreter

---

#### **BLOCK 2: MEMORY MANAGEMENT AND PROCESS SCHEDULING**

##### **UNIT 1 MEMORY MANAGEMENT**

Logical and Physical address protection, paging, and segmentation, Virtual memory, Page replacement algorithms, Cache memory, hierarchy of memory types, Associative memory

##### **UNIT 2 PROCESS SCHEDULING**

Process states, virtual processor, Interrupt mechanism, Scheduling algorithms Performance evaluation of scheduling algorithm, Threads



**BLOCK 3: FILE AND I/O MANAGEMENT**

**UNIT 1 FILE SYSTEM**

File systems-Partitions and Directory structure, Disk space allocation, Disk scheduling

**UNIT 2 I/O MANAGEMENT**

I/O Hardware, I/O Drivers, DMA controlled I/O and programmed I/O, I/O Supervisors

**BLOCK 4: BASICS OF DISTRIBUTED OPERATING SYSTEM**

**UNIT 1 DISTRIBUTED OPERATING SYSTEM**

Introduction and need for distributed OS, Architecture of Distributed OS, Models of distributed system

**UNIT 2 MORE ON OPERATING SYSTEM**

Remote procedure Calls, Distributed shared memory, Unix Operating System: Case Studies



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA 104

# **FUNDAMENTALS OF OPERATING SYSTEM**

---

## **BLOCK 1: INTRODUCTION TO OPERATING SYSTEM**

---

### **UNIT 1**

BASICS OF OS

### **UNIT 2**

TYPES OF OPERATING SYSTEM

### **UNIT 3**

BATCH OPERATING SYSTEM

---

# **BLOCK 1: INTRODUCTION TO OPERATING SYSTEMS**

---

## **Block Introduction**

An operating system is important software which makes the computer to run. It handles all the computer processes and runs the hardware. It makes you to communicate with computer without having command on its language. It is seen that your computer operating system manages all software and hardware functions. The main idea of operating system is to coordinate will all processes and links these processes with central processing unit (CPU), memory and storage.

In this block, we will detail about the basic of Operating System and different types of Operating System. The block will focus on the study and concept that led to explanation about Operating System structure. The students will give with the idea about Batch processing system.

In this block, the student will make to learn and understand about the basic of operating system and its functions. The student will be demonstrated practically and theoretically about different types of operating system used.

## **Block Objective**

**After learning this block, you will be able to understand:**

- About Operating system and its features
- Detailed about different types of O/S and their structure
- Knowledge about batch processing system

## **Block Structure**

**Unit 1: Basics of OS**

**Unit 2: Types of Operating System**

**Unit 3: Batch Operating System**

---

# **UNIT 1: BASICS OF OS**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Definition and Function of operating systems**
- 1.3 Evolution of operating system**
- 1.4 Operating system structure-monolithic layered**
- 1.5 Virtual machine and Client server**
- 1.6 Let Us Sum Up**
- 1.7 Answers for Check Your Progress**
- 1.8 Glossary**
- 1.9 Assignment**
- 1.10 Activities**
- 1.11 Case Study**
- 1.12 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- The various task of an Operating System
- Concept of application program
- Idea about Internal Parts of Operating System
- Idea about DOS Operating System
- Study about Client/Server program

---

## 1.1 Introduction

---

Earlier in 1960's, operating system is software which handles the hardware. Presently, we see operating system as set of programs that create the hardware to work. Generally, operating system is set of programs to facilitate controls of a computer. There are different types of operating systems as UNIX, MS-DOS, MS-Windows, Windows/NT, and VM.

Over protecting of computer engage software at numerous levels. We will distinguish kernel services, library services, as well as application-level services, all of which are division of an operating system. Processes run Applications, which are related together by means of libraries that carry out standard services. The kernel supports the development by providing a path to the peripheral devices. The kernel reacts to service calls as of the processes as well as interrupts from the devices. The centre of the operating system is the kernel, a organize program with the purpose to function in restricted state, act in response to interrupts from external devices as well as service requests along with traps from processes. In order to run Computer hardware, we require an Operating System that will be able to recognise all hardware components and enable us to work on it. In this unit, we will study about Operating system and its evolution along with its necessary role.

---

## 1.2 Definition And Function Of Operating Systems

---

An operating system also known as OS is a software program that enables the computer hardware to communicate and operate with the computer software. Operating systems perform basic tasks:

- Recognizing input from the keyboard
- Sending output to Monitor
- keeping track of files and directories
- Controlling peripheral such as disk drives and printers.

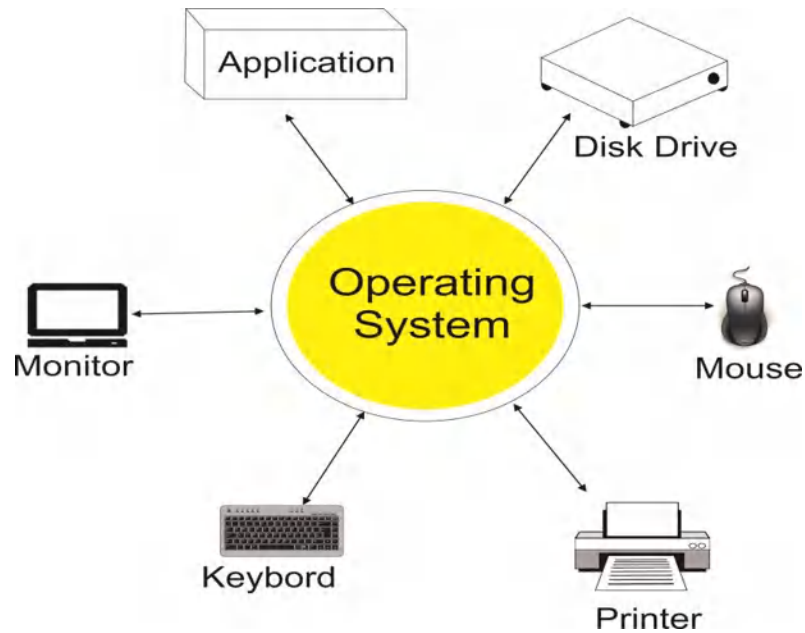


**Fig 1.1 Operating System with Computer hardware**

The operating system is system software that is stored on the storage device such as hard disk, CD-ROM or floppy disk. When a computer is switched on, the operating system is transferred from the storage device into main memory through ROM.

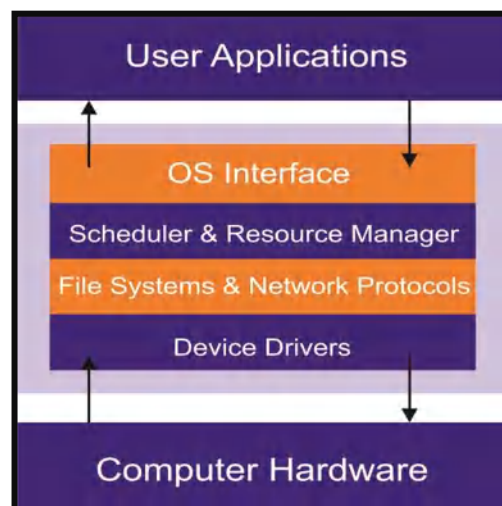


**Fig 1.2 Position of Operating System**

An operating system controls and coordinates the operations of the computer system. It manages the computer hardware, controls the execution of application programs and provides the set of services to the users. It acts as an interface between user and the computer. The users interact with the operating system indirectly through application program.

The work of the operating system involves:-

- Managing the processor
- Managing Random Access Memory
- Managing Input/output
- Managing execution of applications
- Managing Files
- Controlling Information management

### Parts of Operating System

#### i) Resident part-

It is called as kernel that contains critical functions. It is loaded inside the main memory during the booting. It performs various functions residing in the main memory.

#### ii) Non-resident part-

This part of operating system is loaded into main memory when required. It includes:

- Disk Operating System (DOS) developed by Microsoft.
- Operating System 2 (QS/2) developed by IBM.
- XENIX or ZENIX developed by Microsoft.
- WTNOWS developed by Microsoft
- WINDOWS- NE

### Check your progress 1

1. It is studied that an Operating System is a \_\_\_\_\_.
  - a. System software
  - b. Stores information on the storage device
  - c. Controls and coordinates the operations of the computer system
  - d. All of above



2. Which among the following is not a function of an operating system?
- a. recognize input from keyboard
  - b. shows output on monitor
  - c. loads keyboard
  - d. track of files

---

## 1.3 Evolution of Operating System

---

Initially, the computer utilises batch operating systems where batches of jobs are run without taking a break. These programs are punched into cards where the processing was performed by copied into tape. After finishing the first job, the computer would soon start with the next job on the tape.

Professional operators when interacted with computer found that users drop such jobs and finally returned to hold the result soon after running of particular job. It was quiet difficult for users as expensive computer were made to involve in such type of processing of jobs.

During late 1960s, invent of timesharing operating systems led to replacement of batch systems. Users when involved directly by way of printing terminal found that Western Electric Teletype shown was ok.

With this time sharing OS, many users shared the computer and then spent only a fraction of second on every job before moving to the next job. It is found that a fast computer will work for many user's jobs at the same time thereby making the illusion that they were full attentive while receiving such jobs.

Printing terminals found that the programs were set of characters or command line user interfaces (CLI) where user had to type responses in order to typed commands which led to scrolled down the instructions on paper.

During mid-1970, the personal computers allows pockets and Altair 8800 were initially used for commercial purposes for an individuals. In the start of 1975, the Altair was sold to hobbyists in kit form. It was without the operating system because it has only toggle switches and light emitting diodes which serves as input and output.

After sometimes, people started connected terminals and floppy disk drives to Altairs. During the year 1976, Digital Research introduced CP/M operating

system for such Computer. CP/M and later on DOS had CLIs which were similar to timeshared operating systems where computer was only for a particular user.

With the success of Apple Macintosh in 1984, the particular system pushed the state of hardware art which were restricted to small with black and white display. As hardware continued to develop, many colour Macs were under developed position and soon Microsoft introduced Windows as its GUI operating system.

It was found that the Macintosh operating system was based on decades of research on graphically-oriented personal computer operating systems and applications. Computer applications today require a single machine to perform many operations and the applications may compete for the resources of the machine. This demands a high degree of coordination which can be handled by system software known as an operating system

The internal part of the OS is often called the kernel which comprises of:

- File Manager
- Device Drivers
- Memory Manager
- Scheduler
- Dispatcher

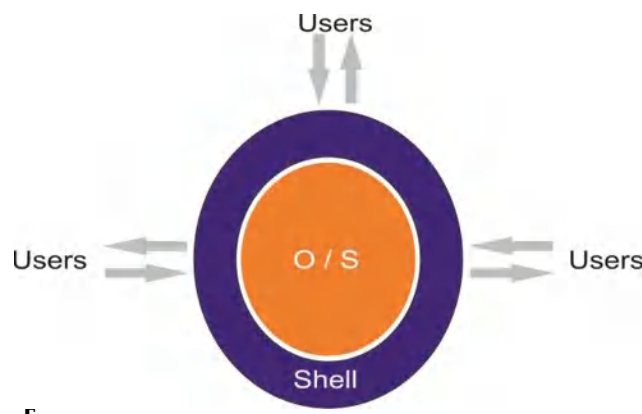


**Fig 1.3 Interfaces of OS**

### **Check your progress 2**

1. The commercial computer Altairs was developed in the year:
  - a. 1980
  - b. 1970
  - c. 1985
  - d. 1955

---

## 1.4 Operating system structure-monolithic layered

---

The mean of operating system architecture usually follows the leave-taking of particular principle. Such principle guide to re-structure the operating system mainly into relatively independent parts that can be easily provide basic independent features by keeping complicated designs in manageable conditions.

Apart from controlling complexity, the architecture of operating system influences key features that are in terms of robustness or efficiency as:

- The OS receives importance which allows to work if not then protected resources like physical devices or application memory. With such importance, the various related parts of OS or OS as a whole will be both accidental and malicious privileges misuse gets lowered.
- By breaking OS into different parts will led to adverse effect on efficiency since the overhead linked with communication among individual parts be exacerbated when coupled with hardware mechanisms?

### Monolithic Systems

Aboriginal concept of the operating system arrangement brings about no definite accommodation for the discriminating nature of the operating system. Furthermore the concept follows the separation of concerns; no action is acted to limit the blessings granted to the single parts of the operating system. The complete operating system acts with maximum approvals. The communication overhead inside the basic operating system is the identical as the communication overhead inside numerous other software, considered relatively low.

It is seen that CP/M and DOS are examples of monolithic operating systems that share common address space with certain applications. It is found in CP/M, 16 bit address space will begins with system variables along with application area additionally ends with 3 parts of O/S which are known as:

- CCP or Console Command Processor
- BDOS or Basic Disk Operating System
- BIOS or Basic Input/output System

If we see that in a DOS Operating System, there exists a 20 bit address space that begins with an array of interrupt vectors along with system variables that are followed by local DOS and its application area which will end with memory block utilised by video card and BIOS as shown in fig 1.4.

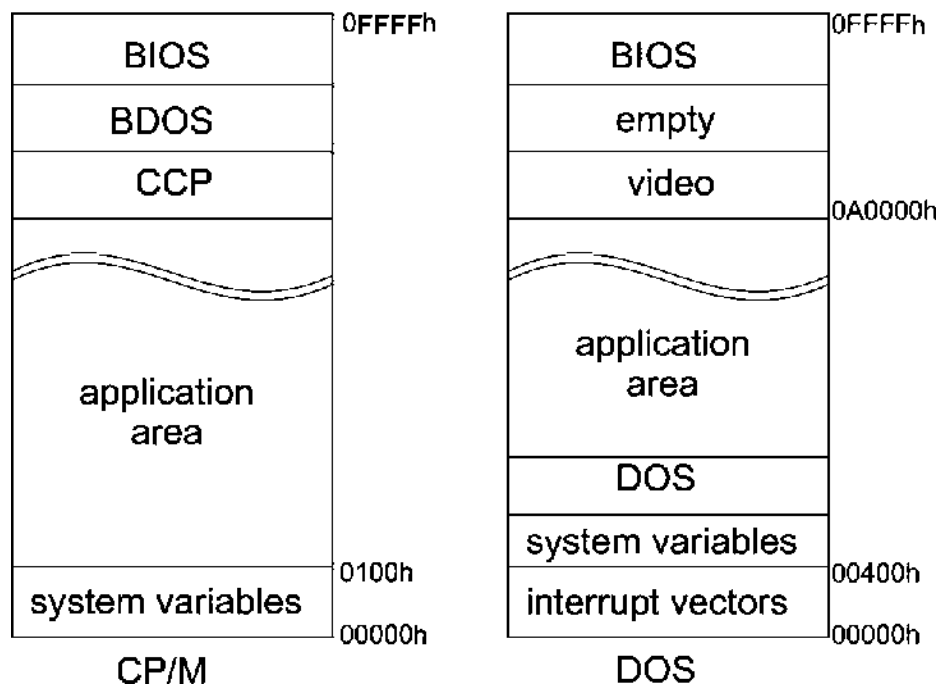


Fig 1.4: Monolithic Operating Systems

### Check your progress 3

1. Which is not a part of Operating System?

- |         |         |
|---------|---------|
| a. CCP  | c. BIOS |
| b. BDOS | d. DOS  |

## 1.5 Virtual machine and Client server

### Virtual machine

A virtual machine (VM) abides an operating system OS or conduct environment that is embedded on software which copies consecrated hardware. The end user embraces the equivalent experience on a virtual machine as they would acquire on dedicated hardware.

Individualized software designated a hypervisor copies the PC client or server's CPU, memory, hard disk and network as well as other hardware resources collectively, allowing virtual appliances to participate the resources. The hypervisor can copy integral virtual hardware platforms that are occasional from each other, assigning virtual machines to run Linux as well as Windows server operating systems on the identical underlying physical aggregation. Virtualization conserves costs by depreciating the need for physical hardware systems. Virtual machines additional desirably use hardware, which lowers the quantities of

hardware as well as associated maintenance costs, along with reduces power furthermore cooling demand. They also allay management due to virtual hardware does not collapse. Administrators can take advantage of virtual circumstances to simplify backups, disaster recovery, new deployments as well as elementary system administration tasks.

Virtual machines do not constrain distinguished hypervisor-specific hardware. Virtualization appears although require more bandwidth, storage along with processing capacity than a conventional server or desktop if the physical hardware is going to host multiple running virtual machines. VMs can easily move, be copied and reassigned between host servers to optimize hardware resource utilization. Because VMs on a physical host can consume unequal resource quantities (one may hog the available physical storage while another stores little), IT professionals must balance VMs with available resources.

### **Client server**

Client/server is a program relationship in which one program (the client) requests a service or resource from another program (the server).

It is seen that in client/server model, the programs are used by single computer only. It serves as an important concept for networking. Here, the client makes a connection with the server through local area network (LAN) or wide-area network (WAN) like Internet. After clearing the client's request, the connection gets terminated. In this case the Web browser serves as a client program which further appeals for a service from the server. The service and resource of the server will show the delivery of such Web page.

Computer assignments in which the server accomplishes a request created by a client are very customary furthermore the client/server model has served one of the main concepts of network computing. Most business approaches facilitate the client/server model as appears acts the Internet's core program, TCP/IP. For exemplary, when you examine your bank account from your computer, a client approximation in your computer overtures a request to a server program at the bank. That program may in twist forward an approach to its own client program, which that time conveys, a request to a database server at another bank computer. Once your account balance sheet has been acquired from the database, it is acknowledged back to the bank data client, which in turn applies it back to the client in your personal computer, which that time displays the information to you.

Both client programs as well as server programs are usual constituent of a larger program or application. On account of multiple client programs participate

the services of the equivalent server program, a special server identified a daemon may be charged due to anticipate client requests. In marketing, the client/server had been once used to differentiate allocated computing by personal computers (PCs) from the monolithic, concentrated computing model exercised by mainframes. This differentiation has largely evaporated, although, as mainframes along with their applications possess additionally turned to the client/server model further become part of network computing.

### Check your progress 4

1. Virtual machine can run:
  - a. Windows
  - b. Linux
  - c. DOS
  - d. all
2. In a Client/server program:
  - a. one program requests a service from another program
  - b. one program requests a copy to another program
  - c. one program requests a hardware to run another program
  - d. one program requests a software to load operating system

---

## 1.6 Let Us Sum Up

---

### In this unit, we have learned:

- That an operating system or OS is software program that enables the computer hardware to communicate and operate with the computer software.
- We see that there are many functions of an operating system which will help in managing:
  1. Processor.
  2. Random Access Memory:
  3. Input/output
  4. Execution of applications

5. Files
6. Information management

- A virtual machine (VM) abides an operating system OS or conduct environment that is embedded on software which copies consecrated hardware.
- Client/server is a program relationship in which one program (the client) requests a service or resource from another program (the server).

---

## 1.7 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-a), (2-c)

**Check your progress 2**

**Answers:** (1-b)

**Check your progress 3**

**Answers:** (1-d)

**Check your progress 4**

**Answers:** (1-a), (2-a)

---

## 1.8 Glossary

---

1. **Shell** - It makes interface with user
2. **File Manager** - It manages mass memory
3. **Device Drivers** - These carry different drives for various peripherals
4. **Memory Manager** - It handles the main memory
5. **Scheduler and Dispatcher** - It helps in managing the processes

---

## 1.9 Assignment

---

Write note on Client Operating System?

---

## **1.10 Activities**

---

Basics of OS

Establish a Client System in Linux Operating System.

---

## **1.11 Case Study**

---

Can an Operating System be able to handle all Computer Hardware?

---

## **1.12 Further Readings**

---

1. The Operating system by Andrew Tannenbaum.
2. Operating System by Mach.



---

## **UNIT 2: Types of Operating System**

---

### **Unit Structure**

#### **2.0 Learning Objectives**

#### **2.1 Introduction**

#### **2.2 Different types of operating system**

##### 2.2.1 Real time Systems

##### 2.2.2 Multi-user System

##### 2.2.3 Distributed system.

#### **2.3 Let Us Sum Up**

#### **2.4 Answers for Check Your Progress**

#### **2.5 Glossary**

#### **2.6 Assignment**

#### **2.7 Activities**

#### **2.8 Case Study**

#### **2.9 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Various types of Operating System
- About Batch processing system
- Concept of Multi-user System
- Distributed System

---

### **2.1 Introduction**

---

There are abundant Operating Systems those monopolize be constructed for functioning the performances those are demanded by the user. There are ample Operating Systems which acquire the ability to behave the entreaties those are acquired from the approach. The Operating system can behave a unique Operation

furthermore also multiple movements at duration. Hence there are numerous categories of Operating systems those are arranged by utilizing their Working mechanisms.

---

## 2.2 Different Types of Operating System

---

There are many types of operating system such as:

- 1) Serial Processing
- 2) Batch Processing
- 3) Multi-Programming
- 4) Real Time System
- 5) Distributed Operating System
- 6) Multiprocessing
- 7) Parallel operating systems

### 2.2.1 Real time Systems

There occurs additionally an Operating System which is comprehended as Real Time Processing System. In this acknowledgment duration is already adjusted. Indicates duration to show the after-effects after acquiring has adjusted by the Processor or CPU. Real Time System is exercised at those areas in which we binds higher along with well-timed return. These categories of approaches are exercised in reservation. Hence when we discriminate the demand, the CPU will conduct at that duration. There are two Types of Real Time System:

- **Hard Real Time System:** In the Hard Real Time System, Time is fixed and we can't Change any Moments of the Time of Processing. Means CPU will Process the data as we Enters the Data.
- **Soft Real Time System:** In the Soft Real Time System, some Moments can be Change. Means after giving the Command to the CPU, CPU Performs the Operation after a Microsecond.

### **2.2.2 Multi-user System**

As we comprehend that in the Batch Processing System there are multiple jobs appear by the System. The System foremost compose a batch furthermore later that he will accomplish all the jobs those are saved into the Batch. Furthermore the innermost difficulty is that if a mechanism or job needs an Input as well as Output Operation, that time it is not achievable and second there will be the wastage of the duration when we are composing the batch as well as the CPU will continue idle at that duration.

Although with the help of Multi programming we can achieve Multiple Programs on the System at a duration besides in the Multi-programming the CPU determination never get idle, hence with the help of Multi-Programming we can achieve ample algorithms on the System including When we are functioning with the Program that time we can additionally acknowledge the supplement or Another Program for sprinting additionally the CPU will that time behave the secondary Program following the completion of the original Program. Additionally in this we can further differentiate our Input means a user can additionally interact with the System.

The Multi-programming Operating Systems never utilize numerous cards on account of the approach is accessed on the Spot by the user. But the Operating System also utilizes the Process of Allocation and De-allocation of the Memory Means he will provide the Memory Space to all the Running and all the Waiting Processes. There must be the Proper Management of all the Running Jobs.

### **2.2.3 Distributed system**

Distributed Means Data is Stored and Processed on Multiple Locations. When a Data is stored on to the Multiple Computers, those are placed in Different Locations. Distributed means In the Network, Network Collections of Computers are connected with Each other.

Then if we want to Take Some Data from other Computer, Then we use the Distributed Processing System. And we can also Insert and Remove the Data from out Location to another Location. In this Data is shared between many users. And we can also Access all the Input and Output Devices are also accessed by Multiple Users.

### Check your progress 1

1. In Hard Real Time System, Time:
  - a. varies
  - b. fixed
  - c. zero
  - d. none of these
2. Real Time System is used when we require:
  - a. delay time return
  - b. no time constrain
  - c. timely return
  - d. none of these

---

## 2.3 Let Us Sum Up

---

In this unit, we have learned:

- About Different Types of Operating System
  1. Real Time System is an Operating system which works to achieve timely return
  2. Soft Real Time System is a part of Real Time O/S where each moment changes
  3. Multi programming is a programming technique where many programs perform

---

## 2.4 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-b), (2-c)

---

## 2.5 Glossary

---

1. **Real Time System** - It is an Operating system which is exercised to get timely return
2. **Soft Real Time System** - It is a type of Real Time O/S where each moment changes

3. **Multi programming** - it a programming techniques where many programs perform

---

## 2.6 Assignment

---

Write note on Batch processing Operating System?

---

## 2.7 Activities

---

Explain the cycle of operation of Real Time Operating System

---

## 2.8 Case Study

---

Can a multiuser Operating System be installed on the server?

---

## 2.9 Further Readings

---

1. The Operating system by Andrew Tannenbaum.
2. Operating System by Mach.

---

## **UNIT 3: BATCH OPERATING SYSTEM**

---

### **Unit Structure**

#### **3.0 Learning Objectives**

#### **3.1 Introduction**

#### **3.2 Basic terms**

#### **3.3 Batch processing system**

##### **3.3.1 Jobs**

##### **3.3.2 Processes files**

##### **3.3.3 Command interpreter**

#### **3.4 Let Us Sum Up**

#### **3.5 Answers for Check Your Progress**

#### **3.6 Glossary**

#### **3.7 Assignment**

#### **3.8 Activities**

#### **3.9 Case Study**

#### **3.10 Further Readings**

---

### **3.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- The concept of Batch
- Concept about Batch Processing
- Idea about File Processing

---

## 3.1 Introduction

---

Batch is the term which is given to the work of doing similar jobs continuously again and again but with a difference as in this the input data is shown for every iteration of the job and probably the output file.

Batch operating system is a kind of operating system requirement which mainly involves in mainframe computer that was used with the intention that it may perform great repetitive data processing work. It is analysed that a mainframe set is used to process 30 million pension statements which belongs to individual customers.

It is found that a batch job requires no intervention by a person once the initial commands are set up. Setting up a batch job is similar to filling out a form, with specific details required to be shown.

---

## 3.2 Basic terms

---

### Databases

Batch processing is additionally applied for capable bigness database updates as well as automated transaction processing, as compared to collaborative online transaction processing (OLTP) approaches. The extract, transform, load (ETL) extent in populating data warehouses is inherently a batch approach in most implementations.

### Images

Batch processing is often used to perform various operations with digital images such as resize, convert, watermark, or otherwise edit image files.

### Conversions

Batch processing may additionally be applied for altering computer files from one format to another. For exemplary a batch job may change proprietary as well as legacy files to conventional standard formats for end-user queries along with display.

### Batch window

A batch window continues “a duration of less-intensive online assignment”, when the computer system is accomplished to plunge batch jobs without obstruction from online systems.

Numerous untimely computer systems granted sole batch processing; hence jobs could be plunge any time within a 24-hour day. With the accomplishment of assignment processing the online approaches might singular is required from 9:00 a.m. to 5:00 p.m., abstracting dual shifts obtainable for batch work, in this casket the batch window would be sixteen hours. The difficulty is not normally that the computer system is inadequate of admitting combined online along with batch work, although that the batch systems normally constrain approach to data in an integrated state, released from online updates until the batch processing is complete.

In a bank, for exemplary, so-called end-of-day (EOD) jobs constitute interest calculation, duration of reports as well as data sets to disjoint systems, printing statements, as well as payment processing.

**Check your progress 1**

1. Which is not a batch job?
  - a. Priority of the Job
  - b. Uses CPU hogging
  - c. Avoids infinite printouts
  - d. Data details for input and output

---

### 3.3 Batch processing system

---

A batch processing approach is one where facts are assorted together in a batch preceding processing begins. A batch procedure behaves as an account of commands in arrangement. It be scramble by a computer's operating system facilitating a script or batch file, or may be accomplished within a system utilizing a macro or inner scripting tool. The mechanism of data entry for premature computers existed facilitating punched cards, which were experienced in batches, further on account of the term batch processing. Each bit of work for a batch processing system is designated a job.

Jobs are assigning up so they can race to accomplishment without human intercommunication. All entryway parameters are predefined through scripts, command-line arguments, control files, or job control jargon. This is in contradicting to "online" or to-and-fro programs which advice the user for similar



input. A program acquires a portion of data files as input, processes the data, as well as brings about a set of output facts files. This operating arrangement is identified as “batch processing” since the input data are acquired into batches or sets of records as well as each batch endures processed as a unit. The output exists another batch that can be reused for assessment.

Batch processing has been affiliated with mainframe computers owing to the earliest decades of electronic computing in the 1950s. There was a multifariousness of reasons why batch processing commanded premature computing. One logic is that the foremost bustling business problems for analyses of profitability as well as competitiveness were initially accounting problems, like as billing. Billing may effectively be appeared as a batch-oriented business process, along with appropriately every business constraining bill, reliably as well as on-time. Furthermore, every computing resource had been costly; hence consecutive submission of batch jobs on punched cards matched the resource constraints as well as technology evolution at the duration. Later, interactive sessions with coupled text-based computer terminal interfaces or graphical user interfaces became additional common. Furthermore, computers originally were not even cogent of having multiple programs loaded into the main memory.

Batch processing is still pervasive in mainframe computing, but practically all types of computers are now capable of at least some batch processing, even if only for “housekeeping” tasks. These include UNIX-based computers, Microsoft Windows, Mac OS X and even Smartphone’s. Increasingly, as computing in general becomes more pervasive batch processing is unlikely to lose its significance.

Batch approaches are still fault-finding in maximum organizations in big part on account of many common business processes are adaptable to batch processing. While online systems can additionally function when manual facilitation is not expected, they are not definitely optimized to harmonize high-volume, consecutive tasks. hence, even new systems commonly contain one or more batch approaches for updating information at the accomplishment of the day, generating reports, printing documents, as well as other non-interactive efforts that inevitable fulfil reliably within assured business deadlines.

Modern batch applications make utilize of modern batch architectures like as Jem the Bee or Spring Batch, which is composed for Java, as well as irrelevance frameworks for external programming languages, to deliver the defect tolerance as well as scalability necessary for high-volume processing. In steadiness to promise high-speed processing, batch applications are habitual

integrated with grid computing solutions to measure a batch job above a large number of processors; however there are relevant programming conflicts in doing so? High volume batch processing grounds particularly heavy demands on system along with application architectures as well. Architectures that feature energetic input/output performance as well as vertical scalability, along with modern mainframe computers, tend to cater better batch performance than alternatives.

Batch processing is most suitable for tasks where a large amount of data has to be processed on a regular basis.

### **Examples**

- A. payroll systems
- B. examination report card systems

### **Advantages**

- Once the data are submitted for processing, the computer may be left running without human interaction.
- The computer is only used for a certain period of time for the batch job.
- Jobs can be scheduled for a time when the computer is not busy.

### **Disadvantages**

- There is always a delay before work is processed and returned.
- Batch processing usually involves an expensive computer and a large number of trained staff.

## **3.3.1 Jobs**

In batch processing, job contains relevantly common group of processing along with calculation actions that utilizes small or very less cooperation among you along with the computer system. When a batch job is acknowledged, that time the job will primarily enter in a job queue where it will functionally halt till the system captures ready to process the next job. Here the system formed off its processing mechanism of job when it acquires the job from the job sequence. A batch job is put in a job queue by:

- Choosing a menu option that submits a batch job
- Submitting a job into the system using the SBMJOB command

It is found that a job queue carries several work or jobs which are halted for the system to process them. Your job waits while the system processes other jobs that other users submit prior to your job or have a higher priority. When system resources are available, the system processes your job.

### **3.3.2 Processes files**

A conventional activity is to conduct a set of equivalent operations on data sets in a group of files. This endures batch processing. For exemplary, you might desire to read multiple Excel files from a plate reader approximate concentrations from a common curve, fit a four parameter logistic application to duplicate data in each file, construct a graph of the data besides fit, along with export assured fit parameters to an Excel results file. On account of the automation language in Sigma Plot is transcribed for Visual Basic that time implementing a batch processing activity is purely a factor of writing a Sigma Plot macro or a Visual Basic program.

An exceptional exemplary of a batch processing program is the Batch Process Excel Files macro in Sigma Plot. The dialog from this macro is demonstrated. It is considered as a basic design around which you can construct your own macro to appear a definite task. It approves you select a group of Excel files by clicking on the Add File button that invokes the file open argument, fit an equation to a data domain in the Excel file that you discriminate as well as bring about a graph along with report of the results. In this casket the results for each file are accredited in a distinct section of a Sigma Plot notebook. This can be adjusted to establish the results in an Excel file if you desire.

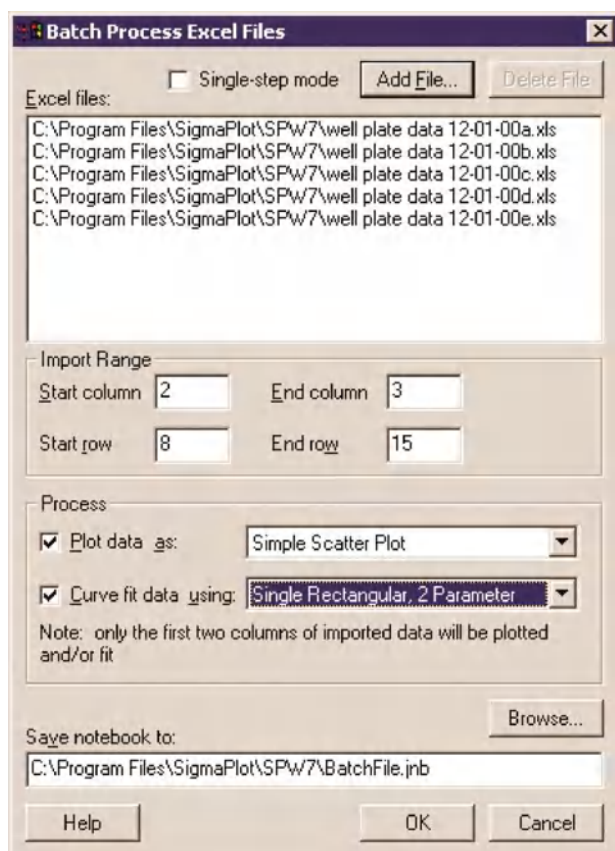


**Fig 3.1 Dialog for Batch Process Excel Files macro.**

Each of the Excel files from a well plate reader observes prefer the one demonstrated in Figure 3.2. Five duplicate measurements of definite binding are demonstrated in columns C through G. For tutorial approaches, the liberated radio ligand entrancement has been accumulated in column B. The macro has been written to appropriate an equation to two columns of data so for this example we will ignore the replicates. It is at ease to change the macro to encompass the row wise duplicate format in the curve fit.

| well plate data 12-01-00a.xls |               |          |            |       |       |       |       |   |
|-------------------------------|---------------|----------|------------|-------|-------|-------|-------|---|
|                               | A             | B        | C          | D     | E     | F     | G     | H |
| 1                             |               |          |            |       |       |       |       |   |
| 2                             | Machine Name: | X123     | User Name: | SPSS  |       |       |       |   |
| 3                             | Protocol:     | Gen1     |            |       |       |       |       |   |
| 4                             | Comments:     | demo     |            |       |       |       |       |   |
| 5                             | Plate Number: | 2        |            |       |       |       |       |   |
| 6                             | Time:         | 12:01:00 | 1/23/2001  |       |       |       |       |   |
| 7                             |               |          | 1          | 2     | 3     | 4     | 5     | 6 |
| 8                             | A             | 5        | 34.81      | 29.93 | 38.49 | 26.69 | 33.42 |   |
| 9                             | B             | 10       | 49.97      | 51.03 | 51.88 | 48.17 | 47.79 |   |
| 10                            | C             | 30       | 76.75      | 79.18 | 75.57 | 72.73 | 74.47 |   |
| 11                            | D             | 50       | 89.03      | 80.75 | 84.79 | 87.36 | 84.52 |   |
| 12                            | E             | 70       | 93.09      | 90.10 | 88.60 | 84.93 | 90.65 |   |
| 13                            | F             | 90       | 97.95      | 84.49 | 80.35 | 93.47 | 91.53 |   |
| 14                            | G             | 110      | 90.49      | 94.15 | 97.24 | 94.07 | 91.81 |   |
| 15                            | H             | 130      | 94.84      | 90.13 | 92.34 | 98.10 | 94.38 |   |
| 16                            |               |          |            |       |       |       |       |   |

**Fig 3.2 one of the excel files to analyse.**

You can then select the appropriate region of the Excel file containing the data to fit. This is shown in Figure 3.3 for the data in Figure 3.2.

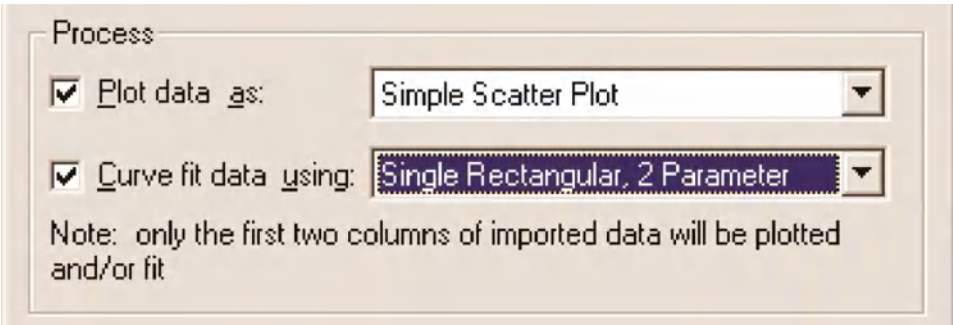


Fig 3.3 data in Figure

The distorted function is chosen to fit each data set additionally a simple scatter plot continues used to demonstrate the results. Note that every equation in the Sigma Plot curve suited library is accounted in the dropdown box in Figure 3.4. It is comfortable to conduct this on account of Sigma Plot Automation authorizes you to look for a notebook (in this case the standard.jfl notebook containing all the curve fit equations) for benefit objects (or objects of any type) as well as create a list of them. If you expect you can displaced in this macro a contrary notebook with another group of fit equations. The new equations will that time display in the dropdown list. If a user-defined equation is acquired to common. jf l that time it will display in the list. The batch process effects are that time saved in a notebook. You may browse to select the suitable file.



Fig 3.4 Specify the notebook to save the results.

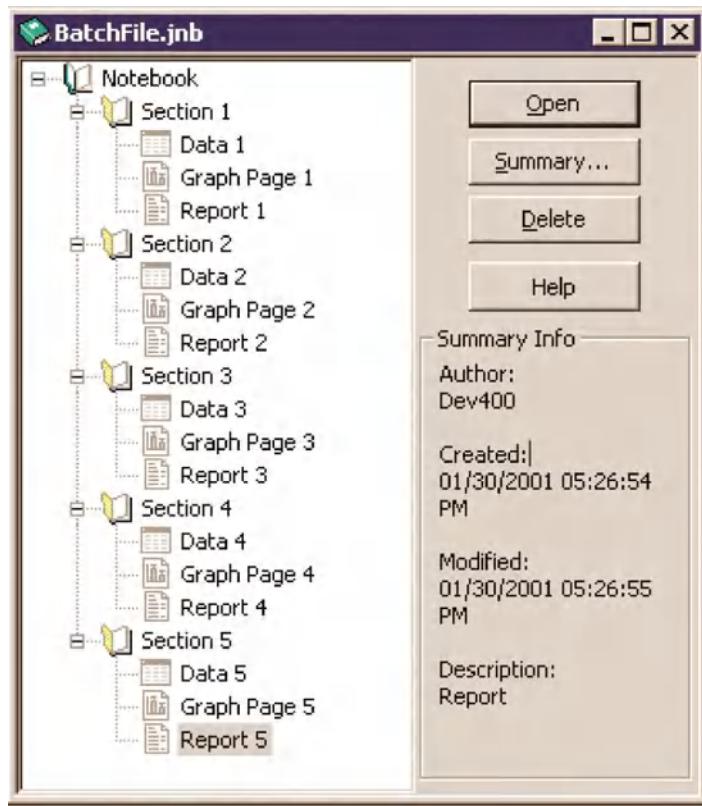


**Fig 3.5 Batch File**

For the five files shown in Figure 3.5, the notebook contains five sections each with worksheets with individual data sets, scatter plots of the data, fit results and detailed curve fit reports.

### 3.3.3 Command interpreter

A command interpreter, or command processor, occurs that critical component of the operating system software that analyses, or programs, the commands you allocate, additionally that time delivers them out for you. In DOS, the command processor is normally COMMAND. COM, furthermore DOS allows you alternate another command processor if you expect. This cries out slightly technical as well as scary, however it really isn't complicated.

#### **Check your progress 2**

1. In an operating system, a batch procedure works as:

- |                 |          |
|-----------------|----------|
| a. instructions | c. rules |
| b. commands     | d. all   |

2. In a program, the data files acts as:
- |           |                  |
|-----------|------------------|
| a. input  | c. processes     |
| b. output | d. none of these |
3. Batch processing is concern with\_\_\_\_\_computers.
- |           |              |
|-----------|--------------|
| a. analog | c. mainframe |
| b. hybrid | d. all       |

---

### 3.4 Let Us Sum Up

---

**In this unit, we have learned:**

- Batch operating system is a kind of operating system requirement which mainly involves in mainframe computer
- Batch processing may additionally be applied for altering computer files from one format to another
- Batch processing has been affiliated with mainframe computers owing to the earliest decades of electronic computing

---

### 3.5 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1- b)

**Check your progress 2**

**Answers:** (1 - d), (2 - a), (3 - c)

---

### 3.6 Glossary

---

1. **Batch** - It is term that describes the amount of the work to do particular jobs continuously.
2. **Batch operating system** - A kind of operating system need mainly involves in mainframe computer.

---

### **3.7 Assignment**

---

Write short details on Batch data.

---

### **3.8 Activities**

---

Study the different techniques of Batch jobs.

---

### **3.9 Case Study**

---

How will the Computer be able to perform Batch Jobs?

---

### **3.10 Further Readings**

---

1. The Operating system by Andrew Tannenbaum.
2. Operating System by Mach.



---

## Block Summary

---

In this block, we have studied about the basic of Operating System and different types of Operating System. We have an idea about the necessary function and advantages of using an operating system. There are abundant Operating Systems those monopolize be constructed for functioning the performances those are demanded by the user. Batch is the term which is given to the work of doing similar jobs continuously again and again but with a difference as in this the input data is shown for every iteration of the job and probably the output file.

The block detailed about the concept that explains the usability and structuring about Operating System. In this block students have given an idea about Batch processing system. The block focuses on practical implications of operating system.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. What is an Operating System?
2. Explain Real time Systems?
3. What is batch processing?
4. What is Distributed system?
5. Explain Multi-user System?

### **Long Answer Questions**

1. Discuss the different types of Operating System?
2. What is the need of Multi-programming Operating System?
3. What are the advantages of Batch processing System?

Enrolment No.

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# ARCHITECTURE OF COMPUTERS

PGDCA 101

## BLOCK 1: LOGICAL ARCHITECTURE OF COMPUTER



**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**

# ARCHITECTURE OF COMPUTER

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Ms. Rashika Paul

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.



Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# ARCHITECTURE OF COMPUTER

## Contents

---

### **BLOCK 1: LOGICAL ARCHITECTURE OF COMPUTER**

#### **UNIT 1 INTRODUCTION TO MEMORY AND PROCESSOR**

Preliminary Overview of Processor, Memory, and Hard Disk.  
Loading Program into Memory, Loading & Storing Data to Hard Disk

#### **UNIT 2 LOGICAL ARCHITECTURE**

Registers, ALU, Internal Bus, Read/Write Cycle of Memory, Internal Bus, Control Unit, Cache Memory, Etc

#### **UNIT 3 PROGRAM EXECUTION**

Execution of Program. Fetch-Decode-Execute Cycle. With Reference To Fetch-Decode-Execute Cycle. Buffering Concept

---

### **BLOCK 2: PHYSICAL ARCHITECTURE OF COMPUTER/LAPTOP AND SMARTPHONE**

#### **UNIT 1 MEMORY CHIP AND CACHE MEMORY**

Processor, Memory chip, and Cache memory – general overview

#### **UNIT 2 LOGICAL BUS**

Logical bus-oriented architecture of PC, Introduction to Buses – FSB, PCI Bus, and USB

#### **UNIT 3 MOTHERBOARD**

Motherboard - various chips on the mother board - processor, RAM, L2 Cache, BIOS & Chipset, Memory Controller Hub (North Bridge) & I/O Controller Hub (South Bridge)

#### **UNIT 4 HARDWARE COMPONENTS OF SMARTPHONE**

Applications Processor, SIM card, Wireless connectivity, Audio subsystem, memory card and flash memory, Power Management Unit, Camera & its interface, Touch screen & its interface



**BLOCK 3: DEVICES FOR DISPLAY, INPUT AND POINTING**

**UNIT 1 DISPLAY DEVICES**

LCD and LED monitors (for PC)

**UNIT 2 INPUT AND POINTING DEVICES**

Keyboard and mouse (for PC and Laptop), touch pad (for laptop), touch screen (for Mobile) – basic principle of each device and its interface with computer

**BLOCK 4: DEVICES FOR NETWORKING, I/O AND EXTERNAL STORAGE**

**UNIT 1 NETWORKING DEVICES**

Modem, Wireless Modem etc.

**UNIT 2 I/O AND EXTERNAL STORAGE**

Pen Drive, Camera, Hard Disk, Optical Storage (CD/DVD),  
Drives for CD/DVD, Inkjet Printers and laser printers



Dr. Babasaheb  
Ambedkar  
Open University

**PGDCA 101**

# **ARCHITECTURE OF COMPUTER**

---

## **BLOCK 1: LOGICAL ARCHITECTURE OF COMPUTER**

---

### **UNIT 1**

|                                      |    |
|--------------------------------------|----|
| INTRODUCTION TO MEMORY AND PROCESSOR | 03 |
|--------------------------------------|----|

### **UNIT 2**

|                      |    |
|----------------------|----|
| LOGICAL ARCHITECTURE | 20 |
|----------------------|----|

### **UNIT 3**

|                   |    |
|-------------------|----|
| PROGRAM EXECUTION | 33 |
|-------------------|----|

---

# **BLOCK 1: LOGICAL ARCHITECTURE OF COMPUTER**

---

## **Block Introduction**

Computer architecture is referred to as a deep study of data and information along with its processing and execution. The main work of a computer is to give instructions and to return the desired output. For this purpose, it uses different hardware and software that plays an important role. It is interesting to see how a computer uses our information and by utilising its technology, gives us the output. We see that once the hardware is ready, now it's time to use it by the way it understands. So this block stresses on the execution, decoding and carrying out the information a computer system understands.

In this block, you will study and learn more about internal hardware structure of a Computer. In this the knowledge regarding data that you entered can be stored for longer duration. The concept related to Memory and Processor is well explained. The knowledge related to data storage in secondary and primary memory will help students to learn more.

After studying this block, user will be able to understand about processor and its functional features. The students will be demonstrated with logical architecture of a computer system that will create interest about the computers on which they used to work. The necessary hardware details will make them aware regarding data calculation, transmission and manipulation of data with more basic facts about Registers, ALU, Internal Bus, Read/Write Cycle of Memory, Internal Bus, Control Unit, Cache Memory, etc. The block gives diagrammatic explanation which will help the students to work of their own.

## **Block Objective**

**After learning this block, you will be able to understand:**

- The various Computer Hardware components required for its processing.
- How information is received and saved.
- Loading and unloading of programs in storage devices.
- How a computer executes the information.

## **Block Structure**

**Unit 1:        Introduction to Memory and Processor**

**Unit 2:        Logical Architecture**

**Unit 3:        Program Execution**

---

# **UNIT 1: INTRODUCTION TO MEMORY AND PROCESSOR**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Preliminary Overview of Processor**
- 1.3 Memory and Hard Disk**
  - 1.3.1 Memory
  - 1.3.2 Hard Disk
- 1.4 Loading Program into Memory**
- 1.5 Loading and Storing Data to Hard Disk**
- 1.6 Let Us Sum Up**
- 1.7 Answers for Check Your Progress**
- 1.8 Glossary**
- 1.9 Assignment**
- 1.10 Activities**
- 1.11 Case Study**
- 1.12 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- The Computer Memory and Processor
- The Hard Disk
- About loading of programs into Memory
- About storing of data into Hard Disk



---

## 1.1 Introduction

---

In this unit, we will learn about the basics of a Computer Processor and its working. It is studied that computers do not understand high level languages, so the need of a processor exists as only a processor will be able to judge the instructions encoded in binary form. In order to execute a program, a computer processor receives the instruction from the computer memory and tries to execute it. A processor performs certain instructions as its basic operation which is kept in the main memory of a Computer system. A processor impacts the total computing power and performs most of the computer's operation. In multiprogramming system, a virtual memory is required to do this work. We say that memory is just a part of a Computer system that remembers things for the CPU.

---

## 1.2 Preliminary Overview of Processor

---

A processor, also known as Central Processing Unit (CPU), is the brain of a Computer which understands and performs many basic instructions that helps in operating a Computer system. The first processor was invented by Intel in 1971 which is a 4 Bit device having a speed of 108 KHz. A Processor contains thousands or millions of small switches known as transistors. In mainframes and supercomputers, the different functions done by the processor expands from an individual chip to multiple circuit boards, where as in a personal computer, the functions of the processor are normally on a single chip. All computers, large and small, must have a processor or central processing unit

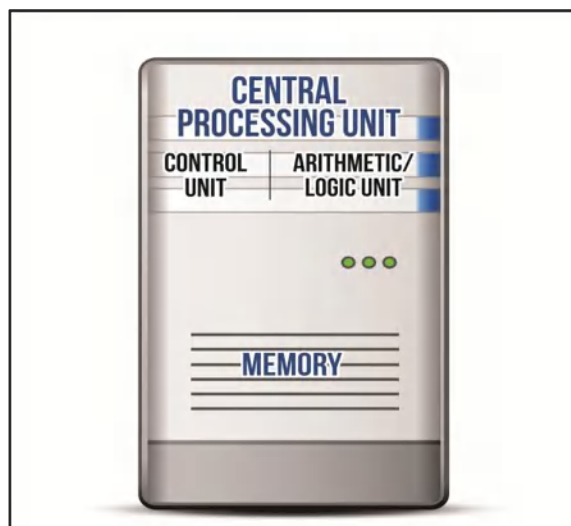


**Fig. 1.1 Layout of Processor**

Figure 1.1 shows the layout of a processor (CPU). Its main part includes:

- Arithmetic logic unit (ALU)
- Registers
- Control unit

We see that the ALU is termed as Arithmetic logic unit which performs arithmetic and logic operations. The register stores the bits of information supplied to the object on which the operations are to be performed and passes them to the ALU and further keeps the result of the ALU operations. The control unit gets the instructions from memory and perform the operations by directing the matched operations of the ALU registers and other components.

A processor operates with a speed of an internal clock when a current is applied. The main function of a processor is to carry out a series of stored instructions which are known as programs. For every instruction, the processor uses the following set of basic operations which is known as machine cycle.

1. Fetches the instruction or data from memory.
2. Decodes the instructions into commands as understood by the computer.
3. Executes the commands.
4. Stores the result in the memory.

### Advantages

1. It performs fast calculation of mathematical data.
2. It controls the configuration of tiny switches known as transistors.
3. It controls the overall working of peripherals.

### Check your progress 1

1. The first processor was invented by\_\_\_\_\_.
  - a. Intel
  - b. Microsoft
  - c. Apple
  - d. Unix
2. Which is not a part of processor?
  - a. Arithmetic logic unit (ALU)
  - b. Memory
  - c. Registers
  - d. Control unit

---

## 1.3 Memory and Hard Disk

---

In any computer system, both memory and hard disk play an important role. Memory is a temporary storage place for data, instructions and information. It stores the operating system, application programs and processed data. Whereas, a hard disk is a spindle of magnetic discs which holds numerous gigabytes of data. It stores the information permanently. It is seen that any stored data is lost in the memory if the power is switched off whereas the data remains permanent in the Hard disk even if you switch off the computer system. Let us discuss both in details.

### 1.3.1 Memory

Memory is a part of a computer system that temporarily stores information that is required by the processor or CPU. Since CPU cannot do any job without a memory so memory contains small flat boxes of chips which are same as CPU chips containing memory devices instead of CPU circuits. While working, a CPU needs both: an information to be manipulated and instructions that tells the CPU to perform actions on that particular information. Finally the memory holds these things and provides it to the CPU whenever the CPU asks for it. It is seen that everything stored in the memory will wipe off when you have switched off the Computer System.

Computer memory is similar to our brain as it stores the data and instruction. It is located inside the computer and is placed on the motherboard. It is located in a particular area in the computer system where the data is calculated and the instructions are stored. The memory is divided into several small memory parts like cells. Each cell has a different address which varies from 0 to -1.

Memory is mainly of three types:

- Cache Memory
- Primary Memory
- Secondary Memory

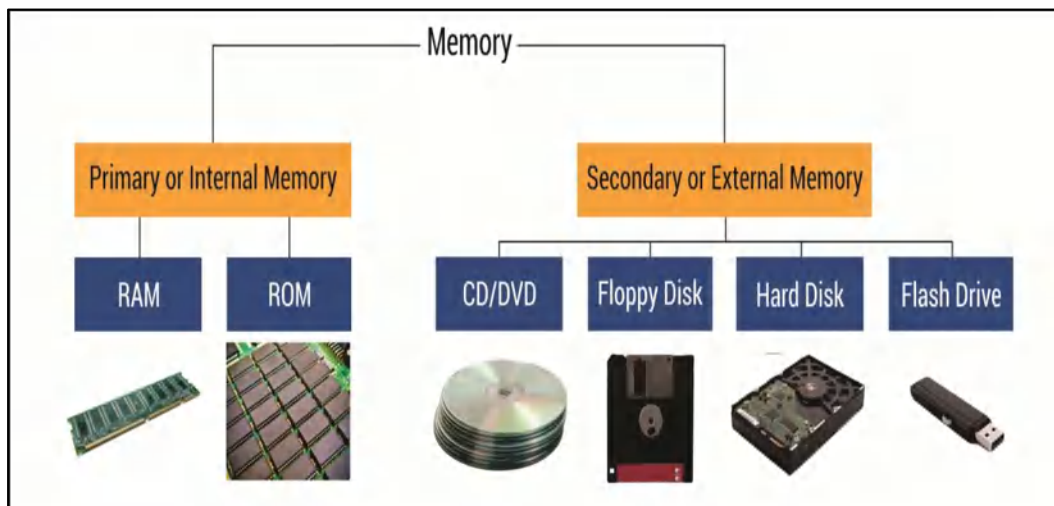


**Fig. 1.2 Types of Computer Memory**

### **1. Cache Memory**

It is a type of high speed computer memory used for CPU to perform the work faster. It serves as a buffer between the CPU/processor and main memory. Such memory holds the data and the program that are required by the CPU/Processor. It has certain advantages and disadvantages.

#### **Advantages**

- Much faster
- Requires less working time
- Stores particular programs to be performed for a short period
- Stores data for short-term purpose

#### **Disadvantages**

- It has low capacity.
- It is expensive.

### **2. Primary Memory**

Primary memory is the main memory of the computer. Such memory keeps that part of the data and instructions on which the computer works currently. It has less capacity. It is a volatile memory in which the data gets erased when the power is switched off.

It is of two types:

- RAM – It is known as Random Access Memory. It is the main working memory that is used by the computer.



**Fig. 1.3 RAM**

- ROM – It is known as Read Only Memory. It is a special type of memory that keeps the software that can only be read but cannot be written.

#### **Characteristic**

- Such type of memory is a semiconductor memory.
- It is the main memory of computer.
- It is not stable.
- In this, the data gets erased when power is switch off.
- It is the working memory of a computer.
- It is faster than any other memory.
- Without such memory, the computer system cannot work.

### **3. Secondary Memory**

Secondary memory is a type of external memory which is also known as non-volatile memory which is particularly used to store data and information. This type of memory is slower than primary memory or the main memory. Because of its non-volatile characteristics, the data or the information remains inside it permanently even after the power is switched off.

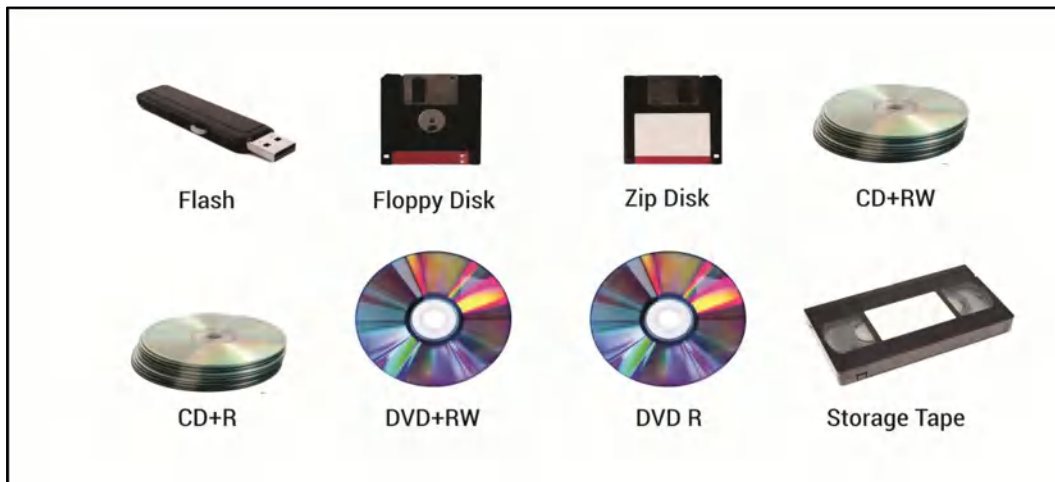


**Fig. 1.4 Types of Secondary Memory**

Figure 1.3, shows the different types of Secondary memories, which are:

- Hard disk
- CD-ROM
- DVD
- Flash drives
- Zip drives
- Storage tapes

#### **Characteristic**

- It is a type of magnetic and optical memory.
- It is commonly known as backup memory.
- In this, the data remains permanently inside it even after the power is switched off.
- Without this memory, the computer can run easily.
- It is slower than the primary memory.

#### **Measuring Computer Memory**

You can measure the Computer memories with the help of:

- **Bit:** It is the smallest unit of measurement of computer memory that stores the binary numbers 1 or 0.
- **Byte:** A byte is a collection of 8 bits.

- **Kilobyte:** A kilobyte (KB) consists of 1024 bytes.
- **Megabyte:** A megabyte (MB) consists of 1024 kilobytes.
- **Gigabyte:** A gigabyte (GB) consists of 1024 megabytes.

### 1.3.2 Hard Disk

A hard disk drive is the device which is used to store large amounts of information permanently of up to 500 GB. Hard disk drives are used to store operating systems, software and data. It is attached inside the computer system.



Fig. 1.5 Hard disk

Hard disk is a permanent storage place which is best for any application which requires very fast access to data both in the case of reading and writing. On the other hand, it is seen that, hard disk drives are not proper for applications that require portability. Nowadays, almost every computer system is installed with a fixed hard disc that can be used for on-line and real time processes with direct access to the computer system. Hard disks are used in file servers for computer networks to store large amount of data and information. It is seen that the capacity of 1 Hard Disk is equal to 7, 50,000 Floppy Disk's space.

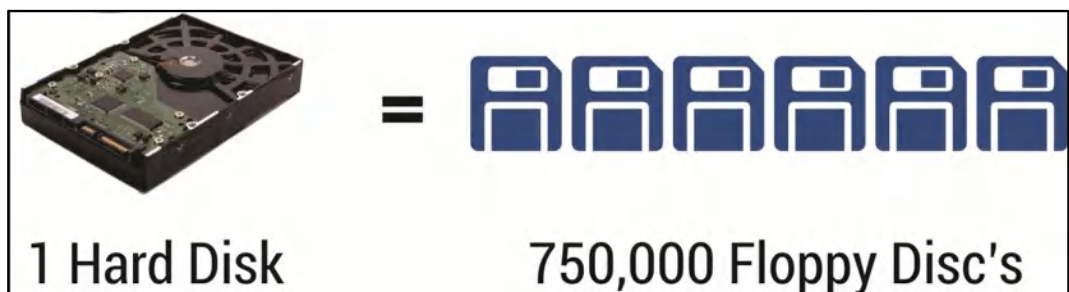


Fig. 1.6 Hard Disk

Everything on the hard disk is organized into electronic paper known as files. These files store the information you write or design by its name. It is easy to locate files stored in the Hard disk, if you have saved the file with a file name. Just like memory, a hard disk also requires instructions to save your data. While saving your file, you need to tell the computer about the location where you want

your file to be placed on the hard disk. Hard Disk's save anything right from files, folders to images and pictures. Apart from saving anything on your hard drive, you can also delete any unwanted items from the Hard disk. Data on Hard disks can be stored permanently for longer duration of time.

### Check your progress 2

1. To store data permanently, we use:-
  - a. Hard disk
  - b. RAM
  - c. Floppy
  - d. Pen drive
2. Which is not an example of Secondary Memory?
  - a. RAM
  - b. Hard Disk
  - c. Floppy
  - d. CD

---

## 1.4 Loading Program into Memory

---

It is the duty of the memory management to load the programs into the primary memory. This loading process is basically done to carry out the action by the processor or CPU. The principle idea of a virtual memory involves the use of Segmentation and Paging techniques. Partitions can be fixed or dynamic.

### 1.4.1 Fixed Partitioning

It is seen that an operating system will itself live in a fixed part of the main memory and leaves the balance space for different user/application processes. To analyse a main memory in fixed-sized partitions there exists two possibilities:

- Equal size partitions
- Unequal size partitions



### Equal-size partitions

In case of a new process having the size less than or equal to the partitions' size it will directly be loaded in such a partition. Figure 1.4 shows Equal Partitioning in a Memory.

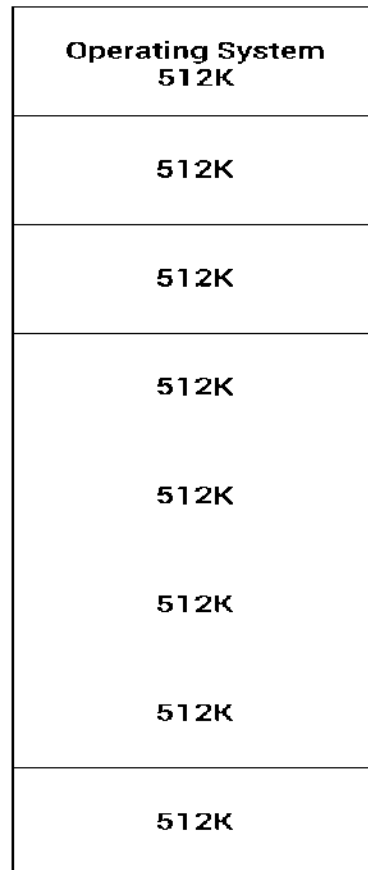


**Fig. 1.7 Equal Partitioning in a Memory**

There exist two problems with Equal sized partitions:

1. The process requirements exceed the size of the partition.
2. Small process occupies a full partition.

### Unequal size partitions

In case of unequal size partitions, there exist two directions in order to allocate processes to partitions. The easy way to do this is to put each process to the smallest partition as per its size. Figure 1.7 shows unequal partitioning in a Memory.

|                          |
|--------------------------|
| Operating System<br>512K |
| 128K                     |
| 256K                     |
| 320K                     |
| 576K                     |
| 758K                     |
| 1M                       |

**Fig. 1.8 Unequal Partitioning in a Memory**

For this, a scheduling queue is required for every partition in order to hold change out processes that is kept for that partition. With this, the processes are allocated in such a way that it will lower the exhaust memory inside the partition. Such process seems to be good for an individual and not for all.

### 1.4.2 Dynamic Partitioning

Dynamic partitioning takes care about some defects of fixed partitioning in which the partitions are of uneven length and number. In this, when a process begins, it will fill a partition of exactly the required size with no extra space. Figure 1.6 shows the arrangement of dynamic partitioning in a memory. There are various parts of this partitioning diagram.

In figure (a), it is seen that the primary memory is empty except for an operating system. The next three diagrams (b), (c) and (d) show that an operating system ends and occupies much space for each process. With this, a hole present at the end of a memory is too small for the next process. It is seen that during certain point, none of the processes in the memory is ready, so the operating

system changes out the second process as shown in diagram (e), which leaves enough space to load a new process as shown by diagram (f). From the diagrams it is seen that the fourth process is smaller than the second process that will result in the creation of another small hole. Now at a point where none of the processes in the main memory is ready but the second process is ready, then a hang up state will become available.

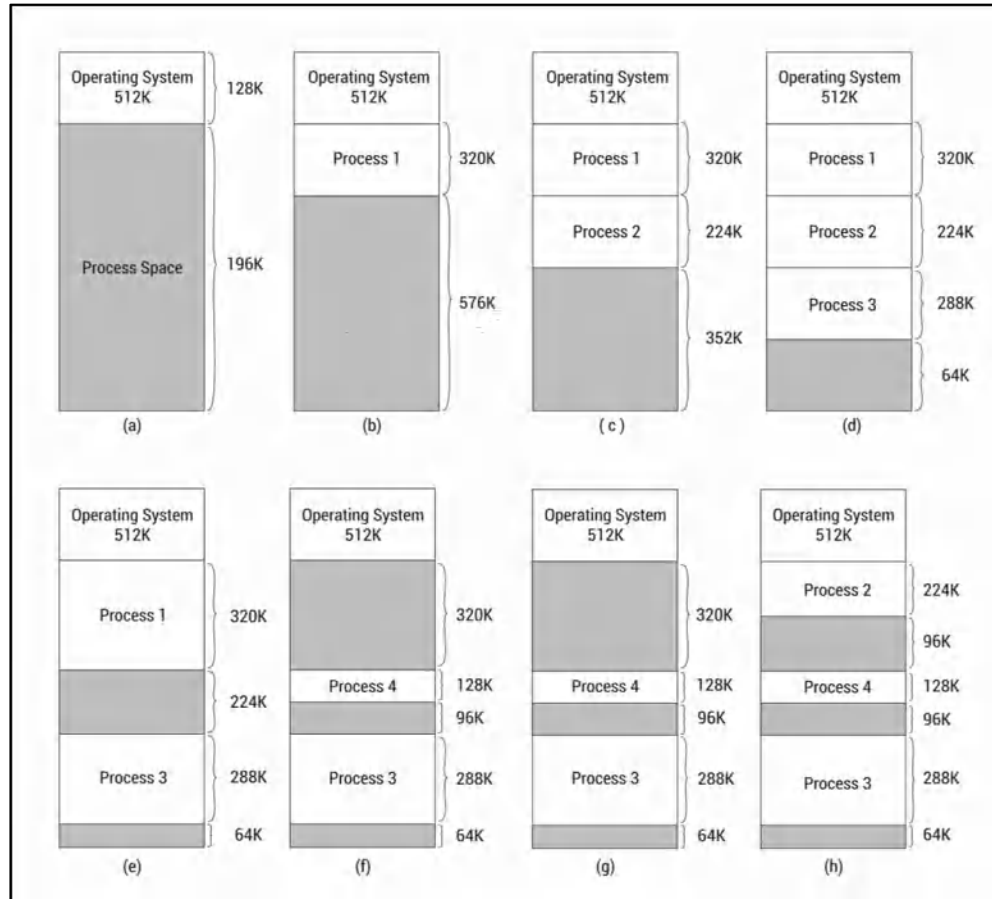


**Figure 1.9 Arrangement of dynamic partitioning**

Because of less space in the memory for the second process to be executed, the operating system hangs the first process out as shown in diagram (g) and changes the second process back as highlighted in diagram (h). It is seen that such methods will work out well, but finally it shows a situation where there are many small holes present in the memory. With the passage of time, the memory gets fragmented and declines.

### Check your progress 3

1. Virtual memory involves the use of \_\_\_\_\_ techniques:
  - a. Segmentation
  - b. Paging
  - c. both
  - d. none
2. In equal partition, the size of the process in all the partition remains:
  - a. same
  - b. different
  - c. varies with number
  - d. less than 1KB
3. Dynamic partition involves the use of \_\_\_\_\_ processes.
  - a. one
  - b. two
  - c. three
  - d. four

## 1.5 Loading and Storing Data to Hard Disk

Hard Disk is a storage media where the data or information is loaded and stored. This information could be software, data files, images etc. A user can access his stored or loaded information anytime. The hardware arrangement of a hard disk is shown in figure 1.7 where there are hardware parts such as Actuator, Read/Write Actuator arm, Spindle, Read/Write head and Platter. It is seen that the data sent or taken from the hard drive is normally done or is understood by the disk controller. The disc controller tells the hard drive what to do and how to move the components inside the drive. The figure 1.7 highlights certain features of the data that is transferred or looked by hardware components.

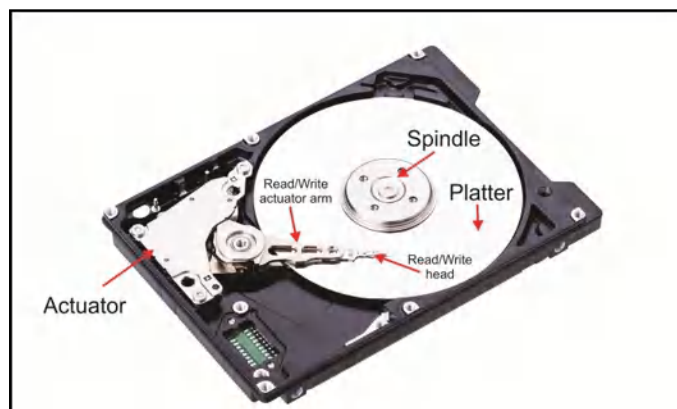


**Fig. 1.10 Data reading arrangement in Hard Drive**

If your hard disk is raw, you will first load an operating system. If any operating system needs to read or write its information, it first checks the hard drive for a File Allocation Table in order to decide about the file location and file place. After doing that, the disk controller directs the actuator to move read/write arm and arrange read/write head.

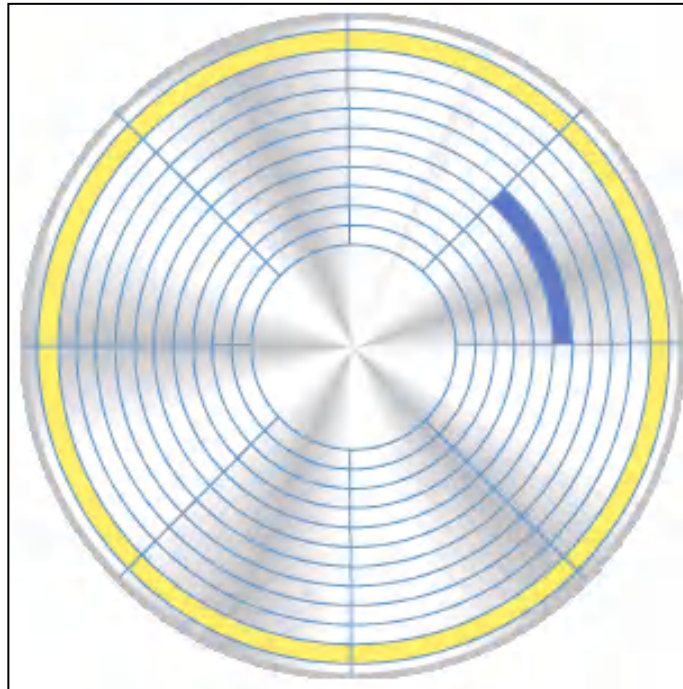


**Fig. 1.11 Tracks and sectors on Hard Drive**

It is observed that in the hard disk, the data is stored on the surface of a platter which is made up of aluminium, glass or ceramics that carries sectors and tracks as shown in figure 1.8. Each platter has two read/write head. We can see from the figure that tracks which are yellow in colour are concentric circles and sectors which are blue in colour are pie shaped wedges on a track. Further, a sector contains a standard number of bytes i.e. 256 or 512. During the loading of data the files spreads throughout the platter. Since files are scattered on the platter, the head tends to move to different areas in order to get all the relevant information.

#### **Check your progress 4**

1. Hard Disk is used to store:

- |             |                 |
|-------------|-----------------|
| a. pictures | c. information  |
| b. data     | d. all of these |

- |                                                    |             |
|----------------------------------------------------|-------------|
| 2. Which is not a hardware part of the Hard Drive? |             |
| a. spindle                                         | c. ribbon   |
| b. actuator                                        | d. platter  |
| 3. Hard Disk platter is made of:                   |             |
| a. iron                                            | c. zinc     |
| b. copper                                          | d. aluminum |

---

## 1.6 Let Us Sum Up

---

### In this unit we have learned:

- Memory is made up of one or more chips which hold the data or information temporarily during the processing.
- Certain memories are volatile, such as RAM, in which the data gets lost when the power is switched off.
- In a non-volatile memory, such as ROM, the data remains permanently even when the power is switched off.
- Hard disk is a storage media which stores data, instructions and information for future use as it is nonvolatile.
- Data or information on the hard disk can be loaded for longer use and any unwanted information can be deleted anytime.

---

## 1.7 Answers for Check Your Progress

---

|                              |
|------------------------------|
| <b>Check your progress 1</b> |
|------------------------------|

**Answers:** (1-a), (2-b)

|                              |
|------------------------------|
| <b>Check your progress 2</b> |
|------------------------------|

**Answers:** (1-a), (2-b)

**Check your progress 3**

**Answers:** (1-c), (2-a), (3-a)

**Check your progress 4**

**Answers:** (1-d), (2-c), (3-d)

---

## 1.8 Glossary

---

1. **Bit** - It is the smallest unit of a computer memory.
2. **Byte** - A unit of memory storage that is equal to 8 bits.
3. **CPU** - It stands for Central Processing Unit and is also known as a processor.
4. **Hard Disk** - It is a type of storage media which is used to store data, programs and information permanently.
5. **MB** - megabyte, it is a unit of memory storage that is equal to 1,048,576 bytes.
6. **Primary memory** - It is the main memory that helps the processor to work.
7. **Processor** - It is a computer component that interprets all the instructions.
8. **RAM** - Random Access Memory it is the primary memory and is volatile.
9. **ROM** - Read Only Memory is non-volatile memory.
10. **Volatile memory** - Memory that requires power to maintain the stored information. If the power is turned off, the stored memory is lost.

---

## 1.9 Assignment

---

Prepare a report showing the internal structure of a Hard disk and label its parts.

---

## 1.10 Activities

---

Write the major steps of comparison between a memory and a hard disk drive.

---

## 1.11 Case Study

---

Highlight the important features among volatile and nonvolatile memory and compare it with practical presentations.

---

## 1.12 Further Readings

---

1. The Indispensable PC Hardware Book. Addison-Wesley.
2. Introduction to Direct Access Storage Devices, M. Bohl, IBM.
3. Dandamudi, Sivarama P., Fundamentals of Computer Organization and Design, Springer.
4. Goda, K., Kitsuregawa, M. The History of Storage Systems.



---

## UNIT 2: LOGICAL ARCHITECTURE

---

### Unit Structure

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 Registers**
- 2.3 ALU**
- 2.4 Internal Bus**
- 2.5 Read/Write Cycle of Memory**
- 2.6 Control Unit**
- 2.7 Cache Memory**
- 2.8 Let Us Sum Up**
- 2.9 Answers for Check Your Progress**
- 2.10 Glossary**
- 2.11 Assignment**
- 2.12 Activities**
- 2.13 Case Study**
- 2.14 Further Readings**

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About registers.
- About ALU and its features.
- The Internal Bus Structure architecture.
- The Read/Write Cycle of Memory.
- About Control Unit and Cache Memory.

## 2.1 Introduction

The Computer system works on a set of logical processes which is known as logical architecture. The architecture is a group of regulations that shows functionality, organization and implementation of computer systems. The Logical Architecture explains the different activities and functions that are required to give certain user services. These processes work in a group and share information to give user service. Such processes are carried out with the help of software, hardware or firmware.

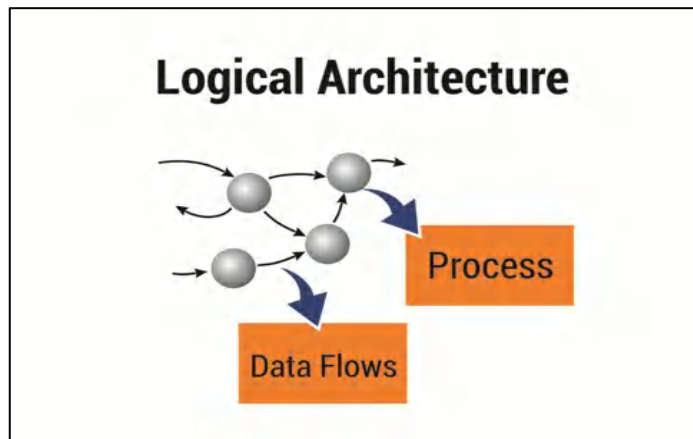


**Fig. 2.1 Logical process of a Computer**

Figure 2.1 gives us an idea about the logical process of a Computer system where different processes such as data flow; terminator and data storage takes place. Here data flows find the information that was shared by different processes. The basic entry and exit points for Logical Architecture comprises of sensors, computers and terminators. The basic logical structure of a computer is shown in figure 2.2.

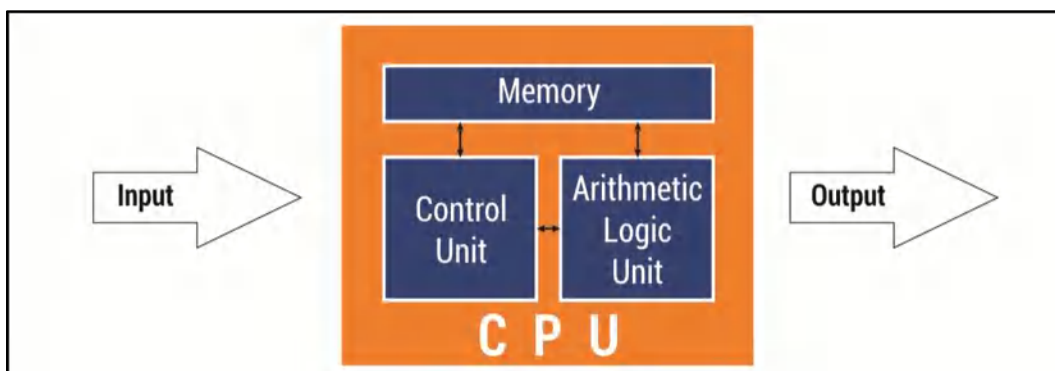


**Fig. 2.2 Logical Structure of Computer**

It includes:

- **BIOS:** It is basic input output system which is responsible for the booting of a computer.
- **CPU:** It is a processor which is the brain of the computer that executes processes.
- **Memory / RAM:** It is used to store temporary data.
- **Hard Disk:** This is used for storing permanent information or data.
- **Input / output devices:** It gives input to a computer and derives the output from it.
- **Communication Channel:** It serves as an interface for external and internal devices.

---

## 2.2 Registers

---

A register is an important part of a processor or a CPU having a separate memory location which helps in storing the temporary data and instructions. It is made of Flip flops. There are different types of registers:

1. **Memory Address Register (MAR):** Such register keeps the address of memory where the processor wants to read or write data.
2. **Memory Buffer Register (MBR):** This will hold the information of data or instruction that will be read from or written in the memory.
3. **I/O Address Register (I/O AR):** This register specifies the address of a particular I/O device.
4. **I/O Buffer Register (I/O I3R):** This Register exchanges the data among the I/O module and the processor.
5. **Program Counter (PC):** it is a register which is used to keep the address of the next instruction that is to be fetched for execution.
6. **Instruction Register (IR):** This register stores the fetched instruction from the main memory.
7. **Accumulator Register:** This register is placed inside the ALU and is used at the time of arithmetic & logical operations performed by the ALU.

8. **Stack Control Register:** This register manages the stacks in memory. Such type of register is of the size 2 or 4 bytes.
9. **Flag Register:** This register is used to show the occurrence of a certain condition at the time of operation of the processor. In this type of special purpose register with size being 1 byte or 2 bytes, each bit of the register forms a flag.

### Check your progress 1

1. The computer architecture is a group of regulations that shows\_\_\_\_\_of computer systems.
  - a. functionality
  - b. organization
  - c. implementation
  - d. all
2. The basic logical structure of computer carries:
  - a. memory
  - b. uses
  - c. instructions
  - d. data
3. Registers are made of:
  - a. RAM
  - b. ROM
  - c. Logic gates
  - d. Flip flops
4. \_\_\_\_\_register is placed in ALU and is used at the time of logical operations.
  - a. Instruction
  - b. Accumulator
  - c. Stack Control
  - d. Flag

---

## 2.3 ALU

---

ALU means Arithmetic and Logic Unit. It performs arithmetic and logical operations on data and information. It is an electronic component in the computer which is based on the use of simple digital logic devices that can store binary digits and perform simple operations.

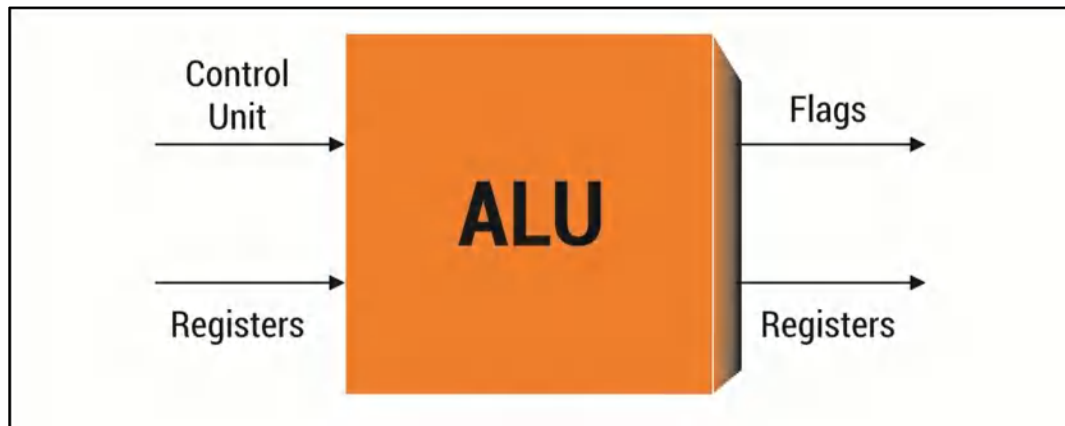


**Fig. 2.3 Arrangement of ALU**

Figure 2.3 shows the arrangement of ALU along with a processor. It is a major component of the CPU that enables arithmetic and logic operations to be done on instruction. It is divided into Arithmetic unit (AU) and Logic unit (LU). Data are presented to the ALU in registers whose results are stored in registers. The registers are temporary storage places inside the processor that is joined by the signal paths to the ALU. The ALU sets flags as the result of an operation. The arithmetic-logic section performs arithmetic operations, such as:

- Addition
- Subtraction
- Multiplication
- Division

Such operations of ALU can be described as:

- Logical Operations: It performs logical operations with Logic gates.
- Bit Shifting Operations: It performs movement of bits by certain places either right or left.
- Arithmetic Operations: It performs addition and subtraction of bits.

As shown in the figure 2.4, the arrows show the flow of data between the arithmetic logic unit and memory at the time of processing. Here the data is transferred from the memory to arithmetic logic section for processing.

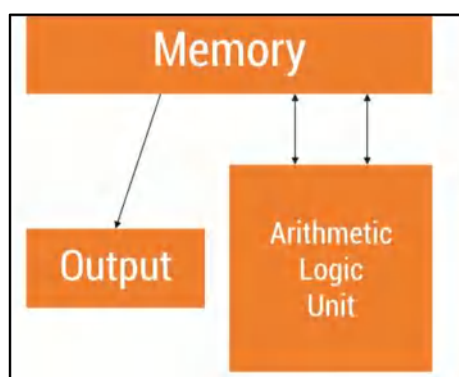


**Fig. 2.4 Flow of Data between Arithmetic logic unit and Memory**

It is seen that soon after processing, the data is returned back to the memory again. This data is transferred back and forth between these two sections continuously. The result so obtained is transferred from memory to the output device. If you see the binary addition and multiplication, you will find the grouping as:

$$0+0=0$$

$$0+1=1$$

$$1+1=10$$

$$1+0=1$$

$$0\times 0=0$$

$$0\times 1=0$$

$$1\times 1=1$$

$$1\times 0=0$$

From the above mathematical operation table, it is seen that if you ignore the carry, then subtraction of two single digit binary numbers will result in the same bit addition.

### Check your progress 2

1. ALU stands for

a. Arithmetic logic Unit

c. Arithmetic Local Unit

b. Addition Logic unit

d. Addition Local unit

2. Which is not performed by ALU
  - a. Addition
  - b. Subtraction
  - c. Multiplication
  - d. Exponent
3. Which is correct logical operation
  - a.  $0+0=1$
  - b.  $0+1=0$
  - c.  $1+1=2$
  - d.  $1+0=1$

## 2.4 Internal Bus

In a computer system, a bus is a connection made between two or more than two devices. It is an internal circuitry of a processor that communicates with the internal caches of memory as a part of the CPU chip design. It is faster in operation and is independent of other computer operations. Figure 2.5 shows an arrangement of a bus structure inside a computer.

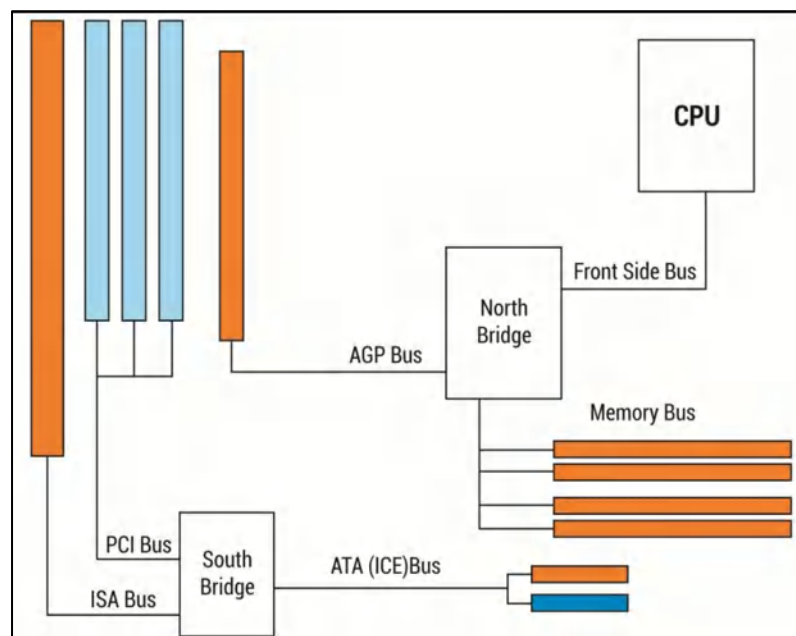


**Fig. 2.5 Bus Structure of a Computer**

Since a bus connects many devices, so it contains multiple wires which are basically signal lines having addressing information that explains about the memory location either to send or retrieved data. Every single wire carries a bit of

information. Today, normally all buses whether parallel or serial are utilized by computer systems. There are two types of Buses:

- Internal bus
- External bus

An internal bus makes a communication among internal components of a computer system such as video card and memory, whereas, an external bus communicates with external components of a computer system such as USB or SCSI device.

The internal bus is commonly known as system bus which comprises of four parts such as:

- **Power bus:** It has wires which gives power to every part of the main board.
- **Control bus:** It sends the timing signals out to make the other components to stay on the main board at a time with the processor.
- **Address bus:** It sends the information on memory addressing that will tell the parts installed on the main board about the instructions and data in memory.
- **Data bus:** It transmits the real data among the system components.

### Check your progress 3

1. In a computer system, \_\_\_\_\_ is a connection made between two or more than two devices.
 

|         |          |
|---------|----------|
| a. bus  | c. rod   |
| b. line | d. plate |
2. External bus can communicate with:
 

|               |               |
|---------------|---------------|
| a. memory     | c. audio card |
| b. video card | d. USB device |
3. \_\_\_\_\_ bus transmits the real data among the system components.
 

|            |            |
|------------|------------|
| a. Power   | c. Address |
| b. Control | d. Data    |



---

## 2.5 Read/Write Cycle of Memory

---

Primary memory is a read/write memory in which the data reads or writes. Such process of reading and writing of data is called as memory cycle. Such type of cycle allows reading of data out of memory and writing of data into the memory. This is done with the help of the read/write operation or separate read and write operations. It is examined that a memory cycle depends on the fixed time periods for reading and writing of data from the memory or into the memory. After the execution of read and/or write operations, there exists an address translation which makes the read and/or write cycle or practically such an operation starts with a memory cycle.

It is seen that before the beginning of a read or write cycle, there exists an important process known as memory address translation. In the memory, series of memory locations begins with address 0 and extends up to the highest memory address that is made available with the requester i.e. CPU or I/O. This process is same as the process that involves sending and receiving of e-mails. Before sending a mail, a user should have a mail address where the mail is to be delivered or sent. Just like that, memory also obeys a similar principle. Memory logic recognizes a memory address where the memory word is read from or written into. Therefore, a memory address could be anything from the memory addresses (0 to maximum). To locate the required memory address, memory logic uses a register which can form address registers or translators or decoders. Practically, memory logic gets the logical address either from the processor or from the I/O which temporarily stores it in the address register. The stored information then gets converted into a physical address which can be read from or written into.

In this, the address register or a translator finds the correct place from where it can read or write the bits. The data or information contained in the address register or translator identifies the correct memory address. Further, the memory logic is designed to have its selection done on the basis of type of memory it utilises. Also, memory cycle is designed to find a memory address of a single memory PCB or to find an address placed in one of four memory locations.

### Check your progress 4

1. To locate the required memory address, memory logic uses:
 

|             |           |
|-------------|-----------|
| a. buses    | c. cpu    |
| b. register | d. memory |

## 2.6 Control Unit

The control unit is an important part of central processing unit as it directs the flow of operations and data. It maintains the order of operations inside the computer.

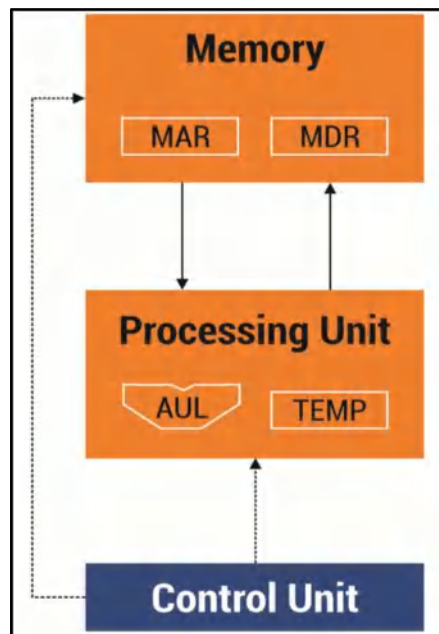


Fig 2.6 Arrangement of Control Unit

In figure 2.6, memory, processing unit and control unit displays a block of operations. The flow of control is shown by dotted lines. The control section selects the program statement from the program storage area and after understanding the desired statement sends it to the arithmetic logic unit and memory unit to carry out the required instructions.

It is examined that a control section does not perform any actual processing operations on the required information or data. The Control unit simply instructs the input device regarding the start and stop of transferring of data to the input storage area. Further the Control unit instructs the output device when to start and stop receiving of data from output storage area.

### Check your progress 5

1. Control Unit is the part of:

- |                    |        |
|--------------------|--------|
| a. memory          | c. CPU |
| b. Processing unit | d. all |

## 2.7 Cache Memory

Cache memory is the CPU memory which is commonly known as Random Access Memory (RAM) is such type of memory that can be easily accessed by a microprocessor. Such type of memory is directly placed around the microprocessor chip as shown in the figure 2.7.

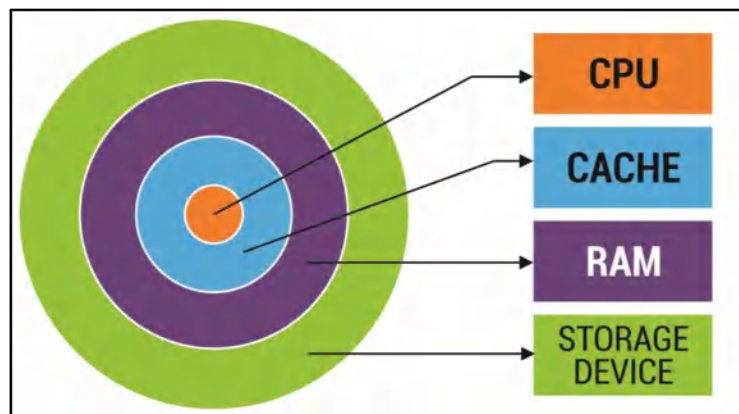


**Fig. 2.7 Position of Cache Memory**

Figure 2.7 shows the arrangement of cache memory. The main function of the cache memory is to store program instructions that are frequently arranged by software at the time of operation. By accessing such instructions, the speed of the software programs gets increased. After the data is processed by the microprocessor, it first sends it to the cache memory.

### Check your progress 6

1. Cache memory is known as:

- |         |           |
|---------|-----------|
| a. ROM  | c. RAM    |
| b. PROM | d. EEPROM |

---

## 2.8 Let Us Sum Up

---

### **In this unit, we have learned:**

ALU stands for Arithmetic and Logic Unit which performs Addition, Subtraction, Multiplication and Division. It is found that data paths are internal registers which can be arithmetic as well as logical part that carries the desired bus structures.

Memory in Computer architecture is the main or primary memory which is a collection of cells or locations. It is found that control unit is the main part of the computer architecture which controls the progress of other parts.

Cache memory is the CPU memory which is commonly known as Random Access Memory is such type of memory that can be easily accessed by a microprocessor.

---

## 2.9 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-d), (2-b)

**Check your progress 2**

**Answers:** (1-a), (2-d), (3-d)

**Check your progress 3**

**Answers:** (1-a), (2-d), (3-d)

**Check your progress 4**

**Answers:** (1-b)

**Check your progress 5**

**Answers:** (1-c)

**Check your progress 6**

**Answers:** (1-c)

---

## 2.10 Glossary

---

1. **ALU** - It is Arithmetic and Logic Unit that performs ADD, SUBTRACT, AND, OR, and NOT operations.
2. **Data path** - These are internal registers, ALU and its connecting buses.
3. **Memory** - It is a collection of cells or locations.
4. **Control Unit** - It is a control tower in the computer architecture that checks the progress of other parts.

---

## 2.11 Assignment

---

Prepare a report and write the various features about the logical architecture of a Computer.

---

## 2.12 Activities

---

Design an internal architecture model of a processor and discuss it.

---

## 2.13 Case Study

---

Visit to a Computer Hardware Store and enquire about the location of cache memory.

---

## 2.14 Further Readings

---

1. Linda Null; Julia Lobur. Essentials of Computer Organization and Architecture
2. C. Gordon Bell; R. Cady; A New Architecture for Mini-Computers

---

## **UNIT 3: PROGRAM EXECUTION**

---

### **Unit Structure**

- 3.0 Learning Objectives**
- 3.1 Introduction**
- 3.2 Execution of Program**
- 3.3 Fetch-Execute Cycle**
- 3.4 Fetch-Decode-Execute Cycle**
- 3.5 Program Execution with Fetch-Decode-Execute Cycle**
- 3.6 Buffering Concept**
- 3.7 Let Us Sum Up**
- 3.8 Answers for Check Your Progress**
- 3.9 Glossary**
- 3.10 Assignment**
- 3.11 Activities**
- 3.12 Case Study**
- 3.13 Further Readings**

---

### **3.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About execution of programs.
- Fetch Decode Execution Cycle.
- About buffering.

---

## 3.1 Introduction

---

In Computer architecture, a program is a set of instructions that the Computer system will follow. A computer is designed to work on the instructions that are written by the programmer in a simple language known as machine language. This type of language is only understood by the computer system. Different computers have their own set of instructions written in a specific machine language which can help to execute a program directly. If the instructions are written in any other language, then such instructions will not be understood by the computer as it is only familiar with the Machine Language.

When a process or performs a certain program, then the output of the program is stored in the computer's main memory which is known as RAM or Random Access Memory. Apart from the program, the computer memory also keeps the data that is being used or developed by the program. There exists a sequence of locations inside the main memory. Such locations are marked with numbers and the sequence number of a location is termed as its address. To show or display any information, the address will pick out a piece of information from the memory where they it is kept. If the CPU wants to work on the program instructions or data in a specific location, it sends the address of that information by way of a signal to the memory. With this the memory sends back the data contained in that location. The processor stores the information in the memory by identifying the information and the address of the location where it is to be stored.

---

## 3.2 Execution of a Program

---

As described, a program is a set of instructions written in the machine language. A computer executes machine language programs as it is easily followed and further manipulates and reads machine language codes as these codes are directly understood by it. It is seen that a computer is a machine which is designed and built by tiny switches known as transistors. These switches have a unique property of getting connected together in such a way that the outputs from one switch will able to make another switch on or off. As the computer starts working, these switches will make each other on or off in the way they are connected and programmed.

Machine level language instructions are binary numbers 0 or 1. In this, each 0 or 1 is known as a bit. It follows that a machine language instruction is a series

of zeros and ones. Every series of sequence will result in a particular instruction. The data that the computer understands and controls is also transformed as binary numbers 0 or 1. Presently, in every computer, each memory location keeps a byte, which is a sequence of 8 bits.

It is seen that a computer can work directly with binary numbers 0 or 1 as switches can easily show these numbers. It means, to turn the switch ON, it is 1, while to turn the switch OFF, it is 0. Hence machine language instructions are kept in the memory as an arrangement of switches which gets turned ON or OFF. When a machine language instruction is kept into the CPU, then some switches get turned ON or OFF in the sequence that encodes such instructions. The processor is designed as such that it should respond to such an arrangement by executing the instruction it encodes.

It is the duty of the main memory to hold the machine language programs and data. Such programs and data are encoded as binary numbers 0 or 1. The CPU gets the machine language instructions from the memory one by one and executes them. This execution of instructions is mechanical, as the processor is manipulating it without thinking or understanding. The program it executes must be perfect, complete and clear-cut as the processor can do nothing more but easily execute what is written. The figure 3.1 shows the movement of information from the processor to the memory.

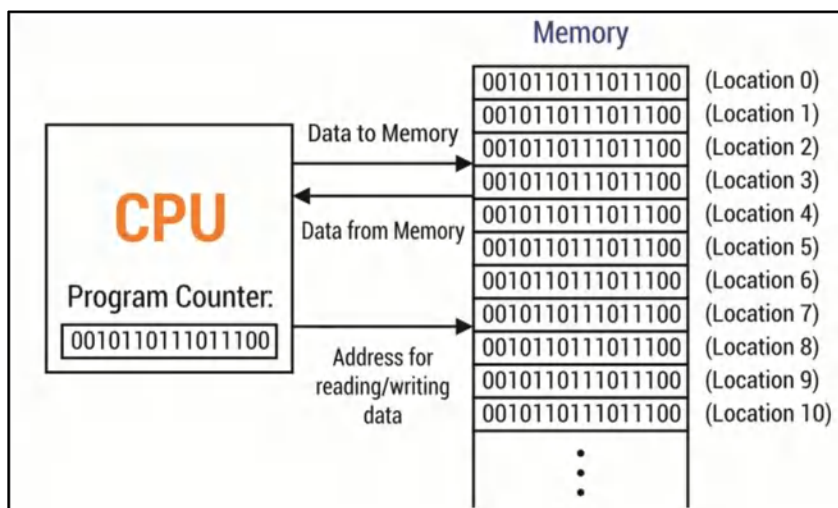


**Fig. 3.1 Data movement inside Computer**

With the view of machine language, the operation of the processor is clear. It executes a program that is stored as a sequence of machine language instructions in the main memory. The work of the program execution is done by the processor by frequently reading or fetching up of instructions from the



memory and then carries out or executes such instructions. This type of process off etching an instruction and executing it and then further fetching another instruction and further executing it is known as fetch-and-execute cycle.

### Check your progress 1

1. The minimum memory address from 0 to 256 is:
  - a. 1
  - b. 0
  - c. 255
  - d. 256
2. Which is a binary number, in terms of machine instructions?
  - a. 1
  - b. 2
  - c. 3
  - d. 7
3. Fetch-and-execute cycle is performed by the:
  - a. memory
  - b. processor
  - c. hard disk
  - d. USB drives

---

## 3.3 Fetch-Execute Cycle

---

It is a sequence of actions that are performed by the processor or central processing unit in order to execute the machine code instructions in a program. Initially, the processor shows the value of a program counter on the address bus. After that the processor fetches the instructions from the main memory through cache or pipeline data bus into the instruction register. Once the information is there in the instruction register, the data that forms the instruction is decoded and sent to the control unit which transmits a series of control signals to the processors requisite function units so as to enact the actions by the instruction, like:

- Reading values through the registers.
- Further transmitting these values to the ALU.
- Adding the values together and writing the result back to the register.

The address is passed on by the program counter to the next instruction which leads to the continuation of a cycle. The features of Fetch and interrupt are:

### Fetch

- Program Counter (PC) contains the address of the next instruction to be fetched.
- Address in the program counter gets copied to the Memory Address Register (MAR).
- Instruction is copied from the memory location in MAR and is transferred to Memory Buffer Register (MBR).
- Instruction is copied from the MBR is placed in Current Instruction Register (CIR).
- Program Counter is incremented for the next instruction to be fetched.

### Execute

- Address part of the instruction is placed in the MAR.
- Instructions are decoded and executed.
- Processor checks for interrupts.

The fetch execute cycle was initially invented by John von Neumann.

#### Check your progress 2

1. Fetch-and-execute cycle was invented by:
  - a. Pascalline
  - b. John von Neumann
  - c. Abacus
  - d. None of the above

---

## 3.4 Fetch-Decode-Execute Cycle

---

It is clear that a CPU will understand and perform instructions written only in the machine language. While doing so, both the inputs to the CPU are stored in the main memory. Now to function, the CPU requires a cycle that will fetch the instruction, decode the instructions and execute such instructions. This process is commonly known as fetch decode execute cycle.

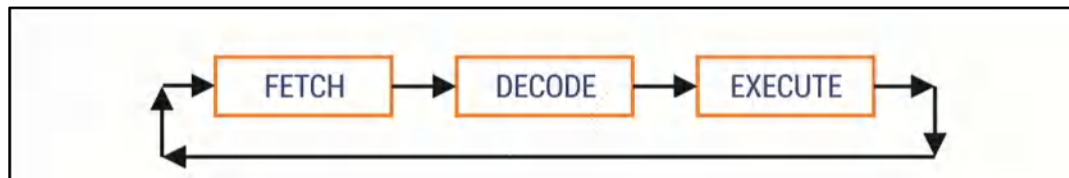


**Fig. 3.2 Fetch decode execute cycle**

The Fetch decode execute cycle as shown in figure 3.2 begins when an instruction which is sent from the memory to the Instruction Register along the data bus. The Instruction Register carries a special bit of patterns that enables the machine language to take out and send the instructions back to the Decoder. Further, during the next part of the cycle, the processor recognises the part of bit pattern that performs the correct operation. The operations of a Fetch decode execute cycle involves:

- Reading data from the memory.
- Storing data in the memory.
- Activating ALU to do mathematical operations.

After the operation, the cycle begins again with the next instruction as it is the work of the processor to find the next instruction. In this, the Program Counter holds the address of the current instruction. Every time when an instruction gets completed, the program counter will move ahead by one memory location.

### Check your progress 3

1. Fetch-and-execute cycle is performed by the:-
  - a. memory
  - b. processor
  - c. hard disk
  - d. USB drives
2. Instruction copied from Memory Buffer Register is kept in
  - a. Memory Address Register
  - b. Program Counter
  - c. Current Instruction Register
  - d. Memory Buffer Register
3. In Fetch decode execute cycle, an instruction is send from
  - a. memory to Instruction Register
  - b. memory to Current Instruction Register
  - c. memory to Program Counter
  - d. memory to Buffer Register

### 3.5 Program Execution with Fetch-Decode-Execute Cycle

To execute a program, the first step what the CPU does is to carry some data and instructions from the main memory which he stores into its own internal temporary memory location. To fetch the instruction, the address bus on which the address is placed for the next item to be fetched on the address bus is used by the CPU. In this, the address is moved from the main memory to the CPU by covering the data buses. Consider the figure 3.3 where a processor is ready to carry out the instruction. In this the instruction pointer has address 0100 which shows the instruction MOV AX, 0. The instruction is kept on the data bus by the memory. The instruction from the data bus is then copied by the CPU to the instruction register.

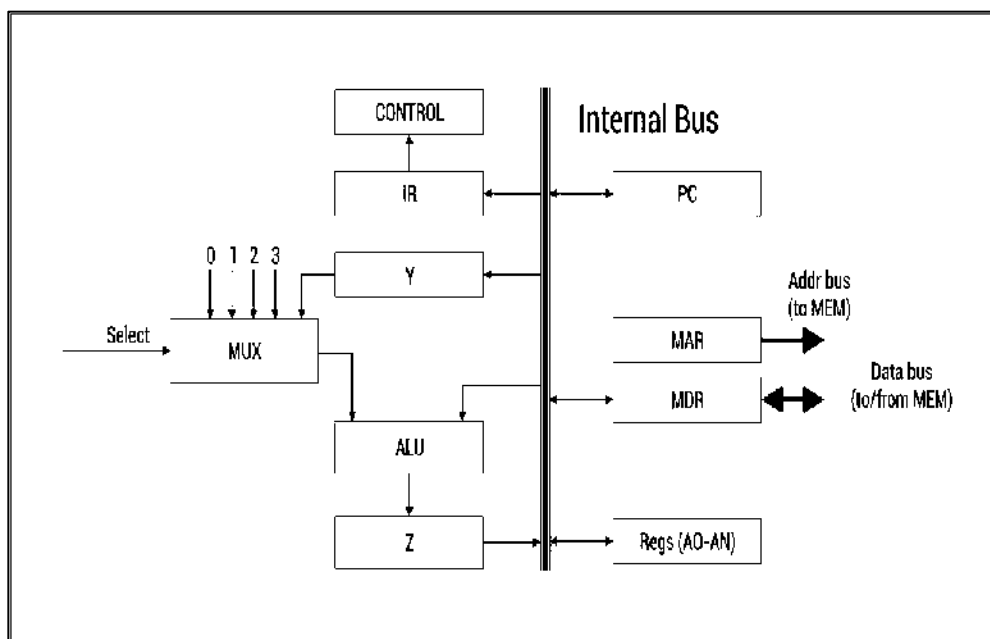


**Fig. 3.3 Fetching of Instructions**

The CPU then senses the fetched instruction to further decode it. For this, if needed, the processor can obtain an operand by the instruction. The instruction MOV AX 0. Stores the value 0 in A x location as seen in the figure above. Now, before the execution of the instruction the processor carries the fixed value 0 from the next memory location. In figure 3.3, the instruction from the instruction register is sent by the CPU or processor to the decoder. The instruction instructs the computer to store 0 in AX register. After decoding, the decode unit keeps all the details of the process.

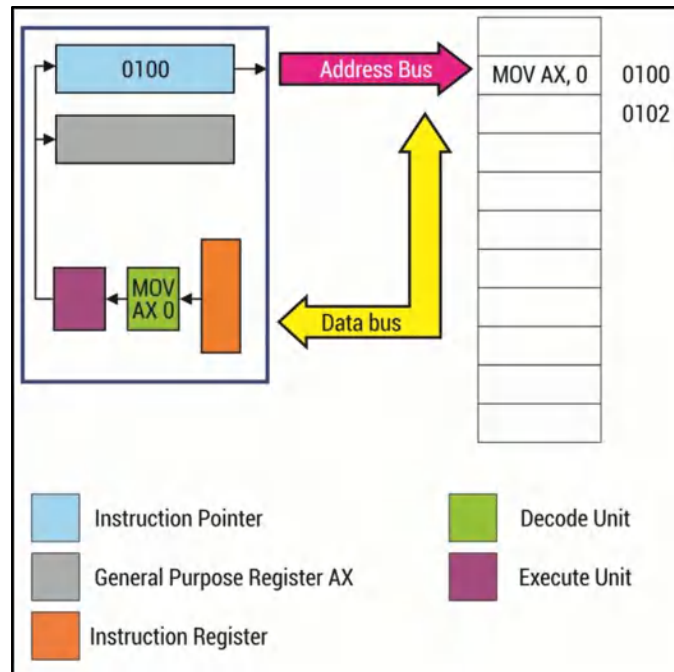


**Fig. 3.4 Decoding Instructions**

Finally, the CPU will execute the instruction 0 what is stored in register AX. In figure 3.5, the processor starts executing the MOV AX, 0 instructions. Then, the CPU will adjust the instruction pointer to point to the next instruction which needs to be executed that is kept at address 0102 location.

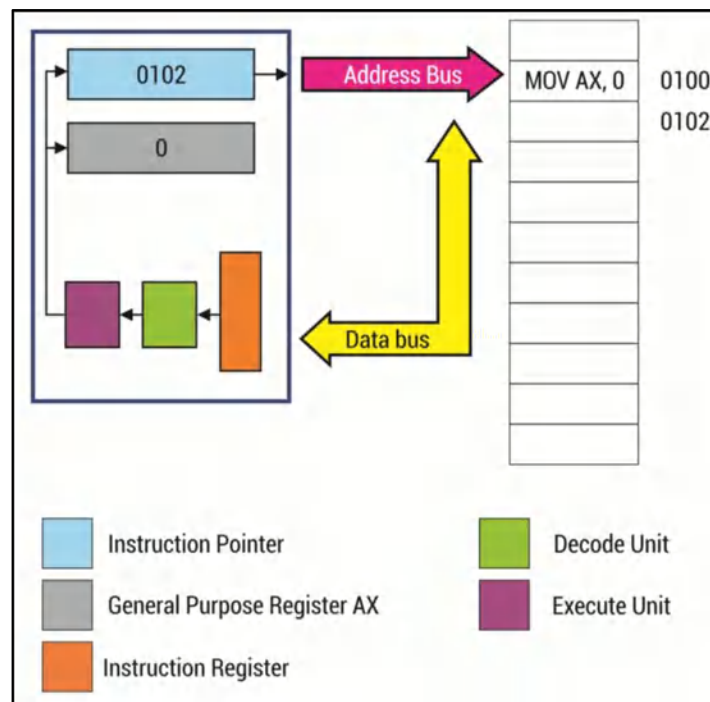


**Fig. 3.5 Executions of Instructions**

### Check your progress 4

1. To fetch an instruction, a CPU uses:
  - a. data bus
  - b. address bus
  - c. control bus
  - d. instruction bus
2. MOV AX, 0. is refer to as
  - a. data
  - b. information
  - c. instructions
  - d. register

---

## 3.6 Buffering Concept

---

Buffering is a term which is used in Computer hardware and programming which means screening of the processed data from its final place so that it can again be processed before being transferred to the database.

There are different kinds of buffering strategies:

- Characters which are written to or read from an un buffered location that are moved alone.
- Characters which are written to line buffered location that are moved to file when newline characters appear.
- Characters which are written to or read from the buffered location are moved to or from the file of any size.

### Check your progress 5

1. Buffering means:
  - a. saving of processed data
  - b. screening of processed data
  - c. deleting of processed data
  - d. mixing of processed data

---

## 3.7 Let Us Sum Up

---

**In this unit, we have learned:**

- The Computer understands the Machine language.
- The instructions are in the form of binary numbers 0 and 1.
- The processor processes the information by the way of a fetch execute cycle.
- In fetch decode and execute cycle, the instructions are fetched, then decoded and finally executed by the processor.
- Buffering is the checking of processed data from its location for execution before delivering it to the final database.

---

## 3.8 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-b), (2-a), (3-b)

**Check your progress 2**

**Answers:** (1-b)

**Check your progress 3**

**Answers:** (1-b), (2-b), (3-a)

**Check your progress 4**

**Answers:** (1-b), (2-c)

**Check your progress 5**

**Answers:** (1-b)

---

## 3.9 Glossary

---

1. **Addresses** - memory addresses show the location of the instruction in the ALU.
2. **Registers** - Are to keep data such as floating point numbers or addresses.
3. **Instructions** - These are machine instructions with the size same as an architecture's word.

---

## 3.10 Assignment

---

Give reasons why some processors can work faster than the other.

---

## 3.11 Activities

---

Write steps about how a CPU executes a set of instructions.

---

## 3.12 Case Study

---

Locate what a processor does at the end of a fetch decode execute cycle.

---

## 3.13 Further Readings

---

1. Computer Organization by Goldberg
2. Computer Architecture by Sarah Harris
3. Computer Mainframes by Almasi and Gottlieb



---

## Block Summary

---

In this block of Logical Architecture of a Computer, we have learned and summarized the main hardware with which a Computer processes the instructions.

Unit 1 of our block explains about the Memory and Processor. In this we have given an idea about what a processor is and its functional features. We explained the different hardware's for data storage and data manipulations.

This unit also explains about the role of a hard drive in accessing and keeping the data for longer use.

Unit 2 of this block explains the logical architecture of a computer system. It describes the necessary hardware used for data calculation, transmission and manipulation. This unit tells us about Registers, ALU, Internal Bus, Read/Write Cycle of Memory, Internal Bus, Control Unit, Cache Memory, etc.

Unit 3 of our block is an output of all the hardware's what we come across in unit 1 and 2. This unit explains the working of each hardware in shaping a computer program. . We see that once the hardware is ready, now it's time to use it by the way it understands. So this unit stresses on the execution, decoding and carrying out the information a computer system understands. It explains how a program is executed.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. What is a Computer Memory?
2. What is Buffering?
3. What are registers?
4. What do you mean by Program Counter?
5. What is the function of ALU?

### **Long Answer Questions**

1. What is the role of CPU in Computer architecture?
2. How is Data written on a Hard Drive?
3. How are registers different from Program Counters?

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# WEB APPLICATION DEVELOPMENT

PGDCA 202

## BLOCK 2: DATABASE AND STATE MANAGEMENT IN PHP

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# WEB APPLICATION DEVELOPMENT



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Mr. Sanjay Thapar

### **Language Editor**

Mr. K. J. Sharma

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.



Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# WEB APPLICATION DEVELOPMENT

## Contents

### **BLOCK 1: INTRODUCTION TO PHP**

#### **UNIT 1 INTRODUCTION TO WEB APPLICATION**

The architecture of a web application, static and dynamic web application, Installing WAMP.

#### **UNIT 2 PHP BASICS**

Embedding PHP Code in Your Web Pages, Commenting Your Code, Outputting Data to the Browser, PHP's Supported Data types, Identifiers, Variables, Constants, Expressions, String Interpolation, Control Structures.

### **BLOCK 2: ARRAY, FUNCTION AND EXPRESSION**

#### **UNIT 1 ARRAY IN PHP**

Array introduction, creating an array, adding and removing array element, determining array size and uniqueness sorting array,

#### **UNIT 2 FUNCTION IN PHP**

In built functions: String functions, array functions, mathematical functions, File System functions, Date and Time Functions, Miscellaneous Functions, User Defined Functions, arguments passing by reference , Argument passing by value

#### **UNIT 3 FUNCTION IN PHP**

Regular Expression, Error Handling Regular Expressions, Exception handling



## **BLOCK 3: OO AND FILE HANDLING IN PHP**

### **UNIT 1 OBJECT ORIENTED PHP**

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

### **UNIT 2 FILE AND DIRECTORY HANDLING**

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

---

## **BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP**

### **UNIT 1 PHP AND MYSQL DATABASE**

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

### **UNIT 2 STATE MANAGEMENT IN PHP**

Session, Cookies

---



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA 202

## **Web Application Development**

---

### **BLOCK 3: DATABASE AND STATE MANAGEMENT IN PHP**

---

#### **UNIT 1**

PHP And Mysql Database

03

#### **UNIT 2**

State Management in PHP

24

---

# **BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP**

---

## **Block Introduction**

PHP carries various state management features which will overcome certain problems which are taken from stateless Web which can be cookie. It is noted that cookie serves as piece of information which is send to server which further sends it to Web browser. In case, a web browser enters to subsequent page within similar domain, it will show server access with similar cookie which is quiet easy to implement. It is found that cookies are restricted to how information gets stored with lots of security problems. Apart from cookies, another state management technique in PHP is session. In PHP, session will keep all data on server. With this, sessions get more versatile which helps in securing cookies. Among the entire biggest potential problem related to sessions is that it depends on session ID which serves as an identifier which points to every unique set of session data. Such session ID gets access on every page through passing it from page to page or by keeping it in cookie.

In this block, we will detail about the basic of database software as Oracle or Sybase with respect to MYSQL database. The block will focus on MySQL extension and MySQL systems versions 4.1.3 with study about their characteristics. The concept of installation of host computer with WAMP, MAMP or XAMPP is also explained.

In this block, you will make to learn and understand about the basic of session cookies and their related techniques. The concept related to Database connectivity with MySQL and characteristics working of script in PHP using MySQL are also explained to you. You will be demonstrated practically about cookies in local host computer.

## **Block Objective**

**After learning this block, you will be able to understand:**

- About ways to use PHP in MySQL.
- Basic of Database connectivity with MySQL.
- Feature of selecting data.

Database and state  
management in PHP

- Characteristics about working in PHP using MySQL.
- Concept of Session.
- Idea about Cookies.

## **Block Structure**

**Unit 1:     PHP and MySQL Database**

**Unit 2:     State Management in PHP**

---

# UNIT 1: PHP AND MYSQL DATABASE

---

## Unit Structure

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension)**
- 1.3 Database connection**
- 1.4 Select data**
- 1.5 Insert, update and delete data in PHP using MySQL**
- 1.6 Let Us Sum Up**
- 1.7 Answers for Check Your Progress**
- 1.8 Glossary**
- 1.9 Assignment**
- 1.10 Activities**
- 1.11 Case Study**
- 1.12 Further Readings**

---

## 1.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About MySQL
- About Database connection

---

## 1.1 Introduction

---

It is seen that PHP will work with the help of database software which can be Oracle or Sybase which commonly present in MySQL database. To run any website using dynamic information, you need to have a database so that the information can be kept inside it. There are many types of things you can perform while interacting with PHP and MySQL. To run any PHP script, you only need three items that can be accessed by MySQL databases. Initially, you require web



server which can be a computer or can be web host along with web server software compatible with PHP and MySQL.

---

## 1.2 Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension)

---

PHP will be able to handle MySQL with the application of its extension. Many projects have been composed and written across the globe using PHP and MySQL. In a span of time, PHP deprecated MySQL extension and was replaced with alternatives as MySQLi which is improved MySQL version and PDO which is PHP Data Objects. There are three main API options when considering connecting to MySQL database server:

- PHP's MySQL Extension
- PHP's mysqli Extension
- PHP Data Objects (PDO)

It is found that MySQLi and PDO are object oriented and support Prepared Statements which can be used to support Transactions, Stored Procedures and many. The Prepared Statements are important for use of web application security since they can save from SQL injection. With the application of Prepared Statements, there will be no possibility of escape strings before insertion in Database. It is found that PDO supports many databases apart from MySQL.

### PHP's MySQL Extension

PHP MySQL Extension is an original extension which is created to help in development of PHP applications which interact with MySQL database. In this, mysql extension will show a process interface which is planned for applied only with MySQL versions. It can also be applied with MySQL 4.1.3 or newer versions, with restriction on latest MySQL server features. It is found that mysql extension source code is present in PHP extension directory of ext/mysql.

### PHP's mysqli Extension

PHP mysqli extension is an improved extension of MySQL which was created in order to make use of latest features of available in MySQL systems versions 4.1.3 and above that. This extension is covered with PHP versions 5 and later. The mysqli extension carries many advantages over key mysql extension as:

- Having Object-oriented interface

- Supporting for Prepared Statements
- Supporting for Multiple Statements
- Supporting for Transactions
- Advanced debugging capabilities
- Advanced server help

The mysqli extension is created with the help of PHP extension framework having its source code present in the directory ext/mysqli.

### Example:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
// Create connection
$conn = new mysqli($servername, $username, $password);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

### PDO

PDO is PHP Data Objects which is type of database abstraction layer particularly designed for PHP applications. It provides steady API for certain PHP application immaterial of database type server that connects to it. Normally it is found that when applying PDO API, selection of database is upto you which can be from Firebird to MySQL with small alteration in order to apply in PHP code.

With certain advantages related to simplicity, easy, portable API, POD has certain drawbacks as it fails to allow in using several advanced features which is present in current MySQL server version. PDO is carried out with the help of PHP extension framework where its source code available in directory ext/pdo.

### Example

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

try {
    $conn = new PDO("mysql:host=$servername;dbname=myDB", $username,
$password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
    echo "Connected successfully";
}
catch(PDOException $e)
{
    echo "Connection failed: " . $e->getMessage();
}
?>
```

### Check your progress 1

1. Advance MySQL extension released with PHP 5 is:
  - a. MySQL
  - b. mysql
  - c. mysqli
  - d. mysqlnd
2. Which version carries PHP mysqlnd:
  - a. PHP 5.0
  - b. PHP 4.1
  - c. PHP 4.2
  - d. PHP 5.3

## 1.3 Database connection

In order to work in PHP and MySQL, you need to initially establish a connection to MySQL database which is an important part as without script interconnection with database will fail all queries related to database.

It is good way to use database is initially to apply username, password and should give name to database at the starting of script code which can be altered anytime with no effort.

```
$username="your_username";$password="your_password";$database="your_database";
```

In the above code, you should change "your\_username", "your\_password" and "your\_database" with required MySQL username, password and database which can be used with your script.

Here, you will not worried about security risk of keeping your password in a file as PHP source code gets processed by server before it will; send data to browser which will help in safe guarding from visitors. Further, you need to connect PHP script to database using mysql\_connect PHP function which is:

```
mysql_connect(localhost,$username,$password);
```

The above script will direct the PHP to connect to MySQL database server at localhost. Once the connection is made, then choose the database which you want to use where your username had access which can be done by using the following command:

```
@mysql_select_db($database) or die( "Unable to select database");
```

This command will tell PHP so as to select database which is kept in variable \$database or in "your\_database". If the script fails to connect in such case it will stop executing and displays an error message as Unable to select database. Apart from these, there is another feature of PHP function which is:

```
mysql_close();
```

The above function is important since it closes the connection to database server. This script is required in order to stop the script from running. It is noted that many open MySQL connections will generate problem in the account. So it is better to close MySQL connection after all queries are done. After this, you can connect to server and select database which you wish to work with and start querying about it.

So it is noted that in order to run database connection, you have to initially create the database and add database user along with MySQL user before running the script on server. Many times you require connecting PHP website to database which can be easily performed with content management systems using config.php file as shown in example below:

```
<?php

//Sample Database Connection Script

//Setup connection variables, such as database username
//and password

$hostname="localhost";

$username="your_dbusername";

$password="your_dbpassword";

$dbname="your_dbusername";

$usertable="your_tablename";

$yourfield = "your_field";


//Connect to the database

$connection = mysql_connect($hostname, $username, $password);

mysql_select_db($dbname, $connection);


//Setup our query

$query = "SELECT * FROM $usertable";


//Run the Query

$result = mysql_query($query);


//If the query returned results, loop through
// each result

if($result)
{
8
```

```
while($row = mysql_fetch_array($result))
{
    $name = $row["$yourfield"];
    echo "Name: " . $name;

}
}

?>
```

In the above PHP script which is connected to database showing relevant fields for particular table is specified in the code above. To run database script on local computer, set the computer with Apache, MySQL, and PHP which can be easily done by installing WAMP, MAMP or XAMPP.

### Check your progress 2

1. The function which diagnose and display information about MySQL connection error is:
  - a. connect\_errno()
  - b. connect\_error()
  - c. mysqli\_connect\_errno()
  - d. mysqli\_connect\_error()

---

## 1.4 Select data

---

Till now we have seen creating and starting of database. Once the data is inserted into database, you can easily retrieve it. In SQL, you have to made a query with the help of SELECT statement which you have to execute by passing it to mysqli\_query() function so as to retrieve data from database or data table. The basic syntax of SELECT query is:

SELECT column\_name(s) FROM table\_name

With SQL query using SELECT statement, now you will execute SQL query by passing it to mysqli\_query() function in order to select tables records. For this you consider the data table showing details of persons:

```
+-----+-----+-----+-----+
| person_id | first_name | last_name | email_address |
+-----+-----+-----+-----+
|      1 | Sanjay    | Mathur   | Sanjaymathur@gmail.com |
|      2 | Amit     | Charan   | amitcharan@gmail.com |
|      3 | Renu    | Kapoor   | renukapoor@gmail.com |
|      4 | Rajiv    | Sood     | rajivsood@gmail.com |
|      5 | Amkrish  | Bajaj    | amrishbajaj@gmail.com |
+-----+-----+-----+-----+
```

We see that the PHP code in above example selects all data which is stored in persons table.

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
$link = mysqli_connect("localhost", "root", "", "demo");

// Check connection
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
```

```
// Attempt select query execution
$sql = "SELECT * FROM persons";
if($result = mysqli_query($link, $sql)){
    if(mysqli_num_rows($result) > 0){
        echo "<table>";
        echo "<tr>";
            echo "<th>person_id</th>";
            echo "<th>first_name</th>";
            echo "<th>last_name</th>";
            echo "<th>email_address</th>";
        echo "</tr>";
        while($row = mysqli_fetch_array($result)){
            echo "<tr>";
                echo "<td>" . $row['person_id'] . "</td>";
                echo "<td>" . $row['first_name'] . "</td>";
                echo "      "<td>"      . $row['last_name']      . "</td>";
                echo "<td>" . $row['email_address'] . "</td>";
            echo "</tr>";
        }
        echo "</table>";
        // Close result set
        mysqli_free_result($result);
    } else{
        echo "No records matching your query were found.";
    }
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);
}

// Close connection
mysqli_close($link);
?>
```



From the above example we see that, data returned by `mysqli_query()` function gets stored in `$result` variable. It is seen that every time `mysqli_fetch_array()` is called upon and returns next record from result set array. In this, the while loop is used to loop all the records in result set. Lastly, value of individual fields gets accessed from record either by passing field index or field name to `$row` variable such as:

- `$row['person_id']` or `$row[0]`
- `$row['first_name']` or `$row[1]`
- `$row['last_name']` or `$row[2]`
- `$row['email_address']` or `$row[3]`

### Check your progress 3

1. The compound datatype is:

- a. Array
- b. String
- c. Float
- d. Boolean

---

## 1.5 Insert, update and delete data in PHP using MySQL

---

### Delete data:

DELETE statement is applied in order to delete records from table. The syntax is:

DELETE FROM table\_name

WHERE some\_column = some\_value

We see that WHERE clause in DELETE syntax describes which record to be deleted. On removing the WHERE clause, you will see that all records gets deleted. Consider "Students" table:

| id | firstname | lastname | email                                                  | reg_date            |
|----|-----------|----------|--------------------------------------------------------|---------------------|
| 1  | Sanjay    | Mathur   | Sanjay@gmail.com                                       | 2014-11-21 14:26:15 |
| 2  | Monica    | Goel     | <a href="mailto:monica@gmail.com">monica@gmail.com</a> | 2014-11-22 10:22:30 |
| 3  | Naine     | Malik    | naine@gmail.com                                        | 2014-11-25 10:48:23 |

In the above example we see that the record of student with id=3 gets deleted in "Students" table:

**Example:**

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// sql to delete a record
$sql = "DELETE FROM Students WHERE id=3";
if ($conn->query($sql) === TRUE) {
    echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . $conn->error;
}

$conn->close();
?>
```

**Example: MySQLi Procedural**

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
```

Database and state  
management in PHP

```
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
// sql to delete a record
$sql = "DELETE FROM Students WHERE id=3";
if (mysqli_query($conn, $sql)) {
    echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

### Example (PDO)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";
```

```
try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
    // sql to delete a record
    $sql = "DELETE FROM Students WHERE id=3";
    // use exec() because no results are returned
    $conn->exec($sql);
    echo "Record deleted successfully";
}
catch(PDOException $e)
{
    echo $sql . "<br>" . $e->getMessage();
}
$conn = null;
?>
```

After the record is deleted, the table will look like this:

| id | firstname | lastname | email            | reg_date               |
|----|-----------|----------|------------------|------------------------|
| 1  | Sanjay    | Mathur   | Sanjay@gmail.com | 2014-11-21<br>14:26:15 |
| 2  | Monica    | Goel     | monica@gmail.com | 2014-11-22 10:22:30    |

### Update Data:

To update data in MySQL Table with the help of MySQLi and PDO, consider an UPDATE statement which will update present records in the table. Consider the syntax:

UPDATE table\_name

SET column1=value, column2=value2,...

WHERE some\_column=some\_value

In the above UPDATE syntax, WHERE clause is applied in UPDATE syntax in order to describe which record to be updated. To erase WHERE clause, you will find that all records get updated. For this, consider Students table:

| id | firstname | lastname | email            | reg_date            |
|----|-----------|----------|------------------|---------------------|
| 1  | Sanjay    | Mathur   | Sanjay@gmail.com | 2014-11-21 14:26:15 |
| 2  | Monica    | Goel     | monica@gmail.com | 2014-11-22 10:22:30 |

| id | firstname | lastname         | email | reg_date            |
|----|-----------|------------------|-------|---------------------|
| 1  | John Doe  | john@example.com |       | 2014-10-22 14:26:15 |
| 2  | Mary Moe  | mary@example.com |       | 2014-10-23 10:22:30 |

We see from the this is that the record with id=2 in students table get adjusted.

### Example (MySQLi Object-oriented)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "UPDATE Students SET lastname='Doe' WHERE id=2";
if ($conn->query($sql) === TRUE) {
    echo "Record updated successfully";
} else
{
```

```
    echo "Error updating record: " . $conn->error;
}
```

```
$conn->close();
```

```
?>
```

### Example (MySQLi Procedural)

```
<?php
```

```
$servername = "localhost";
```

```
$username = "username";
```

```
$password = "password";
```

```
$dbname = "myDB";
```

```
// Create connection
```

```
$conn = mysqli_connect($servername, $username, $password, $dbname);
```

```
// Check connection
```

```
if (!$conn) {
```

```
    die("Connection failed: " . mysqli_connect_error());
```

```
}
```

```
$sql = "UPDATE Students SET lastname='Doe' WHERE id=2";
```

```
if (mysqli_query($conn, $sql)) {
```

```
    echo "Record updated successfully";
```

```
} else {
```

```
    echo "Error updating record: " . mysqli_error($conn);
```

```
}
```

```
mysqli_close($conn);
```

```
?>
```

### Example (PDO)

```
<?php

$servername = "localhost";

$username = "username";

$password = "password";

$dbname = "myDBPDO";

try {

    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);

    // set the PDO error mode to exception

    $conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);

    $sql = "UPDATE Students SET lastname='Doe' WHERE id=2";

    // Prepare statement

    $stmt = $conn->prepare($sql);

    // execute the query

    $stmt->execute();

    // echo a message to say the UPDATE succeeded

    echo $stmt->rowCount() . " records UPDATED successfully";

}

catch(PDOException $e)

{

    echo $sql . "<br>" . $e->getMessage();

}

$conn = null;

?>
```

Once the record is updated, the table will look like:

| id | firstname | lastname | email            | reg_date               |
|----|-----------|----------|------------------|------------------------|
| 1  | Sanjay    | Mathur   | Sanjay@gmail.com | 2014-11-21<br>14:26:15 |
| 2  | Monica    | Goel     | monica@gmail.com | 2014-11-22 10:22:30    |

### Insert Data:

Once the database and table is framed, now we can insert data by following certain syntax rules:

- Query in SQL should be quoted in PHP
- String values in SQL query should be quoted
- Numeric values be excluded from quoted
- NULL word should not be quoted

We see that INSERT INTO statement is applied in order to add new records to MySQL table:

INSERT INTO table\_name (column1, column2, column3,...)

VALUES (value1, value2, value3,...)

The examples given below will add new record to Students table:

### Example (MySQLi Object-oriented)

```
<?php
$servername = "localhost"
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
```



Database and state  
management in PHP

```
$sql = "INSERT INTO Students (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com)";
if ($conn->query($sql) === TRUE) {
    echo "New record created successfully";
} else

{
    echo "Error: " . $sql . "<br>" . $conn->error;
}
$conn->close();
?>
```

### Example (MySQLi Procedural)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
```

```
$sql = "INSERT INTO Students (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
if (mysqli_query($conn, $sql)) {
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

### Example (PDO)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
    $sql = "INSERT INTO Students (firstname, lastname, email)
VALUES ('Nishit', 'Mathur', 'nishit@gmail.com')";
    // use exec() because no results are returned
    $conn->exec($sql);
    echo "New record created successfully";
}
```

```
catch(PDOException $e)
{
    echo $sql . "<br>" . $e->getMessage();
}
$conn = null;
?>
```

### Check your progress 4

1. In SQL statement, inserting new data is done by:
  - a. INSERT
  - b. UPDATE
  - c. ADD
  - d. INSERT NEW
2. Which commands creates database named “school”:
  - a. CREATE ?I school
  - b. CREATE DATABASE school
  - c. DATABASE /school
  - d. DATABSE school

---

## 1.6 Let Us Sum Up

---

In this unit we have learnt that PHP will work using database software that can be Oracle or Sybase which is available in MySQL database. It seen that PHP will handle MySQL with application of its extension as certain projects carried and written across globe using PHP and MySQL.

It is found that PHP depreciate MySQL extension and was removed with alternatives as MySQLi which is improved MySQL version and PDO which is PHP Data Objects. The PHP mysqli extension is improved extension of MySQL created to make use of latest features of available in MySQL systems. PHP Data Objects is type of database abstraction layer designed for PHP applications that

gives steady API for certain PHP application immaterial of database type server that connects to it.

PHP and MYSQL  
database

It is noted that to run database script on local computer, you need to set computer with Apache, MySQL, and PHP by installing WAMP, MAMP or XAMPP.

---

## 1.7 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-c), (2-d)

**Check your progress 2**

**Answers:** (1-c)

**Check your progress 3**

**Answers:** (1-a)

**Check your progress 3**

**Answers:** (1-a), (2-b)

---

## 1.8 Glossary

---

1. **MySQL** - It is a database server applied in PHP.
2. **localhost** - It can be a computer at home or in office used to run particular web pages for websites.
3. **Database** - It is a collection of information that get stored in server and can be taken on calling it.

---

## 1.9 Assignment

---

What is an improved extension of MySQL?

---

## 1.10 Activities

---

Create an activity on PHP Data Objects.

---

## 1.11 Case Study

---

Does your institute carries database designed in PHP? Comment

---

## 1.12 Further Readings

---

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22.
2. "Using PHP from command line", PHI, Dixit, 2009-09-11.

---

## UNIT 2: STATE MANAGEMENT IN PHP

---

### Unit Structure

- 2.0 Learning Objectives
- 2.1 Introduction
- 2.2 Session
- 2.3 Cookies
- 2.4 Let Us Sum Up
- 2.5 Answers for Check Your Progress
- 2.6 Glossary
- 2.7 Assignment
- 2.8 Activities
- 2.9 Case Study
- 2.10 Further Readings

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About Session.
- About Cookies.

---

### 2.1 Introduction

---

In PHP, there are certain sessions that take care and keep record of genuine users. It serves as building block for websites with many user activities. In absence of sessions, everyone would be an unknown visitor. With fast evolution of Web programming, stateless nature of protocol brings problems to Web applications which maintain state across protocol requests. With this the role of session management emerges as straight response to certain problem with present method being applied by PHP developers in context to Web application requirement to track state during occurrence of various HTTP requests.

---

## 2.2 Session

---

In PHP, session is a way by which you can keep your information that is applied across multiple pages. Similar to cookie, in session, the information is not stored on user's computer. On working with application and opening it for certain alterations further can be closed. It is same as Session where computer will be able to identify you. It knows when you start an application and when you close. In case of web there appears one problem that web server is unable to find you and your activities as protocol address fails to maintain the state.

It is noted that session variables will be able to solve problem with storage of user information that can be applied across various pages. By default, session variables last till user closes the browser. Hence, session variables keep the information of single user that is present to all pages in single application. We see that session starts with `session_start()` function where session variables are set using PHP global variable: `$_SESSION`.

Consider a new page "project\_session1.php" where we start with new PHP session and set session variables as:

```
<?php
// Session Start
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// Set session variables
$_SESSION["favpicture"] = "junun";
$_SESSION["favplant"] = "rose";
echo "Session variables are set.";
?>
</body>
</html>
```

Output:

Session variables are set

### Creating PHP session:

Now we will see another page "project1\_session2.php" where from we can access session information set on first page ("project\_session1.php"). We see that session variables are not passed singly to every new page rather than are recovered from session what gets open at start of every page (session\_start()). Further, all session variable values are kept in global \$\_SESSION variable as shown in the code:

```
<?php
session_start();
?>

<!DOCTYPE html>

<html>

<body>

<?php
// Echo session variables that were set on previous page
echo "Favorite picture is " . $_SESSION["favpicture"] . "<br>";
echo "Favorite plant is " . $_SESSION["favplant"] . ".";
?>

</body>

</html>
```

Output:

Favorite picture is junun.

Favorite plant is rose.

### Changing PHP Session Variable:

You can change the session variable simply by overwriting it as shown in the code below:

```
<?php
session_start();
```



```
?>
<!DOCTYPE html>

<html>

<body>

<?php

// to change session variable, overwrite it

$_SESSION["favpicture"] = "junun";

print_r($_SESSION);

?>

</body>

</html>
```

Output:

On running the above code, we will get the output as:

Array ( [favpicture] => junun [favplant] => rose )

### Check your progress 1

1. Session data can be kept in \_\_\_\_\_ways.
  - a. 2
  - b. 4
  - c. 3
  - d. 5
2. The default PHP session name is:
  - a. PHPSESSID
  - b. PHPSESID
  - c. PHPSESSIONID
  - d. PHPIDSESSION

3. \_\_\_\_\_function is used in starting of a session.

- a. start\_session()
- b. session\_start()
- c. session\_begin()
- d. start\_\_begin,session()

---

## 2.3 Cookies

---

In PHP, cookies also is used to find a user which is a type of small file where server embossed on user's computer. In this, every time, similar computer will requests for page having a browser which sends the cookies. PHP helps in creating and retrieving cookie values. A cookie is created by setcookie() function. The Syntax of cookies is:

```
setcookie(name, value, expire, path, domain, secure, httponly);
```

In this only name parameter are used while others are optional.

### Create/Retrieve a Cookie

You can create and retrieve cookies in PHP. The example below creates cookie as "user" having value "Sanjay Mathur" that expires after 30 days. Here "/" shows that cookie is present in full website. Then the value of cookie "user" is retrieved using isset() function to find when cookie is set as:

```
<?php
$cookie_name = "user";
$cookie_value = "Sanjay Mathur";
setcookie($cookie_name, $cookie_value, time() + (30), "/"); //
?>

<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name])) {
    echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
```

```
echo "Cookie " . $cookie_name . " is set!<br>";  
echo "Value is: " . $_COOKIE[$cookie_name];  
}  
?>  
</body>  
</html>
```

Output:

Cookie named 'user' is not set!

We see that here we will reload the page in order to see the value of cookie which automatically encoded while sending cookies which further will be decoded when received.

### Modify a Cookie Value

In PHP, you can modify a cookie simply by setting cookie by `setcookie()` function as shown in following code:

```
<?php  
$cookie_name = "user";  
$cookie_value = "Sanjay Mathur";  
setcookie($cookie_name, $cookie_value, time() + (30), "/");  
?>  
  
<html>  
  
<body>  
  
<?php  
if(!isset($_COOKIE[$cookie_name])) {  
    echo "Cookie named '" . $cookie_name . "' is not set!";  
} else {  
    echo "Cookie '" . $cookie_name . "' present<br>";  
    echo "Value is: " . $_COOKIE[$cookie_name];  
}  
?>
```

```
</body>
```

```
</html>
```

Output:

Cookie 'user' present

Value is: Sanjay Mathur

From this, we see that we can reload the page anytime to see new value of cookie.

### Delete a Cookie

Further in PHP, you can easily delete a cookie with `setcookie()` function having expiration date in past:

```
<?php
```

```
// set the expiration date
```

```
setcookie("user", "", time() - 36);
```

```
?>
```

```
<html>
```

```
<body>
```

```
<?php
```

```
echo "Cookie 'user' is deleted.";
```

```
?>
```

```
</body>
```

```
</html>
```

Output:

Cookie 'user' is deleted.

### Check your progress 2

1. Which among the following environment variable is used in accessing cookies:

- a. HTTP\_COOKIE\_VARS
- b. COOKIE\_HTTP\_VARS
- c. VARS\_COOKIE\_HTTP
- d. COOKIE\_VARS\_HTTP

2. The correct declare Session syntax of cookies is:

- a. `$_mark['username']='SESSION';`
- b. `$_SESSION['username']='mark';`
- c. `$_SESSION['declrusername']='mark';`
- a. d. `$_mark['usernamedeclr']='SESSION';`

---

## 2.4 Let Us Sum Up

---

In this unit we have learnt that there are certain sessions in PHP which takes care and keep record of genuine users and serves building block for websites with many user activities. In PHP, session is a way by which you can keep your information that is applied across multiple pages which is same as cookie where information is not stored on user's computer.

It is noted that in PHP, cookies are used to find user that serves as small file where server is embossed on user's computer.

---

## 2.5 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 -c), (2 -d), (3 -d), (4 -a)

**Check your progress 2**

**Answers:** (1 -d), (2 -d)

**Check your progress 3**

**Answers:** (1 -a), (2 -d), (3 -a), (4 -d)

---

## 2.6 Glossary

---

1. **Cookies** - It is a program that keeps information and is stored in user's browser memory.
2. **Session** - It is a logical object that will save data across subsequent http requests.

---

## 2.7 Assignment

---

Explain the advantages of Cookies.

---

## 2.8 Activities

---

Write the program which describes session in PHP.

---

## 2.9 Case Study

---

Compile and run the program which shows deletion of Cookies.

---

## 2.10 Further Readings

---

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22.
2. "Using PHP from command line", PHI, Dixit, 2009-09-11.

---

## Block Summary

---

In this block, you will understand about the basic of PHP Data Objects with concept related to database abstraction layer. The block gives an idea on ways to use PHP in MySQL with study about their characteristics. The examples related to concept of selecting data with their working characteristics are also discussed.

In this block, you will understand about the basic of handling MySQL with application of its extension. The concept related to inserting, updating and deleting of data in PHP using MySQL is also detailed. You will be demonstrated practically about data selection and its coding.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. What is session cookie?
2. What do you mean by database in MySQL?
3. What is the function of fopen() in PHP?
4. Highlight predefined classes in PHP?
5. Explain database abstraction layer?

### **Long Answer Questions**

1. How can we destroy a session in PHP?
2. How do you define a constant?
3. What are the different errors in PHP?



Enrolment No.

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |

3. Any Other Comments



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# WEB APPLICATION DEVELOPMENT

PGDCA 202

## BLOCK 2: Array, Function and Expression

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# WEB APPLICATION DEVELOPMENT



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Mr. Sanjay Thapar

### **Language Editor**

Mr. K. J. Sharma

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!





# WEB APPLICATION DEVELOPMENT

## Contents

### **BLOCK 1: INTRODUCTION TO PHP**

#### **UNIT 1 INTRODUCTION TO WEB APPLICATION**

The architecture of a web application, static and dynamic web application, Installing WAMP.

#### **UNIT 2 PHP BASICS**

Embedding PHP Code in Your Web Pages, Commenting Your Code, Outputting Data to the Browser, PHP's Supported Data types, Identifiers, Variables, Constants, Expressions, String Interpolation, Control Structures.

---

### **BLOCK 2: ARRAY, FUNCTION AND EXPRESSION**

#### **UNIT 1 ARRAY IN PHP**

Array introduction, creating an array, adding and removing array element, determining array size and uniqueness sorting array,

#### **UNIT 2 FUNCTION IN PHP**

In built functions: String functions, array functions, mathematical functions, File System functions, Date and Time Functions, Miscellaneous Functions, User Defined Functions, arguments passing by reference , Argument passing by value

#### **UNIT 3 FUNCTION IN PHP**

Regular Expression, Error Handling Regular Expressions, Exception handling

---



## **BLOCK 3: OO AND FILE HANDLING IN PHP**

### **UNIT 1 OBJECT ORIENTED PHP**

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

### **UNIT 2 FILE AND DIRECTORY HANDLING**

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

## **BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP**

### **UNIT 1 PHP AND MYSQL DATABASE**

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

### **UNIT 2 STATE MANAGEMENT IN PHP**

Session, Cookies



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA 202

## Web Application Development

---

### BLOCK 2: ARRAY, FUNCTION AND EXPRESSION

---

#### UNIT 1

Array in PHP

02

#### UNIT 2

Function in PHP

14

#### UNIT 3

Expression

31

---

## **BLOCK 2: ARRAY, FUNCTION AND EXPRESSION**

---

### **Block Introduction**

Arrays are basically applied in order to store and organize data efficiently and quickly within no time. Hence an array having numeric index is called an indexed array while array having name is called an associative array.

In this block, we will detail about the basic of Arrays with their features associated with storing and organizing of data with required efficiency. The block will focus on the study and concept of Error handling and Exception handling concept in respect to code execution with knowledge on exceptional conditions. The students will give an idea on Perl style regular expressions with comparison with POSIX counterparts.

In this block, you will make to learn and understand about the basic of PHP in built functions with their features. The concept related to structure of POSIX regular expression with knowledge about typical arithmetic expression is well explained to you. You will be demonstrated practically about POSIX regular expression implementation.

### **Block Objective**

**After learning this block, you will be able to understand:**

- About File systems-structure and partition
- Basic of Disk space allocation
- Features of Disk scheduling
- Concept of I/O Hardware and Drivers
- Detailed about DMA controlled I/O

### **Block Structure**

**Unit 1: Array in PHP**

**Unit 2: Function in PHP**

**Unit 3: Expression**

---

## UNIT 1: ARRAY IN PHP

---

### Unit Structure

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 creating an array**
- 1.3 Adding and removing array element**
- 1.4 Determining array size**
- 1.5 Uniqueness sorting array**
- 1.6 Let Us Sum Up**
- 1.7 Answers for Check Your Progress**
- 1.8 Glossary**
- 1.9 Assignment**
- 1.10 Activities**
- 1.11 Case Study**
- 1.12 Further Readings**

---

### 1.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About array creation
- About adding array element
- About finding array size

---

### 1.1 Introduction

---

Array is an important part of any programming language since it carries important facilities and built-in functionality which is required by any programming language. In PHP, there exists strong set of efficient ways that will help in dealing with such arrays. Arrays are basically applied in order to store and organize data efficiently and quickly within no time. It serves as an important data

a type which is present in any programming language as it is easily described as ordered list of elements. Here you can access single elements simply by referring their index position which is available inside an array. Its position is shown by numeral or alphabet. Hence an array having numeric index is called an indexed array while array having name is called an associative array.

---

## 1.2 Creating an array

---

Array serves as a map since every key is mapped with some value. In a programming language, the role of arrays is to store data in cases when variable results in many values. The `array()` function is applied to create an array. We can create array in PHP by two ways::

Way 1. Creating an array by passing the value in array function

```
<?php
$arr = array(
    "Sanjay",
    "Rohit",
    "Amit"
);
?>
```

We can also create array as:

```
<?php
$arr = array(
    "Sanjay",
    Rohit,
    "Amit",
    array(
        "Delhi",
        "Mumbai",
        "Pune"
    )
);
```

Array, function  
and expression

```
?>
```

Way 2. Creating an array by creating an index

```
<?php
```

```
$arr[] = "Sanjay";
```

```
$arr[] = "Rohit";
```

```
$arr[] = "Amit";
```

```
?>
```

We see that we can create three types of Arrays available in PHP which are:

### **Indexed array**

In PHP, an indexed array is a type of array which is created inside an index. It can be created in two ways as shown.

```
<?php
```

```
$arr = array(
```

```
    "Neha",
```

```
    "Preeti",
```

```
    "Anika"
```

```
);
```

```
Print_r($arr);
```

```
?>
```

### **Associative array**

In PHP, associative array is a type of array having a key that is associated with every value where a string can be used as index of array.

```
<?php
```

```
$employee = array(
```

```
    "name" => "Sanjay",
```

```
    "address" => "Pune",
```

```
    "batch" => "1998"
```

```
);
```

```
Print_r($employee);
```

?>

### **Multidimensional array**

In PHP, multidimensional array is a type of array having another array as value as shown:

```
<?php
// A two-dimensional array:
$scooters = array(
    array(
        "Mahindra",
        100,
        80
    ),
    array(
        "Bajaj",
        80,
        60
    ),
    array(
        "LML",
        100,
        90
    )
);
?>
```

### **Example for an Indexed array:**

```
<?php
echo "Index Array";

$arr = array(
    "Sanjay",
```



Array, function  
and expression

```
"Rohit",  
"Amit"  
);  
Print_r($arr);  
$array[] = "Sanjay";  
$array[] = "Rohit";  
$array[] = "Amit";  
Print_r($array);  
?>
```

Output:

```
<?php  
isArray Array(  
    [0] => Sanjay[1] => Rohit[2] => Amit  
) Array(  
    [0] => Sanjay[1] => Rohit[2] => Amit  
)  
?>
```

**Example for an Associative array:**

```
<?php  
echo "Associative Array";  
$array1 = array(  
    "name" => "Sanjay",  
    "address" => "Pune",  
    "batch" => "1998"  
);  
print_r($array1);  
echo "";  
foreach ($array1 as $key => $value) {  
    echo "array1[$key] = $value";  
}
```

```

}
echo "";
?>

Output:

<?php
AssociativeArray Array(
    [name] => Sanjay[address] => Pune[batch] => 1998
) array1[name] = Sanjayarray1[address] = Punearray1[batch] = 1998
?>

```

### Check your progress 1

1. Array contains:

- a. string
- b. number
- c. data
- d. all of above

---

## 1.3 Adding and removing array element

---

While working with arrays, it not certain that the particular will fit or is not required. Every time you need to check by adding or removing elements to the array. In PHP, we see that there are certain built-in functions which are framed that will attach or detach elements right from beginning or ends in numerically indexed array.

### Adding an Element in an array

Initially we will start with how to add elements to an array. We can add elements in an array by two simple ways. First we can “push” element onto array with the help of `array_push` function as it is needed when you want to add multiple elements to end of array at the same time. To use `array_push` function, you need to have the following code:

view plaincopy to clipboardprint?

```
$myarray = array('cow', 'bull');
```

```
array_push($myarray,'snake');
```

With the help of above code, we can create a new array which is \$myarray in which we add single element. After adding, we used array\_push function so as to add another new element to an array. We see that the new elements gets added to the end of array in similar order as listed in the function. Hence the order of elements in an array will now be 'cow', 'bull', 'snake'.

On adding single new element to existing array, simply apply empty brackets, hence rather than applying array\_push(\$myarray,'snake') we use \$myarray[] = 'snake'.

### Removing an Element from an array

To make the last element out of an array, you need array\_pop function. With the help of array\_pop function, you can easily remove last element in array and will able to keep particular value in new variable. Consider an example for this:

[view plaincopy to clipboardprint?](#)

```
$filename = 'style.css';  
$filename = explode('.', $filename);  
$extension = array_pop($filename);
```

The above code will convert string style.css into an array and gets split up wherever dot comes in the string. Once that appears we will finish with array having elements 'style' and 'css' in particular order. After it we use array\_pop function so as to remove 'css' from array and keep it as new variable \$extension.

### Shifting and Unshifting an Element in an array

By now you will be able to have an idea regarding the adding of new elements and removing of elements from end of an array. Many times there are possibilities that you need to add element at the beginning of array. In PHP, there are certain options available where there exists two functions which can be applied in order to add new elements or remove elements from beginning of array.

For this, you will apply PHP built-in function array\_unshift in order to add element in the start of array as shown in example below.

[view plaincopy to clipboardprint?](#)

```
$myarray = array('cow','bull');  
array_unshift($myarray,'cat');
```

From the above example, we see that new element gets added at the beginning of array and it will result as 'cat', 'cow', 'bull'.

You can remove the first element from an array using `array_shift` function as shown in example:

view plaincopy to clipboardprint?

```
$myarray = array('cow','bull', 'cat');  
array_shift($myarray)
```

By using `array_shift` function, you will find that first element cow gets removed and the array is left with 'bull', 'cat'.

### Check your progress 2

1. Array `array_push` function is required in order to add:

- a. single element
- b. two elements
- c. three elements
- d. many elements

2. Array `array_shift` function is required to remove:

- a. first element
- b. middle elements
- c. second elements
- d. last elements

---

## 1.4 Determining array size

---

In PHP, the `sizeof()` function will return number of elements in an array which is alias of `count()` function. The syntax of size array is `sizeof(array,mode)`. The `sizeof` function in an array is used to count number of words which is available in the paragraph. It can be determined by:

- Storing paragraph of text in string variable
- Breaking variable and creating array using `split` function with space as delimiter

Array, function  
and expression

- Counting number of elements available in an array

Consider an example program where the arrays are counted recursively:

```
<?php
$fourwheelers=array
(
    "Jeep"=>array
    (
        "JC60",
        "JC90"
    ),
    "Toyota"=>array
    (
        "X7",
        "X9"
    ),
    "Maruti"=>array
    (
        "Cidan"
    )
);
echo "Normal count: " . sizeof($fourwheelers)."<br>";
echo "Recursive count: " . sizeof($fourwheelers,1);
```

If you run the above program, you will get an output as:

Normal count: 7

Recursive count: 16

### Check your progress 3

1. In an array, sizeof() function is used to:
  - a. add number of elements in an array
  - b. return number of elements in an array
  - c. overwrite number of elements in an array
  - d. none of above

## 1.5 Uniqueness sorting array

In PHP, the array\_unique() function will able to remove double values from an array. It is noted that when two or more array values are similar, then first will be kept while other will be removed. The returned array will hold the initial array item's key type. The syntax is array\_unique(array).

Consider an example program of removing duplicate values from an array:

```
<?php
$a=array("a"=>"table","b"=>"chair","c"=>"table");
print_r(array_unique($a));
?>
```

On running the above program, we will get an output as:

Array ( [a] => table [b] => chair )

### Check your progress 4

1. In an array, array\_unique() function is used to:
  - a. add double values from an array
  - b. remove double values from an array
  - c. add single value in an array
  - d. remove single value from an array

---

## 1.6 Let Us Sum Up

---

In this unit we have learnt that arrays stores and organizes data efficiently and quickly within no time and carries numeric index and name index. It is seen that array serves as map as every key is mapped with value which gets stored in an array.

In PHP, we see that there are certain built-in functions which are framed that will attach or detach elements right from beginning or ends in numerically indexed array.

The sizeof() function is applied in order to return the number of elements in an array which is same as count() function with syntax as sizeof(array,mode). It is found that array\_unique() function removes double values from an array where on having two or more array values, the first will be there while other gets removed.

---

## 1.7 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-b)

**Check your progress 2**

**Answers:** (1-d), (2-a)

**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-b)

---

## 1.8 Glossary

---

1. **Array** - A homogeneous container of numerical elements that occupies fixed amount of memory.
2. **Attribute** - It is the property of an object which will able to access by using obj.attribute,

3. **Instance** - It is a class definition which shows blueprint for framing object.
4. **Sizeof() function** - In an array, sizeof() function returns number of elements.

---

## 1.9 Assignment

---

Write short note on arrays?

---

## 1.10 Activities

---

Study about the importance of sizeof() function in an array.

---

## 1.11 Case Study

---

Discuss the importance of array\_unique() function in an array.

---

## 1.12 Further Readings

---

1. Basic of Arrays, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, 1996.
2. Array Programming in PHP by Dan Sydow, 2003.
3. Introduction to PHP, Chandresh Shah, 2010.
4. Features of Arrays and Function with Integrated Approach by B. Rahim, 2004.
5. Programming and Principles in PHP, Oxford, 2010, Jerry Smith.



---

## UNIT 2: FUNCTION IN PHP

---

### Unit Structure

#### 2.0 Learning Objectives

#### 2.1 Introduction

#### 2.2 In built functions

2.2.1 String functions

2.2.2 Array functions

2.2.3 Mathematical functions

2.2.4 File System functions

2.2.5 Date and Time Functions

2.2.6 Miscellaneous Functions

2.2.7 User Defined Functions

2.2.8 Arguments passing by reference

2.2.9 Argument passing by value

#### 2.3 Let Us Sum Up

#### 2.4 Answers for Check Your Progress

#### 2.5 Glossary

#### 2.6 Assignment

#### 2.7 Activities

#### 2.8 Case Study

#### 2.9 Further Readings

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About PHP functions.
- The features of Argument function.
- Various built-in functions in PHP.

---

## 2.1 Introduction

---

In PHP, function resembles with the name given by us for bunch of code which to be executed whenever it is required by us. By using functions in the code, you will save lot of time and effort and can easily be able to write a code which is much readable. On creating a function, initially you require to give a name to it as my Company Mission. With the help of function name, you can simple able to call it which makes simple and easy to type and understand.

It is found that the real syntax for creating a function is self-explanatory as initially you have to tell PHP about creating a function. For this, you need to type the keyword function which can be along function name.

---

## 2.2 In built functions

---

In PHP, there appears many functions which are built-in. Apart from these, there are many functions which requires particular PHP extensions, failure of which lead to fatal errors. It is noted that in case of imagecreatetruecolor() function, PHP will compiled with GD support or with MySQL support. The list below contains different function categories such as:

Array Functions

Calender Functions

Class/Object Functions

Character Functions

Date & Time Functions

Directory Functions

Error Handling Functions

File System Functions

MySQL Functions

Network Functions

ODBC Functions

## String Functions

### Simple XML Functions

### XML Parsing Functions

#### 2.2.1 String functions

There are many string functions that come with PHP which are not to be installed, such as:

|                             |                                                                                                       |
|-----------------------------|-------------------------------------------------------------------------------------------------------|
| <code>addslashes()</code>   | It returns string with backslashes in front of specified characters                                   |
| <code>addslashes()</code>   | It returns string with backslashes in front of predefined characters                                  |
| <code>bin2hex()</code>      | It converts string of ASCII characters to hexadecimal values                                          |
| <code>chop()</code>         | It removes whitespace characters from right end of string                                             |
| <code>chr()</code>          | It returns character from particular ASCII value                                                      |
| <code>chunk_split()</code>  | It splits string into series of tiny parts                                                            |
| <code>count_chars()</code>  | It return information about characters used in string                                                 |
| <code>crc32()</code>        | It calculates 32-bit CRC for string                                                                   |
| <code>crypt()</code>        | It is one-way string hashing                                                                          |
| <code>echo()</code>         | It gives one or more strings                                                                          |
| <code>explode()</code>      | It breaks string into an array                                                                        |
| <code>fprintf()</code>      | It writes formatted string to particular output stream                                                |
| <code>hebrew()</code>       | It converts Hebrew text to visual text                                                                |
| <code>hebrevc()</code>      | It converts Hebrew text to visual text and new lines ( <code>\n</code> ) into <code>&lt;br&gt;</code> |
| <code>hex2bin()</code>      | It converts string of hexadecimal values to ASCII characters                                          |
| <code>htmlentities()</code> | It converts characters to HTML entities                                                               |

|                               |                                                       |
|-------------------------------|-------------------------------------------------------|
| <code>implode()</code>        | It returns string from elements of an array           |
| <code>lcfirst()</code>        | It converts first character of string to lowercase    |
| <code>md5()</code>            | It will find MD5 hash of string                       |
| <code>md5_file()</code>       | It will find MD5 hash of file                         |
| <code>metaphone()</code>      | It will find metaphone key of a string                |
| <code>nl_langinfo()</code>    | It returns particular local information               |
| <code>ord()</code>            | It returns ASCII value of first character of a string |
| <code>parse_str()</code>      | It will parse query string inside variables           |
| <code>print()</code>          | It will give one or more strings                      |
| <code>printf()</code>         | It will give formatted string                         |
| <code>setlocale()</code>      | It will set locale information                        |
| <code>sha1_file()</code>      | It will show SHA-1 hash of a file                     |
| <code>similar_text()</code>   | It will give similarity between two strings           |
| <code>soundex()</code>        | It will find soundex key of a string                  |
| <code>sprintf()</code>        | It will write formatted string to variable            |
| <code>sscanf()</code>         | It will parse input from string as per format         |
| <code>str_repeat()</code>     | It will repeat string particular times                |
| <code>str_replace()</code>    | It changes characters in a string                     |
| <code>str_rot13()</code>      | It will do ROT13 encoding on string                   |
| <code>str_shuffle()</code>    | It shuffles all characters in string                  |
| <code>str_split()</code>      | It splits string into an array                        |
| <code>str_word_count()</code> | It will count number of words in string               |
| <code>strcasecmp()</code>     | It will compare two strings                           |
| <code>strip_tags()</code>     | It will strip HTML and PHP tags from string           |
| <code>strlen()</code>         | it will return length of string                       |

Array, function  
and expression

|                           |                                                             |
|---------------------------|-------------------------------------------------------------|
| <code>strpbrk()</code>    | It searches string for any set of characters                |
| <code>strrchr()</code>    | It locates last occurrence of string in another string      |
| <code>strrev()</code>     | It will reverse the string                                  |
| <code>strtok()</code>     | It splits the string in tiny strings                        |
| <code>strtolower()</code> | It changes string to lowercase letters                      |
| <code>strtoupper()</code> | It changes string to uppercase letters                      |
| <code>strtr()</code>      | It will translate characters in a string                    |
| <code>substr()</code>     | It will return part of string                               |
| <code>ucfirst()</code>    | It changes first character of string to uppercase           |
| <code>ucwords()</code>    | It will changes first character of every word to uppercase  |
| <code>vfprintf()</code>   | It will writes formatted string to particular output stream |
| <code>vprintf()</code>    | It gives out formatted string                               |
| <code>vsprintf()</code>   | It will write formatted string to variable                  |
| <code>wordwrap()</code>   | It will wraps string to given number of characters          |

### 2.2.3 Array functions

In PHP, array functions are part of the software where again there will be no need to install these functions.

|                             |                                                      |
|-----------------------------|------------------------------------------------------|
| <code>array()</code>        | This will creates an array                           |
| <code>array_chunk()</code>  | It will splits array into portions of arrays         |
| <code>array_diff()</code>   | It will compare arrays values                        |
| <code>array_fill()</code>   | It will fills array with values                      |
| <code>array_filter()</code> | It will filters values of array by callback function |
| <code>array_keys()</code>   | It will return all keys of an array                  |
| <code>array_merge()</code>  | It will merge one or more arrays in an array         |
| <code>array_pop()</code>    | It will erase last element of an array               |
| <code>array_push()</code>   | It will insert one or more elements to end of array  |
| <code>array_rand()</code>   | It will return one or more random keys from array    |

|                              |                                                                    |
|------------------------------|--------------------------------------------------------------------|
| <code>array_reduce()</code>  | It returns array as string with user-defined function              |
| <code>array_reverse()</code> | It returns an array in the reverse order                           |
| <code>array_search()</code>  | It will find array for given value and returns the key             |
| <code>array_slice()</code>   | It will return selected parts of an array                          |
| <code>array_splice()</code>  | It will remove and replaces particular elements of array           |
| <code>array_sum()</code>     | It returns sum of values in array                                  |
| <code>array_unique()</code>  | It erase duplicate values from array                               |
| <code>array_unshift()</code> | It will add one or more elements to start of an array              |
| <code>array_values()</code>  | It will return all values of array                                 |
| <code>array_walk()</code>    | It applies user function to all member of array                    |
| <code>compact()</code>       | It will form array having variables with their values              |
| <code>count()</code>         | It will return number of elements in array                         |
| <code>current()</code>       | It will return present element in array                            |
| <code>each()</code>          | It will return present key and value pair from array               |
| <code>end()</code>           | It will set internal pointer of array with last element            |
| <code>extract()</code>       | It takes variables in present symbol table from array              |
| <code>in_array()</code>      | It will check if particular value appears in an array              |
| <code>key()</code>           | It fetches key from an array                                       |
| <code>ksort()</code>         | It will sort associative array in increasing order, as per the key |
| <code>list()</code>          | It will assigns variables on expectation of an array               |
| <code>natsort()</code>       | It will sort array by natural order algorithm                      |
| <code>next()</code>          | It will advance internal array pointer of array                    |
| <code>prev()</code>          | It rewinds internal array pointer                                  |
| <code>range()</code>         | It forms array having range of elements                            |
| <code>reset()</code>         | It will set internal pointer of array to its first element         |
| <code>shuffle()</code>       | It will jumble the array                                           |
| <code>sort()</code>          | It will sort indexed array in increasing order                     |
| <code>uasort()</code>        | It will sort array by values using user-defined function           |
| <code>usort()</code>         | It sort array by user-defined comparison function                  |

### 2.2.4 Mathematical functions

In PHP there are built-in math functions which will take care of values that appear inside range of integer and float types. Following are the list of math functions in PHP.

|                             |                                                                              |
|-----------------------------|------------------------------------------------------------------------------|
| <code>abs()</code>          | It will return positive value of a number                                    |
| <code>acos()</code>         | It will return arc cosine of a number                                        |
| <code>acosh()</code>        | It will return inverse hyperbolic cosine of number                           |
| <code>asin()</code>         | It will return arc sine of a number                                          |
| <code>asinh()</code>        | It will return inverse hyperbolic sine of number                             |
| <code>atan()</code>         | It will return arc tangent of number in radians                              |
| <code>atan2()</code>        | It will return arc tangent of two variables x and y                          |
| <code>atanh()</code>        | It will return inverse hyperbolic tangent of number                          |
| <code>base_convert()</code> | It will convert number from one number base to another                       |
| <code>bindec()</code>       | It will convert binary to decimal number                                     |
| <code>ceil()</code>         | It will round number to nearest integer                                      |
| <code>cos()</code>          | It will return cosine of a number                                            |
| <code>cosh()</code>         | It will returns hyperbolic cosine of a number                                |
| <code>decbin()</code>       | It will change decimal to binary number                                      |
| <code>dechex()</code>       | It will change decimal to hexadecimal number                                 |
| <code>decoct()</code>       | It will change decimal to octal number                                       |
| <code>deg2rad()</code>      | It will change degree to radian value                                        |
| <code>exp()</code>          | It will find exponent of e                                                   |
| <code>expm1()</code>        | It will returns $\exp(x) - 1$                                                |
| <code>floor()</code>        | It will round number down to nearest integer                                 |
| <code>fmod()</code>         | It will return remainder of x/y                                              |
| <code>getrandmax()</code>   | It will return largest possible value returned by <u><code>rand()</code></u> |
| <code>hexdec()</code>       | It will change hexadecimal to decimal number                                 |

|               |                                                                  |
|---------------|------------------------------------------------------------------|
| hypot()       | It will change hypotenuse of right-angle triangle                |
| is_finite()   | It will check for finite value                                   |
| is_infinite() | It will check for infinite value                                 |
| is_nan()      | It will check for value of number                                |
| lcg_value()   | It will return pseudo random number in range from 0 to 1         |
| log()         | It will return natural logarithm of a number                     |
| log10()       | It will return base-10 logarithm of a number                     |
| log1p()       | It will return $\log(1+\text{number})$                           |
| mt_rand()     | It will form random integer by <u>Mersenne Twister</u> algorithm |
| octdec()      | It will change octal to decimal number                           |
| pi()          | It will return value of PI                                       |
| pow()         | It will return $x^y$                                             |
| rad2deg()     | It will change radian to degree value                            |
| rand()        | It will form random integer                                      |
| round()       | It will round floating-point number                              |
| sin()         | It will return sine of a number                                  |
| sinh()        | It will return hyperbolic sine of a number                       |
| sqrt()        | It will return square root of a number                           |

### 2.2.5 File System functions

In PHP, file system functions are used in programming which comes with built-in features. The list below shows file system functions such as

|                |                                         |
|----------------|-----------------------------------------|
| basename       | Returns trailing name component of path |
| chgrp          | Changes file group                      |
| chmod          | Changes file mode                       |
| chown          | Changes file owner                      |
| clearstatcache | Clears file status cache                |
| copy           | Copies file                             |

:



Array, function  
and expression

|                        |                                                      |
|------------------------|------------------------------------------------------|
| <code>delete</code>    | See <code>unlink</code> or <code>unset</code>        |
| <code>dirname</code>   | Returns a parent directory's path                    |
| <code>fclose</code>    | Closes an open file pointer                          |
| <code>feof</code>      | Tests for end-of-file on a file pointer              |
| <code>fflush</code>    | Flushes the output to a file                         |
| <code>fgetc</code>     | Gets character from file pointer                     |
| <code>fgetcsv</code>   | Gets line from file pointer and parse for CSV fields |
| <code>fgetc</code>     | Gets character from file pointer                     |
| <code>fgetcsv</code>   | Gets line from file pointer and parse for CSV fields |
| <code>fgets</code>     | Gets line from file pointer                          |
| <code>file</code>      | Reads entire file into an array                      |
| <code>fileatime</code> | Gets last access time of file                        |
| <code>fileinode</code> | Gets file inode                                      |
| <code>filesize</code>  | Gets file size                                       |
| <code>filetype</code>  | Gets file type                                       |
| <code>fopen</code>     | Opens file or URL                                    |
| <code>fread</code>     | Binary-safe file read                                |
| <code>fseek</code>     | Seeks on a file pointer                              |
| <code>glob</code>      | Find filenames matching a pattern                    |
| <code>is_dir</code>    | Tells whether the filename is a directory            |

|                            |                                               |                    |
|----------------------------|-----------------------------------------------|--------------------|
| <code>is_executable</code> | Tells whether the filename is executable      | Function in<br>PHP |
| <code>is_file</code>       | Tells whether the filename is a regular file  |                    |
| <code>is_link</code>       | Tells whether the filename is a symbolic link |                    |
| <code>is_readable</code>   | Tells whether a file exists and is readable   |                    |
| <code>link</code>          | Create a hard link                            |                    |
| <code>linkinfo</code>      | Gets information about a link                 |                    |
| <code>mkdir</code>         | Makes directory                               |                    |
| <code>pathinfo</code>      | Returns information about a file path         |                    |
| <code>pclose</code>        | Closes process file pointer                   |                    |
| <code>popen</code>         | Opens process file pointer                    |                    |
| <code>readfile</code>      | Outputs a file                                |                    |
| <code>readlink</code>      | Returns the target of a symbolic link         |                    |
| <code>rename</code>        | Renames a file or directory                   |                    |
| <code>rewind</code>        | Rewind the position of a file pointer         |                    |
| <code>rmdir</code>         | Removes directory                             |                    |
| <code>stat</code>          | Gives information about a file                |                    |
| <code>symlink</code>       | Creates a symbolic link                       |                    |
| <code>tempnam</code>       | Create file with unique file name             |                    |
| <code>tmpfile</code>       | Creates a temporary file                      |                    |
| <code>touch</code>         | Sets access and modification time of file     |                    |
| <code>unlink</code>        | Deletes a file                                |                    |

### 2.2.6 Date and Time Functions

In PHP, date and time functions are built-in that are used in PHP scripts. Such functions will be applied to format date and time that appears in many ways. The PHP `date()` function will format timestamp with syntax as `date(format,timestamp)`

Array, function  
and expression

|                              |                                                             |
|------------------------------|-------------------------------------------------------------|
| <code>checkdate()</code>     | Validates a Gregorian date                                  |
| <code>date_create()</code>   | Returns new DateTime object                                 |
| <code>date_date_set()</code> | Sets the date                                               |
| <code>date_format()</code>   | Returns date formatted according to given format            |
| <code>date_modify()</code>   | Alters the timestamp                                        |
| <code>date_sunrise()</code>  | It will return time of sunrise for given day / location     |
| <code>date_sunset()</code>   | It will return time of sunset for given day / location      |
| <code>date_time_set()</code> | It will set the time                                        |
| <code>date()</code>          | It will format local time/date                              |
| <code>gettimeofday()</code>  | It will returns array having current time information       |
| <code>gmdate()</code>        | It will format GMT/UTC date/time                            |
| <code>gmmktime()</code>      | It will return Unix timestamp for GMT date                  |
| <code>gmstrftime()</code>    | It will format GMT/UTC time/date as per locale settings     |
| <code>idate()</code>         | It will formats local time/date as integer                  |
| <code>microtime()</code>     | It will return microseconds for current time                |
| <code>mktime()</code>        | It will return Unix timestamp for a date                    |
| <code>strftime()</code>      | It will format local time/date according to locale settings |
| <code>strtotime()</code>     | It will parse time/date generated with strftime()           |
| <code>time()</code>          | It will return current time as Unix timestamp               |
| <code>timezone_open()</code> | It will returns new DateTimeZone object                     |

### 2.2.7 Miscellaneous Functions

In PHP there are many built-in miscellaneous functions that are:

|                    |                                                   |
|--------------------|---------------------------------------------------|
| connection_aborted | It will check for client disconnection            |
| connection_status  | It will return connection status bitfield         |
| constant           | It will return value of a constant                |
| define             | It will explain named constant                    |
| defined            | It will check for given named constant            |
| die                | It is same as exit                                |
| eval               | It will find a string as PHP code                 |
| exit               | It will show message and terminate current script |
| get_browser        | It direct about user's browser capacity           |
| highlight_file     | It shows syntax by highlighting a file            |
| pack               | It will pack data into binary string              |
| sleep              | It will delay execution                           |
| uniqid             | It will generate unique ID                        |
| unpack             | It will unpack data from binary string            |
| usleep             | It will delay execution in microseconds           |

Function in  
PHP

### 2.2.8 User Defined Functions

In PHP, there are more than 1,000 in-built functions which are called as per requirement. Such functions are known as User Defined Function. The syntax of this function is:

```
function function-name()
{
    statement 1 :
    statement 2 :
    statement 3 :
    .....
}
```

It is found that function declaration begins with function.

**Name of function:**

Array, function  
and expression

- It is defined by user.
- In this, valid function begins with letter or underscore with letters, numbers, or underscores.
- In this, function names are case-insensitive.

### **Opening and Closing curly braces ({ }):**

It is noted that function body is enclosed in pair of braces that has variable names and actual function code. Opening of curly bracket ({}) shows start of function code while closing of curly bracket (}) shows termination of function.

### **Example: PHP function**

view plaincopy to clipboardprint?

```
<?php  
function myfunction()  
{  
    echo "How are you";  
}  
myfunction();  
?>
```

Output:

When we call this function it will display:

How are you

### **Example: Functions within functions**

view plaincopy to clipboardprint?

Function in  
PHP

```
<?php
function function1()
{
    function function2()
    {
        echo "How are you <br>";
    }
}

function1();
function2();
?>
```

In this program we see that function1() is declared and function2() is declared inside function1(). So first we have to execute function1() and then function2() which will show "How are you".

By executing function1() makes function2() workable, hence we cannot call function2() directly without calling function1().

### 2.2.9 Arguments passing by reference

In PHP, we see that by default, function arguments are passed by value. By allowing function to change the arguments, the arguments must be passed by reference.

Also, you will be able to pass variable by reference to function hence function will be able to modify the variable. The syntax of argument passing by reference is:

```
<?php
function foo(&$var)
{
    $var++;
}

$a=8;
foo($a);
```

Array, function  
and expression

```
// $a is 9 here
```

```
?>
```

Now, you can also pass an argument to function as reference by adding ampersand (&) character before variable name as shown.

view plaincopy to clipboardprint?

```
<?php
function cube(&$x)
{
    $x = $x * $x * $x;
}
$result = 9;
cube($result);
echo $result;
?>
```

Output

729

### 2.2.10 Argument passing by value

In PHP, argument is any valid expression that is calculated and its value being assigned to required variable in a function. Consider a program, where function \$a is assigned to value 20 and function \$b is assigned to value 25:

```
function add($a, $b)
{
    ...
}
add(20, 25);
```

### Check your progress 1

1. What is the use of explode() function?
  - a. It breaks string into an array.
  - b. It gives one or more string
  - c. It spits string into parts.
  - d. None of these
2. flush () is a type of \_\_\_\_\_.
  - a. String function
  - b. Date function
  - c. File system function
  - d. None of these
3. What is the output of the following code?

```
<?php
function write() {
    echo "Programming is a fun!";
}
write();
?>
```

  - a. Write
  - b. echo "Programming is a fun!"
  - c. "Programming is a fun!"
  - d. Programming is a fun!

---

## 2.3 Let Us Sum Up

---

While studying this unit, we have learnt that functions in PHP resembles with the name given by us for bunch of code which to be executed whenever it is required by us. It is noted that by applying functions in code, you will save lot of time and effort and can easily be able to write a code which is much readable.

In PHP, there appears many functions which are built-in that requires particular PHP extensions, failure of which lead to fatal errors.



---

## 2.4 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –a), (2 –c), (3-d)

---

## 2.5 Glossary

---

1. **Structured data type** - It is a type of data type which is made of other datatypes.
2. **Declaration** - A syntactical element describing name and type of one or more variables, functions, structures, unions, or enumerations.
3. **Object** - Piece of data that which is manipulated by a program using simple variable, array, structure.

---

## 2.6 Assignment

---

Write short note on Argument passing by value.

---

## 2.7 Activities

---

Try to implement all string functions.

---

## 2.8 Case Study

---

Write a program by using functions and pass parameters by using call by reference and call by value.

---

## 2.9 Further Readings

---

1. Introduction to PHP, Chandresh Shah, 2010.
2. Features of Arrays and Function with Integrated Approach by B. Rahim, 2004.
3. Programming and Principles in PHP, Oxford, 2010, Jerry Smith.

---

## UNIT 3: Expression

---

### Unit Structure

- 3.0 Learning Objectives
- 3.1 Introduction
- 3.2 Regular Expression
- 3.3 Error Handling Regular Expressions
- 3.4 Exception handling
- 3.5 Let Us Sum Up
- 3.6 Answers for Check Your Progress
- 3.7 Glossary
- 3.8 Assignment
- 3.9 Activities
- 3.10 Case Study
- 3.11 Further Readings

---

### 3.0 Learning Objectives

---

After learning this Unit, you will be able to understand:

- About Regular Expression.
- About Error Handling Regular Expressions.
- About Exception handling.

---

### 3.1 Introduction

---

In PHP expressions plays an important building stones which describes what you write is an expression. It can be called as an expression is anything having a value. Important forms of expressions are constants and variables. If you type "\$x = 10", it means you have assigned '10' into \$x. '10' obviously has value 10 or it can be '10' an expression having value 10. If we see \$x's value to be 10, so \$y = \$a, also \$y = 10. We see that, \$x is an expression having value of 10.

---

## 3.2 Regular Expression

---

Regular expressions are strong tool which alter and changes text. These expressions having general pattern notation is like programming language that describes parse text which will search for patterns in string. Such types of expressions are slower as compared to normal expressions but are strong enough and are applied for particular use.

In PHP, these expressions are nothing but series or arrangement of characters which shows foundation for pattern matching functionality. With this, you can locate for particular string which available inside another string that can change one string with another and gets splited into pieces.

Regular expression has characters that will match among themselves as on searching "tra" in a string "Trains are fast," as you have a match since "tra" occurs in particular string. It is seen that many characters carry special meanings in regular expressions. It is noted that dollar sign (\$) is applied to match strings which ends with required pattern. Also, caret (^) character if applied at start of regular expression will show that it will match starting of string. It is noted that regular expression contains own set of functions as:

|                                |                                                                |
|--------------------------------|----------------------------------------------------------------|
| <code>preg_filter()</code>     | Performs a Regular Expression Search & Replace                 |
| <code>preg_grep()</code>       | Returns Array Entries That Match the Pattern                   |
| <code>preg_last_error()</code> | Returns the Error Code of the Last PCRE <u>Regex</u> Execution |
| <code>preg_match_all()</code>  | Performs a Global Regular Expression Match                     |
| <code>preg_match()</code>      | Performs a Regular Expression Match                            |
| <code>preg_quote()</code>      | Quote Regular Expression Characters                            |
| <code>preg_replace()</code>    | Performs a Regular Expression Search & Replace                 |
| <code>preg_split()</code>      | Splits a String by a Regular Expression                        |

PHP offers functions that are particular to two sets of regular expression functions having certain type of regular expression which can be:

- POSIX Regular Expressions
- PERL Style Regular Expressions

### POSIX Regular Expressions

The structure of POSIX regular expression is different of typical arithmetic expression where operators are combined to form complex expressions.

## Brackets

Expression

Brackets ([]) carry special meaning if applied in context of regular expressions and are applied to have a range of characters as shown:

[0-9] It matches any decimal digit from 0 through 9.

[a-z] It matches any character from lower-case a through lowercase z.

[A-Z] It matches any character from uppercase A through uppercase Z.

[a-Z] It matches any character from lowercase a through uppercase Z

## Quantifiers

The frequency or position of bracketed character sequences and single characters be shown by special character having particular connotation as +, \*, ?, {int. range}, and \$ flags:

p+ It matches any string having at least one p.

p\* It matches any string having zero or more p's.

p? It matches any string having zero or more p's.

p{N} It matches any string having sequence of N p's

p{2,3} It matches any string having sequence of two or three p's.

p{2, } It matches any string having sequence of at least two p's.

p\$ It matches any string with p at end of it.

^p It matches any string with p at beginning of it.

## PERL Style Regular Expressions

Perl style regular expressions are same as POSIX counterparts. In this, the syntax can be applied by changing with Perl style regular expression functions.

## Meta characters

Meta characters are alphabetical character which comes before backslash and gives mixture of special meaning. On locating for large money sums with 'd' meta character /(\\d+ )000/, \\d will locate for string of numerical character. The list below shows meta characters applied in PERL Style Regular Expressions.

. single character

\\s whitespace character (space, tab, newline)

Array, function  
and expression

`\S` non-whitespace character

`\d` digit (0-9)

`\D` non-digit

`\w` word character (a-z, A-Z, 0-9, \_)

`\W` non-word character

`[aeiou]` matches a single character in the given set

`[^aeiou]` matches a single character outside the given set

### Modifiers

There are modifiers present which makes work with regexps easily such as case sensitivity or locating in multiple lines.

`i` Makes the match case insensitive

`m` Specifies newline/carriage for string characters

`o` Evaluates expression only once

`s` Allows use of `.` to match a newline character

`x` Allows you to use white space in the expression for clarity

`g` Globally finds all matches

`cg` Allows a search to continue even after a global match fails

### Check your progress 1

1. What are meta characters?
  - a. alphabetical character which comes before backslash
  - b. alphabetical character which comes after backslash
  - c. Special character
  - d. None of these
2. What is POSIX?
  - a. Portable Operating System Interface for Linux
  - b. Portative Operating System Interface for Unix
  - c. Portable Operating System Interface for Unix
  - d. Portative Operating System Interface for Linux

### 3.3 Error Handling Regular Expressions

Error handling is the process of catching errors raised by your program and then taking appropriate action. If you would handle errors properly then it may lead to many unforeseen consequences. It's very simple in PHP to handle errors. You can write your own function to handling any error. PHP provides you a framework to define error handling function. This function must be able to handle a minimum of two parameters (error level and error message) but can accept up to five parameters. Its syntax is:

```
error_function(error_level,error_message, error_file,error_line,error_context);
```

**error\_level**      Shows error report level for user-defined error in terms of value.

**error\_message** Shows error message for user-defined error

**error\_file**      Shows file name of error

**error\_line**      Shows line number where error occurred

**error\_context** Shows array having variable and values in use during error

#### Using die() function

In a PHP program, check possible error condition before going forward and take correct action as required. Consider an example:

```
<?php
if(!file_exists("/tmp/test.txt")) {
    die("File not found");
} else {
    $file = fopen("/tmp/test.txt", "r");
    print "Open file successfully";
}

// Test of the code here.
?>
```

From the above, you will visualise that in such a way you can write good code and with this the program can be stopped with any errors.

### Check your progress 2

1. What is the use of error\_context parameter?
  - a. It Specifies the error message for the user-defined error
  - b. Specifies an array containing every variable, and their values, in use when the error occurred
  - c. Specifies the filename in which the error occurred
  - d. None of these

---

## 3.4 Exception handling

---

Exception handling is applied in order to change normal writing of code execution when exceptional conditions take place which is termed as exception. It is seen that when an exception is thrown, the code will not work and with this the PHP tries to locate matching "catch" block. When exception is trapped, in such case fatal error takes place that is taken by Uncaught Exception message. In this, error can be avoided by framing proper code which can take care of exception that will cover:

Try:

A function carrying exception be placed in try block. If exception fails to trigger, the code will continue as normal or on triggering, will be thrown out.

Throw:

This is how you trigger an exception. Each throw must have at least one catch.

Catch:

A catch block retrieves an exception and creates an object containing the exception information.

Once the exception is thrown, code that follows the statement will not be executed, and PHP will attempt to find the first matching catch block. If an exception is not caught, a fatal error will be given with catching the exception. An exception can be thrown, and caught in PHP where code is surrounded in try block. Each try must have at least one corresponding catch block. Multiple catch blocks can be used to catch different classes of exceptions. Exceptions can be thrown (or re-thrown) within a catch block.

## Example

Expression

```
<?php
try {
    $error = 'Always throw this error';
    throw new Exception($error);

    // Code following an exception is not executed.
    echo 'Never executed';
} catch (Exception $e) {
    echo 'Caught exception: ', $e->getMessage(), "\n";
}

// Continue execution
echo 'Hello World';
?>
```

From the above example, we see that `$e->getMessage` function is applied to have an error message. Study the functions applied from `Exception` class.

- `getMessage()` – message of exception
- `getCode()` – code of exception
- `getFile()` – source filename
- `getLine()` – source line
- `getTrace()` – n array of the `backtrace()`
- `getTraceAsString()` – formatted string of trace

## Creating Custom Exception Handler

You can define your own custom exception handler by applying user-defined exception handler function as:

```
string set_exception_handler ( callback $exception_handler )
```



Array, function  
and expression

Here `exception_handler` is the name of the function to be called when an uncaught exception occurs. This function must be defined before calling `set_exception_handler()`.

### Example

```
<?php

function exception_handler($exception) {
    echo "Uncaught exception: ", $exception->getMessage(), "\n";
}

set_exception_handler('exception_handler');
throw new Exception('Uncaught Exception');
echo "Not Executed\n";
?>
```

### Check your progress 3

1. Which version of PHP was incorporated with Exception handling?
  - a. PHP 4
  - b. PHP 6
  - c. PHP 5
  - d. None of these
2. Which of the following statement is used to call exception class?
  - a. `throw new Exception();`
  - b. `throws new Exception();`
  - c. `new Exception();`
  - d. None of these

---

## 3.5 Let Us Sum Up

---

MS-DOS and OS/2 use another variation on linked list called FAT. Index allocation addresses many of the problems of contiguous and chained allocation.

C-Scan Scheduling is a type of scheduling, where the processes get arranged by using particular circular order list.

Expression

Round Robin is a type of scheduling where the time of CPU is shared into equal numbers which is called as Quantum Time.

---

### 3.6 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 -a), (2 -c)

**Check your progress 2**

**Answers:** (1 -b)

**Check your progress 3**

**Answers:** (1 -c), (2 -a)

---

### 3.7 Glossary

---

1. **Declaration** - A syntactical element describing name and type of one or more variables, functions, structures, unions, or enumerations.
2. **Object** - Piece of data that which is manipulated by a program using simple variable, array, structure.
3. **Regular expressions** - Strong tool required to change and ammend text.
4. **Quantifiers** - They are frequency bracket character sequences or single characters highlighted by special character.
5. **Error handling** - Getting errors raised by program and applying required action.

---

### 3.8 Assignment

---

Explain Error handling with the help of an example?

---

## 3.9 Activities

---

Study regular exceptions in detail and try to implement.

---

## 3.10 Case Study

---

Create a custom exception handler.

---

## 3.11 Further Readings

---

1. Features of Arrays and Function with Integrated Approach by B. Rahim, 2004.
2. Programming and Principles in PHP, Oxford, 2010, Jerry Smith.

---

## Block Summary

---

In this block, you have learnt and understand about the `sizeof()` function and `array_unique()` functions in PHP. The block gives an idea on array having numeric index and array having name index. You have been well explained with the concepts of application of syntax with respect to Perl style regular expression functions.

The block detailed about the Study about Error Handling Regular Expressions and features of Exception handling. The concept related to Uniqueness sorting array and determining array size will be well explained to students. You will be demonstrated practically about Regular Expression.

---

## Block Assignment

---

### Short Answer Questions

1. What is an associative array? Give example using code snippet
2. Explain the String functions?
3. Write a short note on exception handling?
4. Explain user defined functions?

### Long Answer Questions

1. Write any program and try to solve error using error handling.
2. Write a program that throws multiple exceptions.
3. Write a program for multidimensional array.

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# MOBILE APPLICATION DEVELOPMENT

PGDCA 203

**BLOCK 2:  
ANDROID APPLICATION AND  
USER INTERFACE DESIGN**



**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# MOBILE APPLICATION DEVELOPMENT

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# Mobile Application Development

## Contents

### **BLOCK 1:      BASICS OF ANDROID APPLICATION**

#### **UNIT 1            INTRODUCTION TO ANDROID, TOOLS AND BASICS**

The Android Platform, Installing Android Studio, Java for Android, Android Studio for Android Software Development, Building a sample Android application

#### **UNIT 2            ANDROID APPLICATION DESIGN ESSENTIALS - I**

A Framework for a Well-Behaved Application, Application Context, Activities, Services, Intents, Intent Filter, Permissions, Receiving and Broadcasting Intents,

---

### **BLOCK 2:      ANDROID APPLICATION AND USER INTERFACE DESIGN**

#### **UNIT 1            ANDROID APPLICATION DESIGN ESSENTIALS - II**

Using Intent Filter, Permissions, Android Manifest File and its common settings, managing different types application resources in a hierarchy

#### **UNIT 2            Android User Interface Design and Common APIs**

User Interface Screen elements, Designing User Interfaces with Layouts, Drawing and Working with Animation ,Drawing 2D and 3D Graphics and Multimedia,

---



**BLOCK 3:      ANDROID NETWORKING AND DEVELOPMENT**

**UNIT 1      ADVANCED TOPICS - I**

Android Networking, Web and Telephony API, Search, Location and Mapping, Sensors, NFC, Speech, Gestures, and Accessibility,

**UNIT 2      ADVANCED TOPICS - II**

Communication, Identity, Sync, and Social Media, The Android Native Development Kit (NDK)

**BLOCK 4:      ANDROID APPLICATION PUBLISHING AND CONTENT PROVIDERS**

**UNIT 1      MORE ON ANDROID**

Handling and Persisting Data, A Content Provider as a Facade for a RESTful Web Service, Using Content Providers

**UNIT 2      PUBLISHING ANDROID APPLICATION**

Deploying Android Application to the World, Selling your Android application



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA-203

## **MOBILE APPLICATION DEVELOPMENT**

---

### **BLOCK 1: ANDROID APPLICATION AND USER INTERFACE DESIGN**

---

#### **UNIT 1**

ANDROID APPLICATION DESIGN ESSENTIALS - II

03

#### **UNIT 2**

ANDROID USER INTERFACE DESIGN AND COMMON APIS

15



---

# **BLOCK 2: ANDROID APPLICATION AND USER INTERFACE DESIGN**

---

## **Block Introduction**

Intents are asynchronous messages which permit application components to request functionality from other android components. Intents allow you to interact with components from the same applications additionally like components contributed by other applications. The new Fragment API for android, introduced in android 3.0, permits for easier dynamic user interfaces. API that is application program interface may be a set of routines, protocols, and tools for building software applications.

In this block, we will detail about the basic of Intent messages and usability of application components. The block will focus on the study and concept of running android 6.0 with API level 23 with permission requiring for access of additional features. You will get an idea on application in Android with necessity of Manifest.xml file.

In this block, you will make to learn and understand about new Fragment API for android for easy dynamic user interfaces. The concept related to interface elements in android app using view and ViewGroup objects will also be explained to the students. The student will be demonstrated practically about animations.

## **Block Objective**

**After learning this block, you will be able to understand:**

- The basic of applying Intent Filter
- Study features of Permissions
- Concept of Android Manifest File
- Idea about managing different types application resources
- Features of Android User Interface Design
- Idea about User Interface Screen elements
- Concept of Layouts
- Working characteristics of Graphics and Multimedia

## **Block Structure**

**Unit 1:     Android Application Design Essentials - II**

**Unit 2:     Android Application Design Essentials - I**

---

# **UNIT 1: ANDROID APPLICATION DESIGN ESSENTIALS - II**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Using Intent Filter**
- 1.3 Permissions**
- 1.4 Android Manifest File and its common settings**
- 1.5 Managing different types application resources in a hierarchy**
- 1.6 Let Us Sum Up**
- 1.7 Answers for Check Your Progress**
- 1.8 Glossary**
- 1.9 Assignment**
- 1.10 Activities**
- 1.11 Case Study**
- 1.12 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About Intent Filter
- About Android Manifest File
- About application resources

---

## **1.1 Introduction**

---

Your app's user interface is collectively that the user can see and collaborate with. Android provides a variety of pre-built UI components such as structured layout objects and UI controls that allow you to build the graphical user interface

for your app. Android also provides other UI modules for different interfaces such as dialogs, notifications, and menus.

---

## 1.2 Using Intent Filter

---

Intents are asynchronous messages which permit application components to request functionality from other android components. Intents allow you to interact with components from the same applications additionally like components contributed by other applications. Intents are used to signal to the android system that a precise event has occurred. Intents typically describe the action that should be performed and provide data upon that such an action should be done. for example, your application will begin a browser component for a certain url via an intent. This is demonstrated by the following example.

```
String url = "http://www.vogella.com";  
Intent i = new Intent(Intent.ACTION_VIEW);  
i.setData(Uri.parse(url));  
startActivity(i);
```

A component will register itself via an intent filter for a selected action and specific data. an intent filter specifies the kinds of intents to that an activity, service, or broadcast receiver will respond to by declaring the capabilities of a component.

Android components register intent filters either statically within the androidManifest.xml or just in case of a broadcast receiver also dynamically via code. An intent filter is defined by its category, action and data filters. It also can contain additional meta-data.

If intent is sent to the android system, the android platform runs a receiver determination. It uses the data included within the intent. If many components have registered for the same intent filter, the user will decide that component should be started.

### Intent filter

You can register your android components via intent filters for certain events. If a component doesn't define one, it will solely be referred to as by explicit intents. This chapter offers an example for registering a component for intent. The key for this registration is that your component registers for the correct action, mime-type and specifies the correct meta-data.

To inform the system those implicit intents they will handle, activities, services, and broadcast receivers will have one or more intent filters. Each filter describes a capability of the component, a set of intents that the component is willing to receive. It, in effect, filters in intents of a desired kind, whereas filtering out unwanted intents — however solely unwanted implicit intents (those that do not name a target class). A certain intent is usually delivered to its target, no matter what it contains; the filter isn't consulted. However an implicit intent is delivered to a component as long as it will undergo one of the component's filters.

An intent filter is an instance of the `IntentFilter` class. However, since the android system should realize the capabilities of a part before it will launch that part, intent filters are typically not set up in Java code, however within the application's manifest file (`AndroidManifest.xml`) as `<intent-filter>` elements. (The one exception would be filters for broadcast receivers that square measure registered dynamically by calling `Context.registerReceiver()`; they're directly created as `IntentFilter` objects.)

A filter has fields that parallel the action, data, and category fields of an Intent object. An implicit intent is tested against the filter in all 3 areas. To be delivered to the component that owns the filter, it must pass all 3 tests. If it fails even one in all them, the android system will not deliver it to the part — at least not on the basis of that filter. However, since a component can have multiple intent filters, an intent that doesn't pass through one of a component's filters may make it through on another.

### Check your progress 1

1. What is android intent?
  - a. It is an abstract description of an operation to be performed
  - b. activity
  - c. process
  - d. None of these
2. The Primary elements of intent are \_\_\_\_\_.
  - a. Action
  - b. Data
  - c. Both of these
  - d. None of these

---

## 1.3 Permissions

---

It's easy for an app to overwhelm a user with permission requests. If a user finds the app frustrating to use, or the user is worried regarding what the app could be doing with the user's information, they will avoid using the app or uninstall it entirely. Whenever you ask for permission, you force the user to form a decision. You must minimize the amount of times you make these requests. If the user is running android 6.0 (API level 23) or later, whenever the user tries some new app feature that needs a permission, the app needs to interrupt the user's work with a permission request. If the user is running an earlier version of golem, the user needs to grant all of the app's permissions when installing the app; if the list is just too long or appears inappropriate, the user could decide not to install your app at all. For these reasons, you must minimize the amount of permissions your app wants.

If the user is running android 6.0 (API level 23) or later, the user needs to grant your app its permissions whereas they're running the app. If you confront the user with a lot of requests for permissions promptly, you'll overwhelm the user and cause them to quit your app. Instead, you should invite permissions as you need them.

In some cases, one or additional permissions could be absolutely essential to your app. it would add up to ask for all of these permissions as soon because the app launches. For example, if you make a photography app, the app would wish access to the device camera. Once the user launches the app for the first time, they will not be surprised to be asked for permission to use the camera. However if the same app also had a feature to share photos with the user's contacts, you probably mustn't invite the `READ_CONTACTS` permission at first launch. Instead, wait till the user tries to use the "sharing" feature and ask for the permission.

The permissions dialog shown by the system when you call `requestPermissions()` says that what permission your app wants, but doesn't tell why. In some cases, the user may find it confusing. It's has been a good idea to explain to the user why your app wants the permissions before calling `requestPermissions()`.

For example, a photography app might want to use location services so it can geotag the photos. A typical user might not understand that a photo can contain location information, and would be puzzled why their photography app wants to know the location. So in this case, it's a good idea for the app to tell the user about this feature before calling `requestPermissions()`.

### Check your progress 2

1. Which method is called when the app seeks permission from the user?
  - a. startActivity()
  - b. requestPermission()
  - c. broadcastReceiver()
  - d. None of these

---

## 1.4 Android Manifest File and its common settings

---

Every application should have an AndroidManifest.xml file (with precisely that name) in its root directory. The manifest file presents essential info regarding your app to the android system, info the system must have before it will run any of the app's code. Among other things, the manifests will the following:

- It names the Java package for the application. The package name is a unique identifier for the application.
- It describes the components of the application — the activities, services, broadcast receivers, and content providers that the application consists of. It names the classes that implement every of the elements and publishes their capabilities (for example, that Intent messages they will handle). These declarations let the android system know what the components are and under what conditions they will be launched.
- It determines that processes can host application components.
- It declares that permissions the application should have so as to access protected parts of the API and interact with other applications.
- It additionally declares the permissions that others are needed to have so as to move with the application's components.
- It lists the Instrumentation classes that provide profiling and other info because the application is running. These declarations are present within the manifest solely whereas the application is being developed and tested; they are removed before the application is published.
- It declares the minimum level of the android API that the application needs.
- It lists the libraries that the application should be linked against.

AndroidManifest.xml could be a powerful go in the android platform that enables you to describe the functionality and needs of your application to android.

However, working with it's not easy. Xamarin.Android helps to reduce this problem by permitting you to feature custom attributes to your classes, which can then be wont to automatically generate the manifest for you. Our goal is that 99 of our users should never need to manually modify AndroidManifest.xml.

AndroidManifest.xml is generated as a part of the build method, and also the XML found inside Properties/AndroidManifest.xml is merged with XML that's generated from custom attributes. The resulting merged AndroidManifest.xml resides within the obj subdirectory; for example, it resides at obj/Debug/android/AndroidManifest.xml for debug builds. The merging process is trivial: it uses custom attributes within the code to generate XML elements, and inserts those elements into AndroidManifest.xml.

### Structure of Manifest File

The code below shows the structure of manifest file and every element that it can contain. Each element, along with all of its attributes, is documented in full in a separate file.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest>
    <uses-permission />
    <permission />
    <permission-tree />
    <permission-group />
    <instrumentation />
    <uses-sdk />
    <uses-configuration />
    <uses-feature />
    <supports-screens />
    <compatible-screens />
    <supports-gl|texture />
    <application>
```



```
<activity>
  <intent-filter>
    <action />
    <category />
    <data />

  </intent-filter>
  <meta-data />
</activity>
<activity-alias>
  <intent-filter> . . . </intent-filter>
  <meta-data />
</activity-alias>
<service>
  <intent-filter> . . . </intent-filter>
  <meta-data/>
</service>

<receiver>
  <intent-filter> . . . </intent-filter>
  <meta-data />
</receiver>
<provider>
  <grant-uri-permission />
  <meta-data />
  <path-permission />
</provider>
<uses-library />
</application>
</manifest>
```

All the elements that appear in manifest file are highlighted below. These are the standard elements that are used in manifest files.

|                   |                    |                        |               |
|-------------------|--------------------|------------------------|---------------|
| <action>          | <activity>         | <activity-alias>       | <application> |
| <category>        | <data>             | <grant-uri-permission> |               |
| <instrumentation> | <intent-filter>    | <manifest>             | <meta-data>   |
| <permission>      | <permission-group> | <permission-tree>      | <provider>    |
| <receiver>        | <service>          | <supports-screens>     | <uses-        |
| configuration>    | <uses-feature>     | <uses-library>         |               |
| <uses-permission> | <uses-sdk>         |                        |               |

### Check your progress 3

1. What is android manifest file?
  - a. java file
  - b. drivers
  - c. It is a file that application code, resources, permissions, icons and themes
  - d. None of these
2. What is the use of manifest file?
  - a. It names the Java package for the application
  - b. It determines which processes will host application components
  - c. It lists the libraries of the application
  - d. All of these

---

## 1.5 Managing different types application resources in a hierarchy

---

The well-written application accesses its resources programmatically rather than hard coding them into the source code. this is often done for a variety of reasons. Storing application resources in a very single place is a additional organized approach to development and makes the code more legible and maintainable. Externalizing resources like strings makes it easier to localize applications for various languages and geographic regions. All android applications are composed of 2 things:

- Functionality or code instructions

- Data or resources

The functionality is that the code that determines how your application behaves. This includes any algorithms that make the application run. Resources include text strings, images and icons, audio files, videos, and other data employed by the application.

Android resource files are stored separately from the java class files within the android project. Most common resource types are stored in XML. You'll additionally store raw data files and graphics as resources. Resources are organized in a very strict directory hierarchy within the android project. All resources should be stored under the /res project directory in specially named subdirectories that has to be lowercase.

Application resources are created and stored within the android project files under the /res directory. Employing a well-defined however flexible directory structure, resources are organized, defined, and compiled with the appliance package. Application resources are not shared with the rest of the android system.

### **Storing Application Resources**

Defining application data as resources (as opposed to at runtime in code) is good programming practice. Grouping application resources together and compiling them into the application package has the following benefits:

- Code is cleaner and easier to read, resulting in fewer bugs.
- Resources are organized by type and guaranteed to be unique.
- Resources are conveniently set for handset customization.
- Localization and internationalization are simple.

The android platform supports a range of resource sorts, which might be combined to create differing kinds of applications. Android applications will include many different varieties of resources. The following are a number of the most common resource types:

- Strings, colors, and dimensions
- Drawable graphics files
- Layout files
- Raw files of all types

Resource sorts are defined with special XML tags and organized into specially named project directories. Some /res subdirectories, like the /drawable,

/layout, and /values directories, are created by default once a new android project is formed, however others should be added by the developer once required.

Resource files stored within /res subdirectories must abide by the following rules:

- Resource filenames must be in lowercase.
- Resource filenames may contain letters, numbers, underscores, and periods only.
- Resource filenames must be unique.

When resources are compiled, their name dictates their variable name. For example, a graphics file saved within the /drawable directory as mypic.jpg is referenced as @drawable/mypic. It's necessary to name resource names intelligently and be aware of character limitations that are stricter than file system names. Consult the android documentation for specific project directory naming conventions.

Simple resources like string, color, and dimension values ought to be defined in XML files under the /res/values project directory in XML files. These resource files use special XML tags that represent name/value pairs. These types of resources are compiled into the application package at build time.

All application resources are kept within the /res project directory structure and are compiled into the project at build time. Application resources may be used programmatically. They'll even be referenced in different application resources. Application resources may be accessed programmatically using the generated class file called R.java. To reference a resource from within your Activity class, you must retrieve the application's Resources object using the getResources() method then build the appropriate method call, supported the kind of resource you want to retrieve.

#### **Check your progress 4**

1. In android resources are stored inside\_\_\_\_\_.
  - a. Java file
  - b. Layout folder
  - c. Res Folder of XML file
  - d. None of these

---

## 1.6 Let Us Sum Up

---

In this unit we have learnt that Intents are asynchronous messages which permit application components to request functionality from other android components that allow interacting with components from similar applications with components that are contributed by other applications.

If user is running android 6.0 or later, whenever the user tries some new app feature that needs permission, the app needs to interrupt the user's work with a permission request.

It is noted that all application should have AndroidManifest.xml file in its root directory that shows required info regarding an app for android system.

It is found that a well-written application accesses its resources programmatically rather than hard coding them into the source code which often is carried out for variety of reasons.

---

## 1.7 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –a), (2-c)

**Check your progress 2**

**Answers:** (1 -b)

**Check your progress 3**

**Answers:** (1 –c), (2 –d)

**Check your progress 4**

**Answers:** (1 -c)

---

## 1.8 Glossary

---

1. **Activity** - An application screen that supports Java code from Activity class.

2. **Application** - In Android application, there are activities, services, listeners and intent receivers.
3. **Intent** - Message object which uses to communicate with certain applications/activities asynchronously.

---

## 1.9 Assignment

---

Explain intent and intent filters with the help of an example.

---

## 1.10 Activities

---

Study android manifest file of your project.

---

## 1.11 Case Study

---

Study android application resources.

---

## 1.12 Further Readings

---

1. Learn Java for Android Development (2nd edition), Jeff Friesen, 2013
2. Android Application Development for Java Programmers, James C. Sheusi, 2012

---

## **UNIT 2: ANDROID USER INTERFACE DESIGN AND COMMON APIS**

---

### **Unit Structure**

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 User Interface Screen elements**
- 2.3 Designing User Interfaces with Layouts**
- 2.4 Drawing and Working with Animation**
- 2.5 Drawing 2D and 3D Graphics and Multimedia**
- 2.6 Let Us Sum Up**
- 2.7 Answers for Check Your Progress**
- 2.8 Glossary**
- 2.9 Assignment**
- 2.10 Activities**
- 2.11 Case Study**
- 2.12 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About User Interface Screen
- About Working with Animation
- About Drawing 2D and 3D Graphics

---

### **2.1 Introduction**

---

The new Fragment API for android, introduced in android 3.0, permits for easier dynamic user interfaces. API that is application program interface may be a set of routines, protocols, and tools for building software applications. The API specifies however software components should interact and apis are used once

programming graphical program (GUI) components. A good API makes it easier to develop a program by providing all the building blocks. A programmer then puts the blocks together.

---

## 2.2 User Interface Screen elements

---

All user interface elements in an android app are built using view and ViewGroup objects. A view is an object that draws one thing on the screen that the user will move with. A ViewGroup is an object that holds alternative View (and ViewGroup) objects so as to define the layout of the interface.

The user interface for every component of your app is defined using a hierarchy of view and ViewGroup objects, as shown in figure 2.1. every view group is an invisible container that organizes child views, whereas the child views is also input controls or other widgets that draw some part of the UI. This hierarchy tree is often as easy or complicated as you wish it to be.

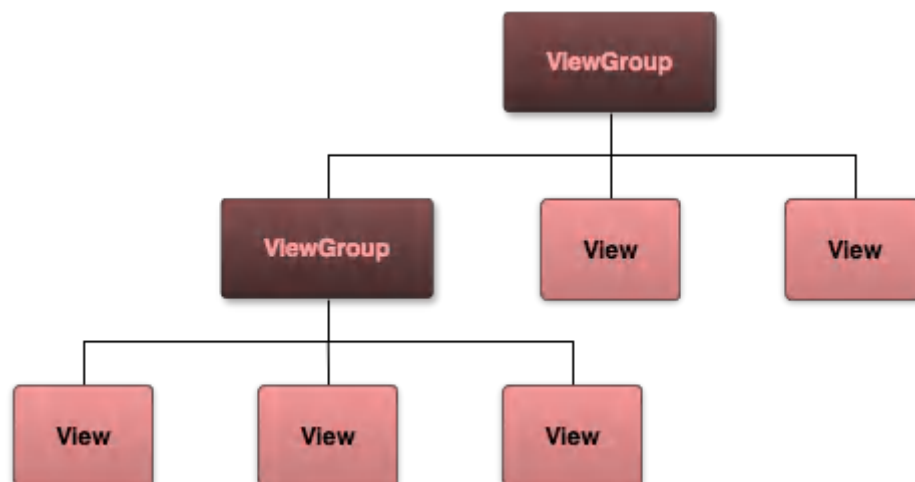


**Fig 2.1 User Interface components**

To declare your layout, you can instantiate View objects in code and start building a tree, but the easiest and the most effective way to define your layout is by using an XML file. XML offers a human-readable structure for the layout, similar to HTML. The name of an XML element for a view is respective to the Android class it represents. So a `<TextView>` element creates a TextView widget in your UI, and a `<LinearLayout>` element creates a LinearLayout view group.



### Check your progress 1

1. What is view group?
  - a. class
  - b. It is a collection of views
  - c. object
  - d. None of these
2. What are the attributes of <TextView>?
  - a. android:id
  - b. android:editable
  - c. android:fontFamily
  - d. All of these

---

## 2.3 Designing User Interfaces with Layouts

---

A layout defines the visual structure for a program, like the UI for an activity or app convenience. You'll be able to declare a layout in 2 ways:

- Declare UI elements in XML. android provides a simple XML vocabulary that corresponds to the real classes and subclasses, like those for widgets and layouts.
- Instantiate layout elements at runtime. Your application will create read and ViewGroup objects (and manipulate their properties) programmatically.

The android framework provides you the flexibility to use either or each of those methods for declaring and managing your application's UI. For example, you'll declare your application's default layouts in XML, including the screen elements which will seem in them and their properties. You'll then add code in your application that would modify the state of the screen objects, including those declared in XML, at run time.

The advantage to declaring your UI in XML is that it allows you to better separate the presentation of your application from the code that controls its behavior. Your UI descriptions are external to your application code, which implies that you simply will modify or adapt it while not having to modify your source code and recompile. For example, you'll produce XML layouts for

different screen orientations, totally different device screen sizes, and totally different languages. Additionally, declaring the layout in XML makes it easier to visualize the structure of your UI, therefore it's easier to debug problems. As such, this document focuses on teaching you how to declare your layout in XML. If you are interested in instantiating view objects at runtime, refer to the ViewGroup and view class references.

XML layout attributes named `layout_something` define layout parameters for the view that are acceptable for the ViewGroup in which it resides.

Every ViewGroup class implements a nested class that extends ViewGroup.LayoutParams. This subclass contains property varieties that define the size and position for every child view, as appropriate for the view group. As you'll see in figure 2.2, the parent view group defines layout parameters for every child view.

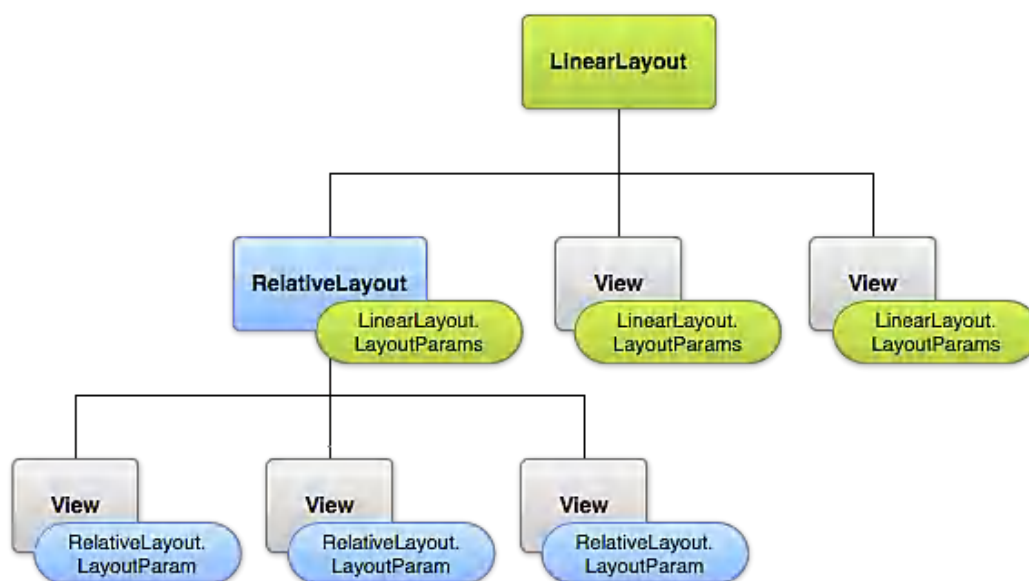


**Fig 2.2 Designing Layout**

The geometry of a view is that of a rectangle. A view includes a location, expressed as a pair of left and top coordinates, and 2 dimensions, expressed as a width and a height. The unit for location and dimensions is the pixel.

It is possible to retrieve the location of a view by invoking the methods `getLeft()` and `getTop()`. The previous returns the left, or X, coordinate of the rectangle representing the view. The latter returns the top, or Y, coordinate of the rectangle representing the view. These methods both return the location of the view relative to its parent. For instance, when `getLeft()` returns 20, which means the view is located 20 pixels to the right of the left edge of its direct parent.

In addition, many convenience strategies are offered to avoid unnecessary computations, specifically `getRight()` and `getBottom()`. These strategies come the coordinates of the correct and bottom edges of the rectangle representing the view. for instance, calling `getRight()` is similar to the following computation: `getLeft() + getWidth()`.

Each subclass of the `ViewGroup` class provides a unique way to show the views you nest within it. Below are a number of the a lot of common layout types that are built into the android platform.

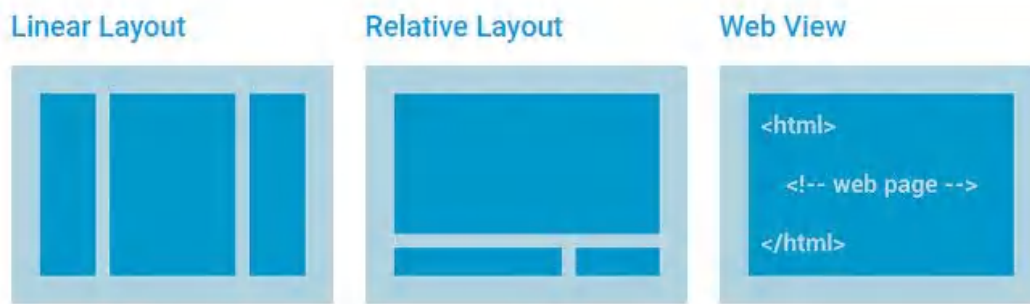


**Fig 2.3 Different Layouts**

### Check your progress 2

1. Where does a layout placed in android?
  - a. In XML file
  - b. In Resource folder
  - c. Not defined clearly
  - d. None of these
2. What is linear layout?
  - a. Placing views in rows and columns
  - b. Placing views either horizontally or vertically
  - c. Placing views in relative position
  - d. None of these

---

## 2.4 Drawing and Working with Animation

---

It is found that in Android, the framework provides dual animation systems:

- property animation
- view animation

The above two animation systems are practical options, whereas the property animation system is preferred method to apply as it is flexible and offers more features. Additionally, these two systems will allow drawable animation that can be easily available to load drawable resources and will show in single frame one after another.

Notwithstanding working on what, normally such devices carried restricted power, which serves as highest rated mobile applications that carries sophisticated User Experience (UX) that carries full high quality graphics as well as animations which shows an perceptive, thus far approachable and energetic outlook. With increase in mobile applications and leading to more and more sophisticatedness, consumers will expect even high from applications.

### Animation

Animation is the base of Android application as users like things which tend to move here and there on the screen. Animations appear as advance way which helps in improving user expertise for particular application that will help in standing out. The best animations are the ones that users don't happen to see as it looks natural. Android provides the following three API's for:

View Animation:

It serves as genuine API which Android gives. Such type of animations is fixed to particular View and will able to do simple transformations of contents regarding particular View. Being simple and ease, such API tends to be of use for things which can be an alpha animation, rotations and further.

Property Animation:

This type of animation is introduced in Android which allows an application to animate with all its features. It is of used when an change in property of an object appears, as if particular object is not visible on screen.

Drawable Animation:

It is a special Drawable resource which on application results in simple animation effect along with its layouts.

### Check your progress 3

1. To change the property of an object we use\_\_\_\_\_.
- a. View Animaton
- b. Drawable Animation
- c. Property Animation
- d. All of these

---

## 2.5 Drawing 2D and 3D Graphics and Multimedia

---

Android is rich and carries varied framework which supports 2D and 3D graphics along with animations. In this, the graphics splits into several approaches which are utilized for 2D and 3D graphic animations. In Android, 3D graphics are present through various built in frameworks which can be OpenGL ES for OpenGL and for 3<sup>rd</sup> party frameworks as MonoGame.

It is seen that Android will have two different API's that frames 2D graphics which can be high level declarative approach while other is of programmatic low-level API:

- **Drawable Resources** – Such resources applies to develop custom graphics which can be programmatical or typical that embeds drawing instructions in XML files. It is noted that Drawable resources normally are explained as XML files having instructions or actions for Android which can be utilized for 2D graphic.
- **Canvas** – It is a sort of low level API which involves drawing directly on a bitmap and shows very fine-grained control on display.

Along with 2D graphics techniques, Android carries numerous ways to develop animations which can be:

- **Drawable Animations:** It is found that Android supports frame-by-frame animations which are called as Drawable Animation. It is easy and simple animation API where Android sequentially loads and show Drawable resources one after another in series just like cartoon.
- **View Animations:** It seems that View Animations appears as original animation API's which is present in Android along with its versions. Such type of API are restricted which allows work with View objects and will do

simple transformations on particular Views. There are normally explained in XML files which are available in /Resources/anim folder.

- **Property Animations:** It is noted that in Android, such animations exists as new set of animation API's. which introduces extensible and flexible system which can be utilized to animate properties of particular object and not simply the View objects. It is flexible which allows animations to put in a nutshell in different classes which allows sharing of code in easy way.

## **2D Graphics**

Drawable Resources are customary as well as elevated mechanism in Android applications. As with external expedients, Drawable Resources are affirmative – they're described in XML files. This access authorizes for a filter estrangement of code from reserves. This can advance execution as well as extension since it is not essential to alter code to update or convert the graphics in an Android application.

Furthermore, while Drawable Resources are advantageous for ample elementary as well as common graphic conditions, however the default the power as well as control of the Canvas API.

The extraneous approach, applying the Canvas material, is very analogous to estranged accustomed API arrangements comparable as System. Drawing or iOS's Core Drawing. Applying the Canvas material assigns the best control of how 2D graphics are developed. It is applicable for circumstances where a Drawable Resource will not perform or will be hard to work with. For exemplary, it may be necessary to draw a custom slider control whose appearance will change based on calculations related to the value of the slider.

### **Canvas Drawing API**

Drawables are authoritative furthermore acquire their boundaries. Assured substances are both not feasible and very complicated. For exemplary, conducting a filter to a blueprint that was acquired by a camera on the appliance. It would be very burdensome to conduct red-eye compaction by utilizing a Drawable Resource. Instead, the Canvas API authorizes an application very establish grained control to selectively adjust colors in a definite fraction of the picture.

One class that is averagely utilized with the Canvas endures the Paint class. This class accepts colour as well as style information about how to illustrate on the class. It is utilized to assign things comparable a color as well as transparency.

The Canvas API utilizes the painter's archetypal to illustrate 2D graphics. Actions are conducted in successive levels on foremost of each other. Each behavior will overlay numerous domain of the basic bitmap. When the domain interlaces a subsequently painted domain the fresh paint will fractionally or entirely darken the old. This is the equivalent approach abundant other drawing APIs comparable as System.Drawing along with iOS's Core Graphics function.

### Check your progress 4

1. To create 2 – D graphics in android we can use \_\_\_\_\_.
  - a. Canvas
  - b. Drawable resources
  - c. Both of these
  - d. None of these
2. Why canvas is better than drawable resource for creating graphics?
  - a. It is appropriate for situations where a Drawable Resource will not work
  - b. the Canvas API allows an application very find grained control
  - c. The Canvas API uses the painter's model to draw 2D graphics
  - d. All of these

---

## 2.6 Let Us Sum Up

---

While studying this unit, we have learnt that new Fragment API for android introduces in android 3.0, which permits easier dynamic user interfaces. API is application program which interface sets the routines, protocols, and tools for building software applications.

It is studied that every user interface elements in android app that are built using view and ViewGroup objects where view serves as object that draws one thing on screen which user will move while ViewGroup serves as object that holds alternative View objects as define in layout of interface.

It is seen that a layout defines visual structure for a program, like UI for an activity or app convenience that you'll be able to declare in a layout by two ways

It is noted that animations are great way to improve user experience of an application which help it stand out. The best animations are the ones that users don't notice because they feel natural.

---

## 2.7 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 -b), (2-d)

**Check your progress 2**

**Answers:** (1 -a), (2-b)

**Check your progress 3**

**Answers:** (1 -c)

**Check your progress 4**

**Answers:** (1 -c), (2-d)

---

## 2.8 Glossary

---

1. **Intent** - Message object which uses to communicate with certain applications/activities asynchronously.
2. **Intent Filter** - It is an object filter application which is present in manifest file which shows system with its types of components.
3. **Animation** - Motion of objects created with software that allows moving an image in motion graphics.

---

## 2.9 Assignment

---

Write differences among canvas and drawable resources for creating graphics.



---

## 2.10 Activities

---

Design UI of a registration system.

Android User  
Interface  
Design and  
Common APIs

---

## 2.11 Case Study

---

Study UI and Layout in android.

---

## 2.12 Further Readings

---

1. Learn Java for Android Development (2nd edition), Jeff Friesen, 2013.
2. Android Application Development for Java Programmers, James C. Sheusi, 2012.

---

## Block Summary

---

In this block, you have learnt and understand about the basic of asynchronous messages and use of permissions for developing Android features. The block gives an idea on use of Intent in interacting with components along with extra components. You have been well explained on the various criteria's and rules to be followed for permission of rights.

The block detailed about the basic of visual structure for a program along with usability of AndroidManifest.xml file. The concept related to User Interface with respect to activity or applications are also explained to you. You will be demonstrated practically about 2D and 3D animations in Android.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. How animation can be applied in android?
2. What is the difference between intent and intent filters
3. What is view group?
4. What is meant by permission and how it can be provided?

### **Long Answer Questions**

1. What is android manifest file? Explain in detail by providing its structure.
2. What is the difference between relative layout and linear layout?
3. Create a login form and study its manifest file.
4. How can we manage different type's application resources in android?

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# OBJECT ORIENTED ANALYSIS AND DESIGN

PGDCA -204



## BLOCK 2: OBJECT ORIENTED ANALYSIS AND SYSTEM DESIGN

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# OBJECT ORIENTED ANALYSIS AND DESIGN

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

PROF. K J Sharma

### **Language Editor**

Mr. Vipul Shelke

## **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

## **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by means of, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

## **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'





## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# OBJECT ORIENTED ANALYSIS AND DESIGN

## Contents

### **BLOCK 1: OBJECT ORIENTED MODELLING**

#### **UNIT 1 INTRODUCTION TO OBJECT ORIENTED MODELLING**

Object Oriented Modeling, Characteristics Object Oriented Modeling: Class and Objects, Links and Association, Generalization and Inheritance, An Object Model. Benefits of OO Modeling, Introduction to OOAD tools,

#### **UNIT 2 ADVANCE MODELING CONCEPTS**

Aggregation, Abstract Class, Multiple Inheritance, Generalization as an Extension, Generalization as a Restriction, Metadata, Constraints, An Object Model

---

### **BLOCK 2: OBJECT ORIENTED ANALYSIS AND SYSTEM DESIGN**

#### **UNIT 1 OBJECT ORIENTED ANALYSIS**

Object Oriented Analysis, Problem Statement: an Example, Differences between Structured Analysis and Object Oriented Analysis, Analysis Techniques: Object Modeling, Dynamic Modeling, and Functional Modeling. Adding Operations. Analysis Iteration

#### **UNIT 2 SYSTEM DESIGN**

System Design: An Object Oriented Approach, Breaking into Subsystems, Concurrency Identification, Management of data store, Controlling events between Objects, Handling Boundary Conditions

---



**BLOCK 3: OBJECT DESIGN AND DYNAMIC MODELING**

**UNIT 1 OBJECT DESIGN**

Object Design for Processing, Object Design Steps, Designing a Solution, Choosing Algorithms, Choosing Data Structures, Defining Classes and delegation of Responsibilities to Methods

**UNIT 2 DYNAMIC MODELING**

Events, State and State Diagram, Elements of State Diagrams, Examples of State Diagrams, Advance Concepts in Dynamic Modeling, Concurrency, A Dynamic model

**BLOCK 4: FUNCTIONAL MODELING AND UML**

**UNIT 1 FUNCTIONAL MODELING**

Functional Models, Data Flow Diagrams, Features of a DFD, Design flaws in DFD, A Functional model, Relationship between Object, Dynamic, and Functional Models

**UNIT 2 USING UML**

UML: Introduction, Object Model Notations: Basic Concepts, Structural Diagrams: Class, Object, Composite, Package, Component, Deployment. Behavioral Diagrams: Use Case, Communication, Sequence, Interaction Overview, Activity, State. Modeling with Objects

**UNIT 3 CASE STUDY**

This unit will cover all the OOAD aspects Covered in previous units of this course.



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA-204

# **OBJECT ORIENTED ANALYSIS AND DESIGN**

---

## **BLOCK 2: OBJECT ORIENTED ANALYSIS AND SYSTEM DESIGN**

---

### **UNIT 1**

#### **OBJECT ORIENTED ANALYSIS**

03

### **UNIT 2**

#### **SYSTEM DESIGN**

14

---

# **BLOCK 2: OBJECT ORIENTED ANALYSIS AND SYSTEM DESIGN**

---

## **Block Introduction**

Object oriented design is totally contrary to both structured methodologies and information engineering. "Both structured design and information engineering use function-oriented decomposition rules, resulting in a structure of procedure-oriented program modules that contain programs, subroutines, and functions with solely procedural code.

In this block, we will detail about the basic of Analysis Techniques such as Object Modelling, Dynamic Modelling and Functional Modelling. The block will focus on the study and concept of Structured Analysis and Object Oriented Analysis. You will show comparison about Structured Analysis and Object Oriented Analysis.

In this block, you will make to learn and understand about the Object oriented analysis design approach supporting system as group of objects in terms of logical frame system. The concept related to Object-oriented analysis with groups items interaction of class, data or behaviour also explained to you.

## **Block Objective**

**After learning this block, you will be able to understand:**

- Understanding of object oriented design;
- Knowledge about breaking of system to subsystems;
- Features of software design
- Identifying handle concurrency;
- Features of object oriented analysis
- Idea about Structured Analysis and Object Oriented Analysis
- Understanding of Analysis Techniques

|                        |                                 |
|------------------------|---------------------------------|
| <b>Block Structure</b> |                                 |
| <b>Unit 1:</b>         | <b>Object Oriented Analysis</b> |
| <b>Unit 2:</b>         | <b>System Design</b>            |



---

# **UNIT 1: OBJECT ORIENTED ANALYSIS**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Object Oriented Analysis**
- 1.3 Problem Statement: an Example**
- 1.4 Differences between Structured Analysis and Object Oriented Analysis**
- 1.5 Analysis Techniques: Object Modeling, Dynamic Modeling, and Functional Modeling**
- 1.6 Adding Operations**
- 1.7 Analysis Iteration**
- 1.8 Let Us Sum Up**
- 1.9 Answer for Check Your Progress**
- 1.10 Glossary**
- 1.11 Assignment**
- 1.12 Activities**
- 1.13 Case Study**
- 1.14 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About Object Oriented Analysis
- About Structured Analysis
- About Object Modeling
- About Functional Modeling

---

## 1.1 Introduction

---

In the system analysis or object-oriented analysis part of software development, the system needs are determined, the classes are identified and the relationships among classes are known. The 3 analysis techniques that are used in conjunction with one another for object-oriented analysis are object modelling, dynamic modelling, and functional modelling.

---

## 1.2 Object Oriented Analysis

---

Object oriented analysis related to the methodology which combines the item that mix with other in terms of class, data or behaviour, thereby creating a model which perfectly shows the required purpose of system. Object-oriented analysis will not factor on the implementation, limitations in the model.

The advantage of using object oriented methodology is to create what you already contain. With this approach, the ability to use the code any time and develop additional maintainable systems in lesser amount of time is possible. Additionally, object-oriented systems are better designed, more resilient to change, and more reliable, since they're built from fully tested and debugged classes.

### Check your progress 1

1. Object Oriented analysis includes\_\_\_\_\_.
- a. Object modeling
- b. Dynamic modeling
- c. Functional modeling.
- d. All of these

---

## 1.3 Problem Statement: an Example

---

You must understand that you are looking for a statement of requirements, not a proposal for a solution. OOA specifies the structure and the behaviour of the object that comprises the requirements of that specific object. Different types of models are required to specify the requirements of the objects. These object

models contain the definition of objects in the system, which includes: the name of object, its attributes, and its relation with other object. As you know, an object is a representation of a real-life entity, or an abstraction. For example, objects in a flight reservation system might include: an airplane, an airline flight, an icon on a screen, or even a full screen with which a travel agent interacts. The behaviour, or state model describes the behaviour of the objects in terms of the states of the object and the transitions allowed between objects, and the events that cause objects to change states.

These models are created and maintained using CASE tools that support the illustration of objects and object behaviour.

You can say that the problem statement could be a general description of the user's difficulties, desires, and its purpose is to identify the overall domain within which you will be operating, and give some plan of why you're doing OOA.

It is vital for an analyst to separate the true necessities from design and implementation decisions.

Problem statement shouldn't be incomplete, ambiguous, and inconsistent. Attempt to state the requirements precisely, and to the purpose. Do not make it cumbersome. Some requirements appear reasonable but don't work. These kinds of requirements should be identified. the aim of the subsequent analysis is to fully understand the problem and its implications, and to bring out the true intent of the client.

### **Check your progress 2**

1. What can be the objects in a library management system?
  - a. Books
  - b. Library
  - c. Student
  - d. All of these

---

## 1.4 Differences between Structured Analysis and Object Oriented Analysis

---

Object oriented analysis design (OOAD) is essentially a bottom up approach that supports viewing the system as a collection of components (objects) which will be logically kept together to create the system.

Advantages and disadvantages of Object oriented Analysis and design

### **Advantages:**

The OO approach inherently makes each object a standalone component which will be reused not only within a particular stat problem domain; however are also completely different downside domains, having the requirement of similar objects.

The other main advantage of object oriented (OO) is that the focus on data relationships. You cannot develop a successful system where data relationships aren't well understood. An OO model provides all of the insight of an ER diagram and contains additional information related to the ways to be performed on the data.

### **Disadvantages:**

You know that OO methods only build functional models within the objects. There's no place within the methodology to build a complete functional model. Whereas this is not a problem for a few applications (e.g., building a software toolset), except for large systems, it will lead to missed needs.

Another disadvantage of the object oriented analysis design (OOAD) is in system modelling for performance and sizing. The object oriented (OO) models don't simply describe the communications between objects. Indeed, a basic concept of object oriented (OO) is that the object needn't know who is invoking it. Whereas this results in a flexible design, performance modelling can't be handled simply.

The object oriented (OO) analysis design itself doesn't give support for identifying that objects can generate an optimal system design. Specifically, there's no single diagram that shows all of the interfaces between objects.

### **Advantages and disadvantages of Structured Analysis**

With expertise, you'll come to know to grasp that almost all customers understand structured methods better than object oriented (OO) ways. Since one of the main reasons of modelling a system is for communication with customers

and users, there's an advantage in providing structured models for information exchange with user groups or customers.

The disadvantage with structured methods is that they do not readily support the use of reusable modules. The top down method works well for new development, however doesn't give the mechanisms for “designing in” the use of existing components. The top down process of functional decomposition doesn't lead to a set of requirements that map well to existing components.

### **Check your progress 3**

1. Object oriented approach is useful because?
  - a. It supports reusability of modules
  - b. It is top – down approach
  - c. It can handles performance modeling
  - d. It describes the communication between objects

---

## **1.5 Analysis Techniques: Object Modelling, Dynamic Modelling, and Functional Modelling**

---

Object Modelling Technique is a methodology that deals with object-oriented development in analysis and design phases. In analysis phase, a problem statement carries list of goals and key ideas within a domain. Problem statement seems in three views or models:

- Object model
- Dynamic model
- Functional model

An object model shows artifacts of system showing static and stable phenomena in model domain. The idea behind it's classes and associations with attributes and operations. Aggregation and generalization are predefined relationships

The notations used in object model are shown in fig 1.1

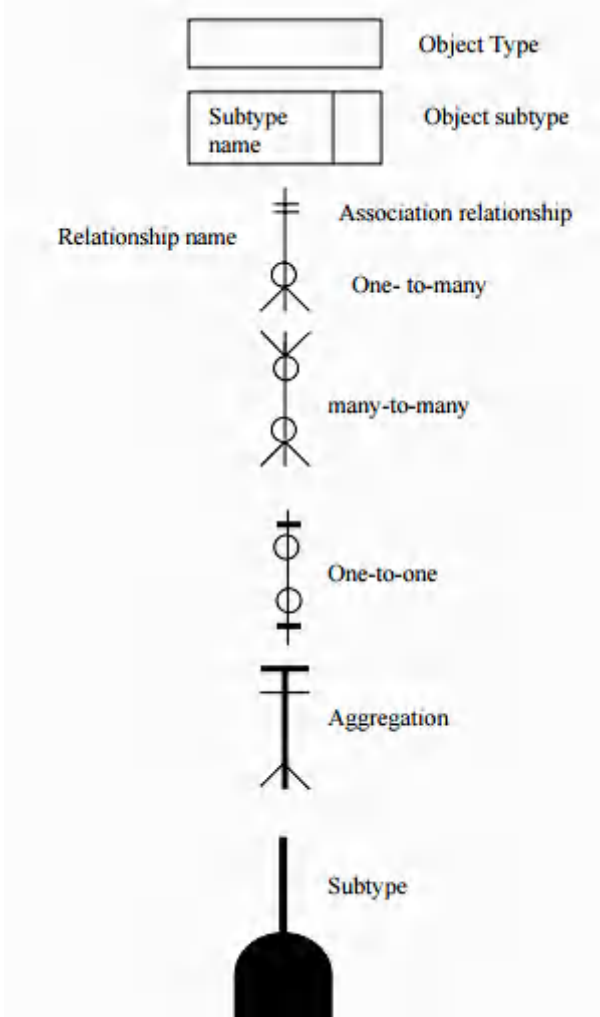


Fig 1.1 Notations in Object Model

The dynamic model shows interaction among such artifacts that shows events, states and transitions. It shows a state/transition view on model having states, transitions that exist among states and events.

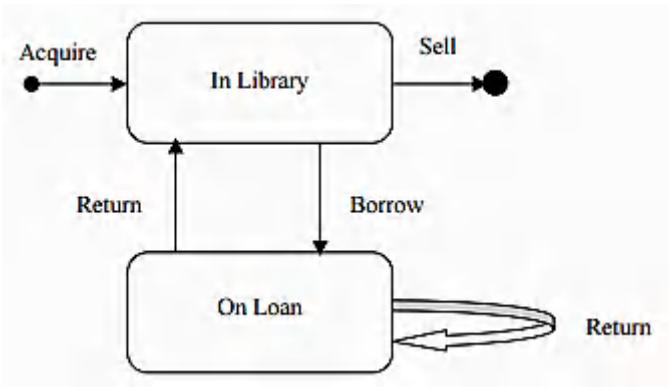


Fig 1.2 State Diagrams

In this, the actions are modelled inside the states. The notations used in this modelling is shown in fig 1.3

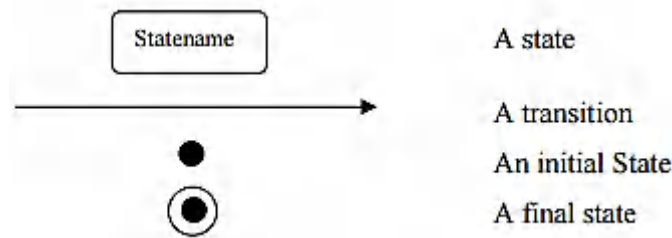


**Fig 1.3 Notations**

The functional model carries system ways from perspective of data flow. It handles the process viewpoint of model that corresponds to data flow diagrams. It carries main ideas related to process, data store, data flow and actors. Steps employed in framing functional Model comprises of:

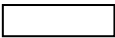
- Finding input/output values
- Creating data flow diagrams with functional dependencies
- Explaining the functions
- Locating constraints
- Showing rules of optimisation

The system design phase moves along with analysis phase where architecture is established and is arranged in subsystems that allocates processes and tasks with account concurrency and collaboration.

The object design phase is with system design phase wherever plan gets implemented. Object classes are carried out with algorithms having attention to optimize path to persistent data.

### Check your progress 4

1. Which notation is used to describe the initial state of an object in object modeling?

- a. 
- b. 
- c. 
- d. None of these

---

## 1.6 Adding Operations

---

Whenever you examine the operations in OOPs you find queries regarding attributes or associations within the object model, to events within the dynamic model and functions within the functional model. You can see operations from the object model. It includes reading and writing attribute values and association links. Within the object model operations are presented with an attribute. During analysis nothing is private, you assume that all attributes are accessible.

During analysis events are sent to the target objects. An operation on the object is presented as labels on transitions and should not be explicitly listed in the object model. The functions should be easy and summarized on the object model. Organise the functions into operations on objects.

### Check your progress 5

1. What is the use of == operator?
  - a. Checks if the values of two operands are equal or not
  - b. Checks if the values of two operands are equal or not
  - c. Checks if the value of left operand is greater than or equal to the value of right operand,
  - d. None of these

---

## 1.7 Analysis Iteration

---

To understand any problem properly you have to repeat the task which means that analysis needs repetition. First, simply get the overview of the problem, build a rough draft, and then, reiterate the analysis according to your understanding. Finally, you must verify the final analysis with the client or application domain experts. During the iteration processes refining analysis and restating of requirement takes place.

### Refining the ratio Analysis

Basically, refinement leads to purity. Thus to get a cleaner, more understandable and coherent design, you would like to reiterate the analysis process.



- Re-examine the model to remove inconsistencies, and imbalances with and across the model.
- Remove anomaly, and wrong ideas from the model.
- Refining sometimes involves restructuring the model.
- Define constraints within the system in a very better way.
- You should keep track of generalizations factored on the wrong attributes.
- Include exceptions within the model, many special cases, lack of expected symmetry, and an object with two or more sets of unrelated attributes, or operations.
- You should remove extra objects or associations from the model. Also, take care to remove redundancy of objects and their extra attributes.

### Restating the necessities

To have clarity of the analytical model of the system you must state the necessities specific performance constraints with the optimization criteria in one document verify within the different document. You'll state the method of a solution. Verify the ultimate model with the client. The requirement analysis should be confirmed and clearly understood by the client.

### Check your progress 6

1. What is Iteration?
  - a. Repetition of design process
  - b. Reusability
  - c. Re-examine the model to remove inconsistencies
  - d. All of these

---

## 1.8 Let Us Sum Up

---

Object oriented analysis (OOA) describes what the customer requires, it establishes as basis for the creation of a software design, and it defines a collection of needs which will be validated. You also remember that the requirement analysis should be designed in such the simplest way that it should tell you what to be done, not how it's implemented.

The object model is that the principal output of an analysis and design process. Dynamic modelling is elaborated further by adding the concept of time: new attributes are computed as a function of attribute changes over time.

---

## 1.9 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –d)

**Check your progress 2**

**Answers:** (1 –d)

**Check your progress 3**

**Answers:** (1 –a)

**Check your progress 4**

**Answers:** (1 –a)

**Check your progress 5**

**Answers:** (1 –a)

**Check your progress 6**

**Answers:** (1 –c)

---

## 1.10 Glossary

---

1. **Object-oriented analysis** - Process having packed with teems which mixup with by class, data or behaviour for modeling purpose.

---

## 1.11 Assignment

---

Explain how you can define an object model of a system.

---

## 1.12 Activities

---

Give an example of operations from State Actions and Activities.

---

## 1.13 Case Study

---

Show the basic dynamic model of telephone.

---

## 1.14 Further Readings

---

1. Norman, Ronald- object oriented system analysis and design –prentice hall 1996.
2. Coad. P and Yourdon .E – “Object Oriented Analysis” – Yourdon press.
3. Coad. P and Yourdon .E – “Object Oriented Design” – Yourdon press.
4. Rambaugh, James, Michael –Object oriented Modelling and design.

---

## **UNIT 2: SYSTEM DESIGN**

---

### **Unit Structure**

#### **2.0 Learning Objectives**

#### **2.1 Introduction**

#### **2.2 System Design: An Object Oriented Approach**

#### **2.3 Breaking into Subsystems**

#### **2.4 Concurrency Identification**

#### **2.5 Management of data store**

#### **2.6 Controlling events between Objects**

#### **2.7 Handling Boundary Conditions**

#### **2.8 Let Us Sum Up**

#### **2.9 Answers for Check Your Progress**

#### **2.10 Glossary**

#### **2.11 Assignment**

#### **2.12 Activities**

#### **2.13 Case Study**

#### **2.14 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Overview of object oriented design
- Breaking down system to subsystems
- Explain about software design as set of interacting objects managing states and operation
- Identify concurrent object, and explain how to handle concurrency

---

## 2.1 Introduction

---

System design is that the process of defining the components, modules, interfaces, and data for a system to satisfy specified requirements. System development is that the process of creating or altering systems, along with the processes, practices, models, and methodologies used to develop them. Object oriented design (OOD) is concerned with developing an object oriented model of a software system to implement the identified needs. Several OOD methods are delineated since the late 1980s.

---

## 2.2 System Design: An Object Oriented Approach

---

If design stage is a set of objects with their responsibilities and collaborators. These objects will most likely become classes, wherever the actual objects in the system are going to be instances of these classes. By creating the item definitions into classes, we will create as many of the object as we want and handle growth and complexity within the system. The outcome of the design stage could be a description detailed enough to come from. The design carries 2 parts:

- A {class|a category} diagram defining the attributes and services of {each of every} class and formally identifying the connections between each class.
- A detailed description of what each service is supposed to try and do.

Software systems designed with structured design methodology don't support some of the required quality attributes like reusability, portability and mapping to the problem domain. Many large organisations find that systems designed with structured approaches are less reusable and fewer maintainable than those designed with object-oriented approaches.

- Although object oriented methods have roots that are powerfully anchored back within the 60s, structured and functional methods were the first to be used. This is not very uncommon, since functional methods are inspired directly by computer architecture (a proven domain, standard to computer scientists). The separation of data and program as exists physically within the hardware was translated into functional methods. This is the reason why computer scientists got into the habit of thinking in terms of system functions. Later it absolutely was felt that hardware should act as the servant of the software that's executed on it rather than imposing architectural constraints on system development method.

## Object Oriented Analysis and System Design

- Traditional System Analysis and design: traditional System Analysis and design (SAD) has 3 basic life cycle models. A typical computer code lifecycle consists of following phases:
- Planning
- Development
- Maintenance

### Key Criteria

- a well-defined methodology
- traceability among steps
- documentation
- control

### Software Life Cycle

The software design consists of a number of stages, or phases

- Requirement Phase– determine use need
- Specification Phase– define requirement
- Design Phase: design the system
- Implementation Phase– fabricate the system
- Testing and Integration Phase– testing the fabricated system
- Maintenance phase
- Retirement

### An object-oriented design process

1. Define the context and modes of Object Oriented Design
2. Designs the system architecture
3. Identifies the principal system objects
4. Identifies concurrency in the problem
5. Handling boundary conditions
6. Develops design models
7. Specifies object interfaces

**Check your progress 1**

1. Which phase of software development life cycle involves designing of system?
  - a. Design phase
  - b. Implementation phase
  - c. Requirement phase
  - d. None of these

---

## 2.3 Breaking into Subsystems

---

Object-oriented decomposition aims at identifying individual autonomous objects that encapsulate both a state and a certain behaviour. Each major component of the system is named a subsystem. Then communication among these objects leads to the specified solutions. Decomposition of system in function-data model (SAD) and object oriented decomposition is given in fig 2.1.

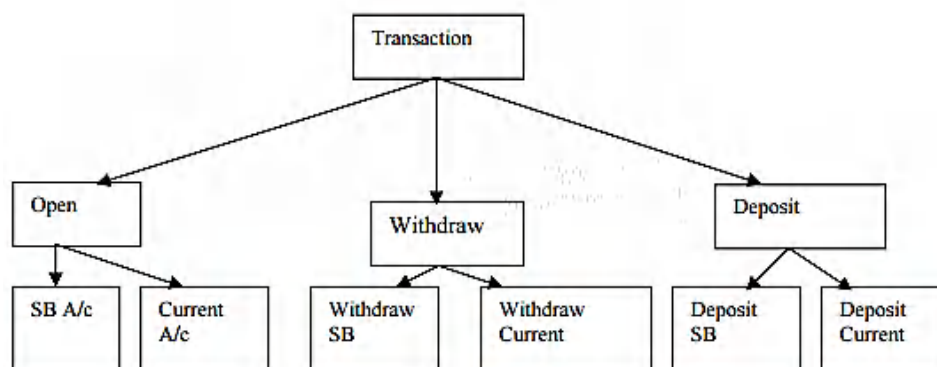


**Fig 2.1 Functional Data Decomposition**

Although each solution facilitates deal with complexity, we've got reasons to believe that an object-oriented decomposition is favourable because, the object-oriented approach provides for a semantically richer framework that leads to decompositions that are more closely related to entities from the real world. Moreover, the identification of abstractions supports having (more abstract) solutions to be reused, and therefore the object-oriented approach supports the evolution of systems better, as those ideas that are additional seemingly to alter can be hidden within the objects.

Object-oriented decompositions of systems tend to be better able to deal with modification. Each subsystem incorporates a well-defined interface that

communicates with rest of the system. Each of these interfaces defines all form of interaction that is needed for proper functioning of the whole system; however the internal implementations are left to the sub-system itself. This can be because they manage to encapsulate those items that are likely to alter (such as functionality, sequence of behaviour and attributes) within an object and hide them from the outside world. The advantage is that the outside cannot see them, and therefore cannot be dependent on them and does not need to be changed if these items change. Also, object-oriented decompositions are closer to the problem domain, as they directly represent the real-world entities in their structure and behaviour. The abstraction primitives built into reuse have a huge potential of reuse as commonalities between similar objects is factored out, and then, the solutions is reused. Finally, object-orientation has the advantage of continuity throughout analysis, design implementation, and persistent representation.

### **Advantages of Decomposition**

- Separate people can work on each subsystem.
- Every software engineer must specialize in a domain.
- Every individual component is smaller, and therefore easier to understand and manage.
- Part of the subsystem can be replaced or changed without having replaced or extensively changed the other subsystems.

### **Check your progress 2**

1. Why decomposition of system is required?
  - a. Sub systems are able to deal with modifications efficiently
  - b. Each subsystem incorporates a well-defined interface that communicates with rest of the system
  - c. Small components are easy to manage
  - d. All of these



## 2.4 Concurrency Identification

Concurrency in objects may be identified by the manner they change state. Current objects will change state independently. Aggregation implies concurrency. Concurrency in OOAD study is described and studied in dynamic modelling. If the objects are concurrent in nature we've to assign them to, different thread of control. For instance, withdraw and deposit operations related to a bank account is also executed in parallel (concurrently).

Concurrency is identified in a dynamic model. 2 objects are said to be concurrent (parallel) if they will receive events at the same time. Concurrent objects are required to be assigned to different threads of control.

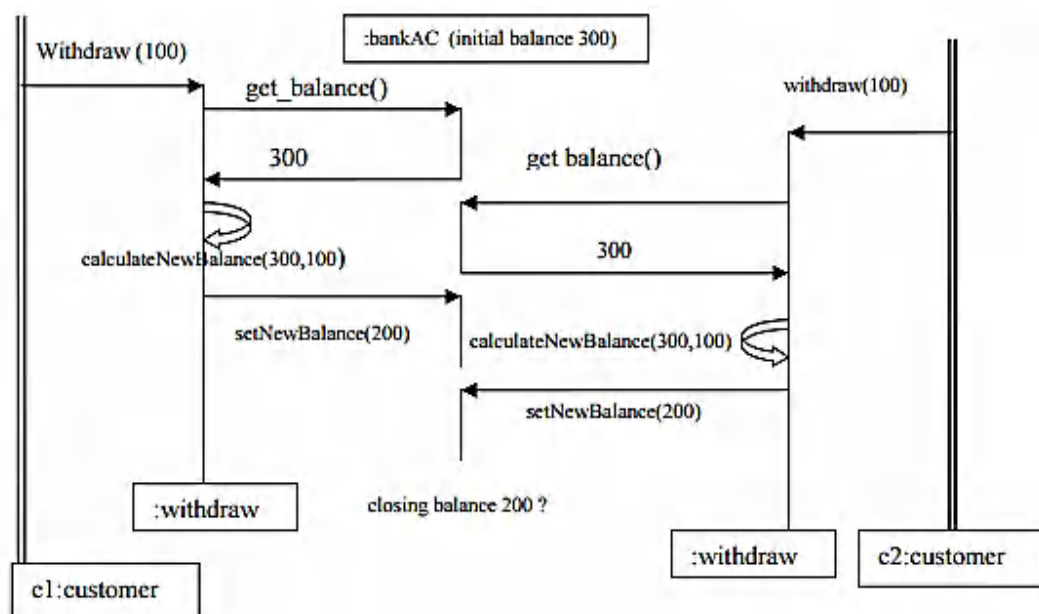


Fig 2.3 Concurrency without synchronisation

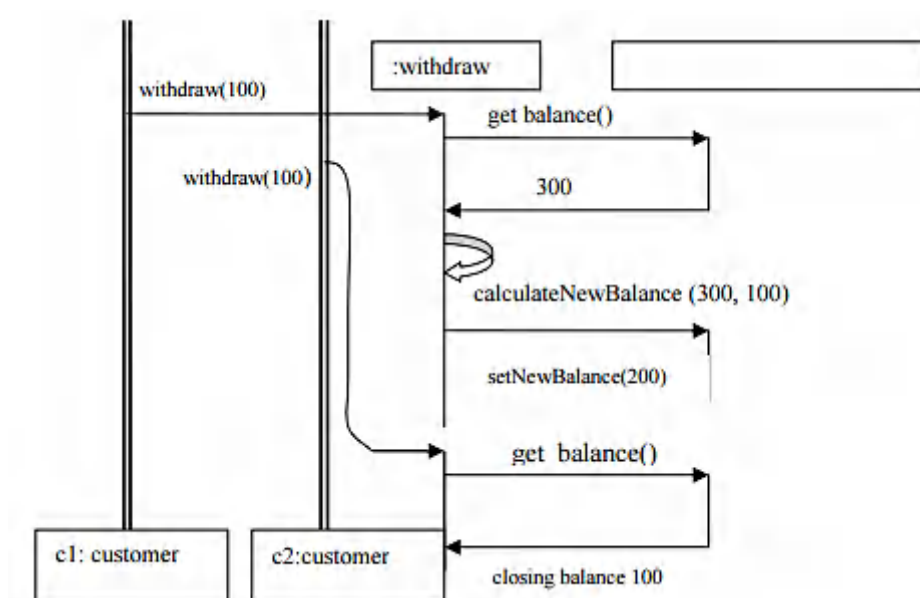


Fig 2.4 Concurrency with synchronisation

### Check your progress 3

1. Two objects are said to be concurrent\_\_\_\_\_.
- a. When they will receive events concurrently
- b. If two objects are same
- c. If they receive same events
- d. None of these

## 2.5 Management of data store

Some objects within the models need to be persistent, to store the state of the object permanently in database. Most systems need persistent data that isn't lost when the system closes down. These data are stored in file system, in a or database. Object-oriented applications often use relational databases to provide persistence.

Designer needs to:

1. Determine what data needs to be persistent
2. Design a suitable database schema for the database.

In most of the cases, persistent class maps to one relational table (leaving aside the inheritance issue, for the moment). Within the simplest model, a class

20

from the logical model maps to a relational table, either in whole, or in part. The logical extension of this can be that a single object (or instance of a class) maps to a single table row. Persistent object is stored with one of the following:

#### Files

- Cheap, simple, permanent storage
- Low level (Read, Write)
- Applications should add code to provide a suitable level of abstraction.

#### Relational database

- Standardized, easy to port
- Supports multiple writers and readers
- Mature technology

#### Object database

- One-to-one mapping from the analysis model
- Associations are directly represented
- Slower than relational databases for complex queries

#### **Issues once selecting a database**

- Storage space requirement: A database requires about triple the storage space of actual information.
- Response time: The response time of the database for I/O or communication bound (in case of distributed databases) request.
- Locking modes pessimistic locking: Lock before accessing an object and unharnessed once object access is complete.

Optimistic locking: read and writes might freely occur (high degree of concurrency).

- Administration: trendy software system needs specially trained support staff to line up security policies, manage the disk space, prepare backups, fine-tune the performance, and monitor performance.

### **Check your progress 4**

1. What is optimistic locking mode?
  - a. It is a Lock that is used before accessing an object
  - b. It is a high degree of concurrency lock
  - c. It restricts read and write operations
  - d. None of these

---

## **2.6 Controlling events between Objects**

---

Examples of events are click and flight leaving from an airport. An event doesn't have a fixed duration. Each factor that happens modelled as an event. After an event, objects change their state, and these are represented by a state diagram.

Events are classified as four types in UML

1. Signals
2. Calls
3. Passing of time
4. Change in State

Events additionally include inputs, decisions, interrupts and actions performed by users or any external device. Every event has a sender and receiver. In most of the cases the sender and receives are the same object. A state without a predecessor and successor are ambiguous, and care should be taken to represent initiations and termination of events. Events that have same effect on the control flow must be grouped together though their value differs. The events are to be allocated to the object classes that send/receive it.

Most of the look issues of systems are concerned with steady-state behaviour. However, the system design phase should additionally address the initiation and finalization of the system. This is often addressed by a set of new uses cases referred to as administration use cases. Now, let us discuss however these problems are handled.

**Check your progress 5**

1. Unified Modeling Language(UML) classifies events into\_\_\_\_\_.
- Signals
  - Calls
  - passing of states
  - All of these

---

## 2.7 Handling Boundary Conditions

---

These are some conditions that to be handled in any system initialization and termination

- Describes however the system is brought from a non-initialized state to steady-state. It describes normal operations like start-up, shutdown, reconfiguration, restart/reset, backup, etc.
- Describes what resources are clean up, and that systems are notified upon termination.
- Describes however the system is adapted to a local installation. Failure
- Unplanned termination of the system.

Many possible causes: failure of system hardware, bugs, operator errors, external problems.

- Sensible system design foresees fatal failures.

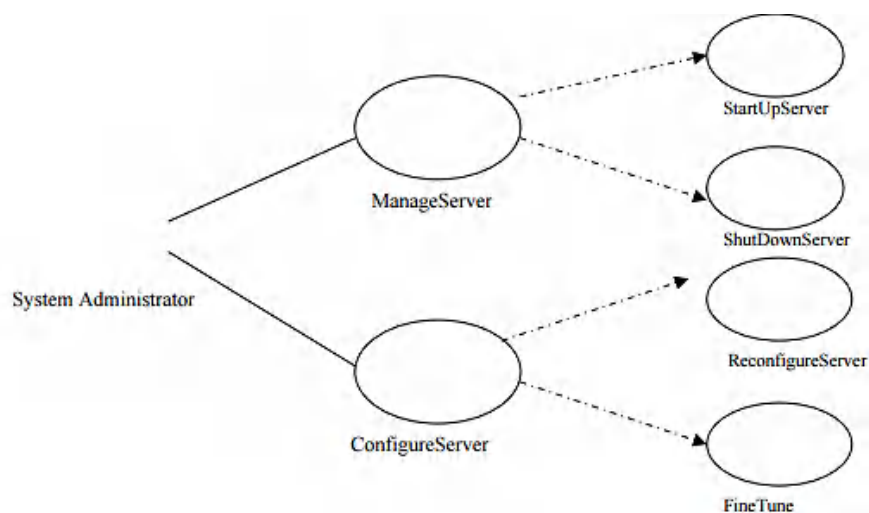


**Fig 2.5 Boundary conditions**

### Check your progress 6

1. What is the need of using boundary conditions?
  - a. It describes normal operations like start-up, shutdown etc.
  - b. It describes how the system is adapted to a local installation
  - c. It describes termination use cases
  - d. All of these

---

## 2.8 Let Us Sum Up

---

In this unit we have learnt that OOD is a technique which is helpful for development of huge and complex systems. It is seen that moving from functional approach to object-oriented needs a translation of functional model elements into object model elements that is easy.

It is noted that high-level system style approach involves breaking down of system into easy and comparatively independent sub systems for higher manageability. Concurrency is inherent to objects, and concurrent objects can't be folded into one thread of control. Concurrency should be dealt with during the design method as dealing with concurrency after the system is implemented is difficult.

Data are often stored in flat files or database management system. Object-oriented databases provide support for all fundamental object modelling concepts like classes, Attributes, Methods. Associations and Inheritance are object-oriented databases which lack theoretical foundation and standards which is concerned with system design in handling boundary conditions.

---

## 2.9 Answers for Check Your Progress

---

### Check your progress 1

**Answers:** (1 –a)

### Check your progress 2

**Answers:** (1 –d)

**Check your progress 3**

**Answers:** (1 –a)

**Check your progress 4**

**Answers:** (1 –b)

**Check your progress 5**

**Answers:** (1 –d)

**Check your progress 6**

**Answers:** (1 –d)

---

## 2.10 Glossary

---

1. **Object oriented analysis design** - A bottom up approach which supports system as group of objects that is logical to frame the system.

---

## 2.11 Assignment

---

What do you mean by Concurrency in objects?

---

## 2.12 Activities

---

What are the advantages of decomposing a system?

---

## 2.13 Case Study

---

Discuss the broad steps of an object-oriented design process.

---

## 2.14 Further Readings

---

1. O.M. Nierstrasz, A Survey of Object-Oriented Concepts.
2. In W. Kim, F. Lochovsky (eds.): Object-Oriented Concepts

---

## Block Summary

---

In this block, you have learnt and understand about the basic of software design and identification of concurrency. The block gives an idea on the study and concept of object oriented analysis design with detailed about support approach in terms of grouping of objects. You have been well explained on the concepts of structured analysis and object oriented analysis.

The block detailed about the basic of various analysing techniques such as object modelling, dynamic modelling and functional modelling. The concept related to object oriented analysis design bottom up approach in terms of system support are well explained to students. You will be demonstrated practically about Object Modelling.



---

## **Block Assignment**

---

### **Short Answer Questions**

1. Explain Iteration Analysis?
2. Define the steps used in constructing Functional Model?
3. What are the shortcomings in structured approach?
4. Differentiate between object oriented decomposition and structured decomposition?

### **Long Answer Questions**

1. How do you see the final model after iterative analysis?
2. What are the benefits of OOA technology?
3. Explain how you can define an object model of a system.

Enrolment No.

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                   | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|-----------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                    | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Language and Style                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Illustration used (Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Conceptual Clarity                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Check your progress Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |
| Feed back to CYP Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <div></div>                  |

3. Any Other Comments



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

'Jyotirmay Parisar', Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# WEB APPLICATION DEVELOPMENT

PGDCA 202

## BLOCK 2: OO AND FILE HANDLING IN PHP

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# WEB APPLICATION DEVELOPMENT



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Mr. Sanjay Thapar

### **Language Editor**

Mr. K. J. Sharma

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)





## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# WEB APPLICATION DEVELOPMENT

## Contents

### **BLOCK 1: INTRODUCTION TO PHP**

#### **UNIT 1 INTRODUCTION TO WEB APPLICATION**

The architecture of a web application, static and dynamic web application, Installing WAMP.

#### **UNIT 2 PHP BASICS**

Embedding PHP Code in Your Web Pages, Commenting Your Code, Outputting Data to the Browser, PHP's Supported Data types, Identifiers, Variables, Constants, Expressions, String Interpolation, Control Structures.

### **BLOCK 2: ARRAY, FUNCTION AND EXPRESSION**

#### **UNIT 1 ARRAY IN PHP**

Array introduction, creating an array, adding and removing array element, determining array size and uniqueness sorting array,

#### **UNIT 2 FUNCTION IN PHP**

In built functions: String functions, array functions, mathematical functions, File System functions, Date and Time Functions, Miscellaneous Functions, User Defined Functions, arguments passing by reference , Argument passing by value

#### **UNIT 3 FUNCTION IN PHP**

Regular Expression, Error Handling Regular Expressions, Exception handling

---

## **BLOCK 3: OO AND FILE HANDLING IN PHP**

### **UNIT 1 OBJECT ORIENTED PHP**

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

### **UNIT 2 FILE AND DIRECTORY HANDLING**

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

---

## **BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP**

### **UNIT 1 PHP AND MYSQL DATABASE**

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

### **UNIT 2 STATE MANAGEMENT IN PHP**

Session, Cookies



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA 202

## Web Application Development

---

### BLOCK 3: OO AND FILE HANDLING IN PHP

---

#### UNIT 1

Object Oriented PHP

03

#### UNIT 2

File and Directory Handling

36

---

# BLOCK 3: OO AND FILE HANDLING IN PHP

---

## Block Introduction

Object-Oriented Programming is a type of programming which can be modular and reusable web applications. It is a methodology which frames an application that could be any sort of web based or windows based. In OOP, all will be around objects and class.

In this block, we will detail about the basic of Object-Oriented Programming and its modelling techniques. The block will focus on declaration of variables, functions, structures, unions, or enumerations with study about their syntax. The concept of structured data type and class in PHP are also explained.

In this block, the student will make to learn and understand about the basic of Comma Separated Values with their usability and necessity in programming the script. The concept related to directory support functions and knowledge related to read and operate directories and entries will also be explained to students. The student will be demonstrated practically about storing tabular data in plain text format.

## Block Objective

**After learning this block, you will be able to understand:**

- The basic of OOP Concepts.
- Features about Creating and Using class properties.
- Basic about Constructors and Destructors Methods.
- Features regarding Cloning and comparing objects.
- Idea about Class Inherit.
- Understanding about Namespaces.
- Idea about Read/Write entire/part of file.
- Features of read and write CSV data.
- Basic of Copy, rename and delete file.
- Idea about File uploading in PHP.

OO and file  
handling in PHP

## **Block Structure**

**Unit 1:     Object Oriented PHP**

**Unit 2:     File and Directory Handling**

---

# **UNIT 1: OBJECT ORIENTED PHP**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 The benefits of OOP**
- 1.3 Key OOP Concepts**
- 1.4 Create and Use class, properties**
- 1.5 Constructors and Destructors Methods**
- 1.6 Create and Use Object, class constant**
- 1.7 Static properties and method**
- 1.8 Loop through an object's properties**
- 1.9 Clone and compare objects**
- 1.10 Inspect an object**
- 1.11 Inherit a class**
- 1.12 Use the protected access modifier**
- 1.13 Create abstract classes and methods**
- 1.14 Create final classes and methods**
- 1.15 Work with interfaces**
- 1.16 Introducing Namespaces**
- 1.17 Let Us Sum Up**
- 1.18 Answers for Check Your Progress**
- 1.19 Glossary**
- 1.20 Assignment**
- 1.21 Activities**
- 1.22 Case Study**
- 1.23 Further Readings**

---

## 1.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About OOP
- About Constructors and Destructors Methods
- About Creating and Using Object class constant
- About Inspecting an object
- About protected access modifier

---

## 1.1 Introduction

---

Object-Oriented Programming is a type of programming which can be modular and reusable web applications. In PHP, OOP will help the code to remain flexible by defining at many places. It is the latest way of software development where all languages such as Java, PERL, PHP, C#, Ruby use in programming.

---

## 1.2 The benefits of OOP

---

Object oriented programming methodology frames an application which could be any type as web based or windows based applications. In OOP, all will be around objects and class. Use of OOP in PHP will form modular web application and can do activity in object model structure. It carries many benefits.

OOP invokes usage of classes in order to arrange data and structure of an application. It is noted that large programmers avoids the use of OOP as programming theory. It is noted that objects in programming is similar to real word object where all programming object carries properties and behaviours.

### **Check your progress 1**

1. Object oriented programming methodology is:
  - a. web based
  - b. windows based
  - c. both a and b
  - d. neither a nor b



---

## 1.3 Key OOP Concepts

---

In terms of programming, object-oriented is hard concept having difficult syntax and roadblocks. Object-oriented programming serves as unique coding style which allows developers to assemble similar work into classes. The concept related to OOP is DRY programming which describes a piece of information when changes in a program, normally required so as to update the programming code. OOP shows threatening to developers since it introduces new syntax and appears to be complex as simple procedural or inline code.

### Check your progress 2

1. OOP is:
  - a. comfortable for users
  - b. simple for user
  - c. threatening for user
  - d. none of above

---

## 1.4 Create and Use class properties

---

While defining a class in PHP, you can add properties in it. It is hard to imagine that to create native class having regular non-typed public PHP properties. To add data to class, properties or class-specific variables are used which works just like regular variables. In order to add property to MyPlant, write the code as:

```
<?php
class MyPlant
{
    public $prop1 = "Rose is my Plant!";
}
$obj = new MyPlant;
var_dump($obj);
?>
```

In this code, keyword `public` will describe visibility of property, while property is named by standard variable syntax with certain value. If you have to read such property and give it to browser, then you need to reference the object from place where you read and read as:

```
echo $obj->prop1;
```

As multiple instances of class exist, when individual object is not referenced. In such case, the script is unable to find object which will read. You can alter the script in `test.php` so as to read the property instead of dumping whole class by simply changing the code as shown:

```
<?php  
class MyPlant  
{  
    public $prop1 = "My Plant is Rose!";  
}  
$obj = new MyPlant;  
echo $obj->prop1; // Output the property  
?>
```

Output:

My Plant is Rose!

### Check your progress 3

1. Which functions shows existence of class?
  - a. `exist()`
  - b. `exist_class()`
  - c. `class_exist()`
  - d. `__exist()`
2. \_\_\_\_\_ keyword describes properties or methods in class itself?
  - a. `exist`
  - b. `public`
  - c. `protected`
  - d. `$this`

---

## 1.5 Constructors and Destructors Methods

---

To start with an object, it is advisable to set few things. To handle this, PHP gives with magic method `__construct()` that is automatic when an object is created. To have an idea of constructors, you have to add constructor to `MyClass` which will output message when new instance of class is created:

```
<?php
class MyClass
{
    public $prop1 = "I'm a class property!";
    public function __construct()
    {
        echo "The class ", __CLASS__, " was initiated!<br />";
    }
    public function setProperty($newval)
    {
        $this->prop1 = $newval;
    }
    public function getProperty()
    {
        return $this->prop1 . "<br />";
    }
}

// Create a new object
$obj = new MyClass;

// Get the value of $prop1
echo $obj->getProperty();

// Output a message at the end of the file
echo "End of file.<br />";

?>
```

In this, we see that `__CLASS__` returns the name of class in which it is called which is magic constant. On reloading file in browser, we have the following result:

OO and file  
handling in PHP

The class "MyClass" was initiated!

I'm a class property!

End of file.

To call a function when object is damaged, the `__destruct()` magic method is available which is good for class cleanup. The message will be outputted when object gets destroyed by explaining magic method as:

`__destruct()` in MyClass:

```
<?php
class MyClass
{
    public $prop1 = "I'm a class property!";
    public function __construct()
    {
        echo "The class ", __CLASS__, " was initiated!<br />";
    }
    public function __destruct()
    {
        echo "The class ", __CLASS__, " was destroyed.<br />";
    }
}
```

```
public function setProperty($newval)
{
    $this->prop1 = $newval;
}
public function getProperty()
{
    return $this->prop1 . "<br />";
}
}
// Create a new object
$obj = new MyClass;
// Get the value of $prop1
echo $obj->getProperty();
// Output a message at the end of the file
echo "End of file.<br />";
?>
```

Output:

The class "MyClass" was initiated!

I'm a class property!

End of file

The class "MyClass" was destroyed.

"When the end of a file is reached, PHP automatically releases all resources."

To explicitly trigger the destructor, you can destroy the object using the function `unset()`:

OO and file  
handling in PHP

```
<?php
class MyClass
{
    public $prop1 = "I'm a class property!";
    public function __construct()
    {
        echo "The class ", __CLASS__, " was initiated!<br />";
    }
    public function __destruct()
    {
        echo "The class ", __CLASS__, " was destroyed.<br />";
    }
    public function setProperty($newval)
    {
        $this->prop1 = $newval;
    }
    public function getProperty()
    {
        return $this->prop1 . "<br />";
    }
}

// Create a new object
$obj = new MyClass;

// Get the value of $prop1
echo $obj->getProperty();

// Destroy the object
unset($obj);

// Output a message at the end of the file
echo "End of file.<br />";

?>
```

Output:

The class "MyClass" was initiated!  
I'm a class property!  
The class "MyClass" was destroyed.  
End of file.

### Check your progress 4

1. Which statements is true about Constructors?

- i PHP has class constructors.
- ii Constructors accept parameters.
- iii Constructors call class methods.
- iv Constructors call other constructors.

- a. ii) and iii)
- b. All of above
- c. i) and iii)
- d. ii), iii) and iv)

2. PHP identifies constructors by:

- a. classname()
- b. \_construct()
- c. function \_construct()
- d. function \_\_construct()

---

## 1.6 Create and Use Object class constant

---

You can define constant values on per-class basis keeping it same and unchangeable. Constants differ from normal variables where no use of \$ symbol is there in order to declare.

### Example #1 Defining and using a constant

```
<?php
```

```
class MyClass
```

OO and file  
handling in PHP

```
{
    const CONSTANT = 'constant value';

    function showConstant() {
        echo self::CONSTANT . "\n";
    }
}

echo MyClass::CONSTANT . "\n";
$classname = "MyClass";
echo $classname::CONSTANT . "\n"; // As of PHP 5.3.0
$class = new MyClass();
$class->showConstant();
echo $class::CONSTANT . "\n"; // As of PHP 5.3.0
?>
```

### Example #2 Static data example

```
<?php
class foo {
    // As of PHP 5.3.0
    const BAR = <<<'EOT'
bar
EOT;

    // As of PHP 5.3.0
    const BAZ = <<<EOT
baz
EOT;
}
?>
```

### Example #3 Constant expression example



```
<?php
const ONE = 1;

class foo {
    // As of PHP 5.6.0
    const TWO = ONE * 2;

    const THREE = ONE ÷ self::TWO;
    const SENTENCE = 'The value of THREE is ' . self::THREE;
}
?>
```

### Check your progress 5

1. Creation of objects based on predefined classes is:

- a. class object
- b. object class
- c. object instance
- d. class instantiation

2. Identify correct way to define constant?

- a. constant G = “9.18”;
- b. const \$G = “9.18”;
- c. constant G = ‘9.18’;
- d. const G = ‘9.18’;

---

## 1.7 Static properties and method

---

In PHP, static methods and properties is an important feature which is directly accessible without creating object of class. Static properties of class are directly available from class using scope resolution operator. Also, you can make property static using static keyword as shown in static variable in php class:

```
class test
{
    public static $a; //Static variable
}

test::$a = 5;

echo test::$a;
```

It is noted that regular property cannot be applied by static way, and if so, will result in fatal error. It is noted that inside a class you can access static property by self keyword. If you are accessing parent class property then, apply parent keyword as shown:

```
class testParent
{
    public static $var1;
}

class testChild extends testParent
{
    public static $var2;
    public $abc = 2;
    function testFunction()
    {
        self::$var2 = 3;
        parent::$var1 = 5;
    }
}

echo testChild::$abc; //throw fatal error
```

It is found that static variable or property is the good means to save value of variable in the context of various instances.

**Check your progress 6**

1. Regular property in static way will lead to:
  - a. data error
  - b. runtime error
  - c. fatal error
  - d. all of above

---

## 1.8 Loop through an object's properties

---

PHP provides a way for objects to be defined so it is possible to iterate through a list of items, with, for example a for each statement. By default, all visible properties will be used for the iteration.

**Example Simple Object Iteration**

```
<?php
class MyClass
{
    public $var1 = 'value 5';
    public $var2 = 'value 6';
    public $var3 = 'value 7';
    protected $protected = 'protected var';
    private $private = 'private var';
    function iterateVisible() {
        echo "MyClass::iterateVisible:\n";
        foreach ($this as $key => $value) {
            print "$key => $value\n";
        }
    }
}

$class = new MyClass();
```

OO and file  
handling in PHP

```
foreach($class as $key => $value) {  
    print "$key => $value\n";  
}  
echo "\n";  
$class->iterateVisible();  
?>
```

On running above program:

var1 => value 5

var2 => value 6

var3 => value 7

MyClass::iterateVisible:

var1 => value 5

var2 => value 6

var3 => value 7

protected => protected var

private => private var

In the output, each iterated properties get accessed where iterator interface gets worked out.

**Check your progress 7**

1. The output of the program will be:

```
<?php
$colors = array("white","safron","magenta","brown");
foreach ($colors as $value)
{
    echo "$value <br>";
}
?>
```

a)white

safron

magenta

a. brown

b. white

c. brown

d. error

---

## 1.9 Clone and compare objects

---

Having a copy of object carrying replicated properties is not always required. In a copy constructor an object having GTK window will holds the resource of GTK window once duplicate or new window carrying similar properties exists. An object copy is created by using the clone keyword (which calls the object's `__clone()` method if possible). An object's `__clone()` method cannot be called directly.

```
$copy_of_object = clone $object;
```

Once the object is cloned, PHP will do shallow copy of object's properties which are referenced to other variables as to remain references. Its syntax is:

```
void __clone ( void )
```

After cloning is over and once clone() method is defined, then newly created object's \_\_clone() method gets called which allow required properties for change.

### Example Cloning an object

```
<?php
class SubObject
{
    static $instances = 0;
    public $instance;

    public function __construct() {
        $this->instance = ++self::$instances;
    }

    public function __clone() {
        $this->instance = ++self::$instances;
    }
}

class MyCloneable
{
    public $object1;
    public $object2;

    function __clone()
    {
        // Force a copy of this->object, otherwise
        // it will point to same object.
        $this->object1 = clone $this->object1;
    }
}
```

```
$obj = new MyCloneable();

$obj->object1 = new SubObject();
$obj->object2 = new SubObject();

$obj2 = clone $obj;

print("Original Object:\n");
print_r($obj);

print("Cloned Object:\n");
print_r($obj2);
?>
```

**The above example will output:**

Original Object:

MyCloneable Object

```
(
  [object1] => SubObject Object
    (
      [instance] => 1
    )

  [object2] => SubObject Object
    (
      [instance] => 2
    )
)
```

Cloned Object:

```
MyCloneable Object  
(  
    [object1] => SubObject Object  
    (  
        [instance] => 3  
    )  
  
    [object2] => SubObject Object  
    (  
        [instance] => 2  
    )  
)
```

### Check your progress 8

1. By cloning:
  - a. object itself gets cloned
  - b. object is cloned with its associations
  - c. both a and b
  - d. neither a nor b

---

## 1.10 Inspect an object

---

Inspecting object in good panel or a panel which locates through optional argument panel Name. Its syntax is inspect(object[, panelName]).

Consider an example:

```
class Person  
  
    attr_accessor :name, :age  
  
    def initialize(name, age)  
  
        @name, @age = name, age
```



```
end  
end  
bob= Person.new("Sanjay Mathur", 45)  
p bob
```

Outputs:

```
#<Person:0x1a58d8 @age=45, @name="Sanjay Mathur">
```

### Check your progress 9

1.The syntax of object inspection involves:

- a. alphabets
- b. numbers
- c. characters
- d. all of these

---

## 1.11 Inherit a class

---

Class inheritance is little complicated as inheritance serves as a connection between a child and its parent. To declare one class inherits code from other class, in such case we have to apply keyword.

```
class Parent {  
    // The parent's class code  
}  
class Child extends Parent {  
    // The child can use the parent's class code  
}
```

In the above code we see that child class makes use of non-private methods and a property which tends to inherit from parent class which allows writing code once in parent and can be applied in parent and child classes.

Consider an example:

```
//The parent class
```

OO and file  
handling in PHP

```
class Vehicle {  
    // Private property inside the class  
    private $model;  
    //Public setter method  
    public function setModel($model)  
    {  
        $this->model = $model;  
    }  
    public function hello()  
    {  
        return "horn! I am a <i>" . $this->model . "</i><br />";  
    }  
}  
  
//The child class inherits the code from the parent class  
class SportsVehicle extends Vehicle {  
    //No code in the child class  
}  
  
//Create an instance from the child class  
$sportsVehicle1 = new SportsVehicle();  
// Set the value of the class' property.  
// For this aim, we use a method that we created in the parent  
$sportsVehicle1->setModel('Maruti');  
//Use another method that the child class inherited from the parent class  
echo $sportsVehicle1->hello();
```

Output:

horn! I am Maruti

### Check your progress 10

1. Which class from which child class inherits?

- i) Child class
  - ii) Parent class
  - iii) Super class
  - iv) Base class
- a. (i)
  - b. (ii)
  - c. (iii)
  - d. (ii) and (iv)

---

## 1.12 Use the protected access modifier

---

In PHP, access modifiers are applied to give access rights to PHP classes and their members which define functions and variables in the class.

### PHP Access Control Modifiers

Following are the PHP keywords used as access modifiers:

**Public:**

It is a class or its members having access modifier that are used publicly from anywhere from outside scope of class. It is used where data is available for all public.

**Private**

The class members with this gets access inside class itself which saves members from outside class access with reference of class instance.

**Protected**

It is similar as private, since it allows sub classes to access protected super class members.

**Abstract**

It is applied only for PHP classes and its functions having abstract functions with abstract class.

### Final

It saves sub classes which override super class members having final keyword.

### **Example: Private & Public Access Modifier in PHP**

```
<? class Computer {  
    private $Model = "HP 1150";  
    private    $HardDrive = 800;  
    private        $Ram = 34;  
    public function Specification(){  
        return $this->Model;  
    }  
}  
  
class User extends Computer{  
    public $UserName = "Sanjay";  
    public $UserExperince = "Engineering";  
    public function UserInformation(){  
        echo $this->UserName." has a ";  
        echo $this->Specification();  
    }  
}  
  
$obj = new User;  
echo $obj->UserInformation();  
?>
```

### **Example: Protected Access Modifier in PHP**

```
<?  
  
class Computer {  
    protected $Name = "Wipro Satellite Pro";
```

```

public function NameDisplay1(){
    echo "Accessing inside the Parent Class: ".$this->Name;
}
}

class Display extends Computer{
    public function NameDisplay2(){
        echo "<br>Accessing inside the child Class: ".$this->Name;
    }
}

$obj = new Computer;
echo $obj->NameDisplay1();
$obj = new Display;
echo $obj->NameDisplay2();
?>

```

### Check your progress 11

1. What will be the class relationship shown in the program?

```

class Employee
{
    private int empid;
    private String ename;
    public double getBonus()
    {
        Accounts acc = new Accounts();
        return acc.calculateBonus();
    }
}
*

```

```
class Accounts
{
public double calculateBonus(){//method's code}
}
```

- a. Aggregation
- b. Simple Association
- c. Dependency
- d. Composition

---

## 1.13 Create abstract classes and methods

---

Abstract class is such class which is not completely implemented that mostly applied as base class for other classes in order to take from complete implementation. It is better in interface as it allows adding common elements of implementation which can be shared by subclasses not simply specifying the interface of subclasses. Normally, we look at creation of abstract class with abstract keyword:

```
<?php
```

```
abstract class MyAbstractClass {}
```

Abstract class contains no body, so it is less importance since it fails to define interface for subclasses so as to inherit and serves as legal class.

```
<?php
```

```
abstract class MyAbstractClass {}
```

```
$obj = new MyAbstractClass();
```

```
/*
```

### Output:

Fatal error:

It is noted that an abstract class is partially implemented, so it is fine for it to contain normal methods:

```
<?php
abstract class MyAbstractClass {
    public function doSomething() {
        echo "MyAbstractClass::doSomething()\n";
    }
}
```

In abstract class, you can declare method as abstract that is not implemented but any non-abstract subclass implements it. The body of abstract method:

```
<?php
abstract class MyAbstractClass {
    public abstract function doSomething();
}
```

It is noticed that an abstract class need not require having any abstract methods. Also if the class is not declared as abstract, it has no abstract method as shown:

```
<?php
class MyClass {
    public abstract function doSomething();
}
/*
```

### Output:

If we run the above program, we see that it will generate fatal error:

### Check your progress 12

1. Abstract class:
  - a. should contain abstract method
  - b. need not contain abstract method
  - c. may or may not contain abstract method
  - d. all of above

---

## 1.14 Create final classes and methods

---

### Final Class

A final class is a class that cannot be extended and can be declared by prefixing 'class' keyword along with 'final'.

#### Example:

```
final class BaseClass {  
    public function myMethod() {  
        echo "BaseClass method called";  
    }  
}  
  
//this will cause Compile error  
class DerivedClass extends BaseClass {  
    public function myMethod() {  
        echo "DerivedClass method called";  
    }  
}  
  
$c = new DerivedClass();  
$c->myMethod();
```

We see that BaseClass here is declared as final and will not inherit. In this, DerivedClass tries to extend from BaseClass and compiler compile an error.

### Final Method

It is a method that cannot be overruled. In order to declare method as final, you need to prefix function name with 'final' keyword as shown in example:

```
class BaseClass {  
    final public function myMethod() {  
        echo "BaseClass method called";  
    }  
}
```



```
class DerivedClass extends BaseClass {  
    //this will cause Compile error  
    public function myMethod() {  
        echo "DerivedClass method called";  
    }  
}  
  
$c = new DerivedClass();  
$c->myMethod();
```

In the above example, DerivedClass extends from BaseClass.

### Check your progress 13

1. BaseClass has myMethod() declared as final which can be:
  - a. can be overruled
  - b. cannot be overruled
  - c. given fatal error
  - d. none of above

---

## 1.15 Work with interfaces

---

Interfaces are abstract classes which has no guts and cannot be worked out as they are designed in OOP allowing qualities in classes. Interface classes are defined by using the keyword interface. All the methods in the interface must be public as this is the nature of an interface.

```
// Declare the interface 'iTemplate'  
  
interface iTemplate  
{  
    public function setVariable($name, $var);  
    public function getHtml($template);  
}
```

### Example of class using interface:

```
class dog implements sound{
```

```
function sound() {  
    echo "sa, re, ga, ma ...";  
}  
  
/* the interface methods/functions implements in class */  
  
function sound() { echo "piano has sound ...";}  
function playing() { echo "guitar is playing ...";}  
}  
  
/*
```

It is seen that when a class implements interface, it should explain all methods or functions of interface else php engine will result in error.

```
*/
```

### Check your progress 14

1. The abstract class carries:
  - a. empty concrete method
  - b. working concrete method
  - c. both a and b
  - d. neither a nor b

---

## 1.16 Introducing Namespaces

---

In PHP, there are namespaces which is added to language. There are no two classes having similar name. They are different as if you are using another party library having user as class named, in such case you fail to have your own class. With this, PHP uses namespaces so as to get out of such issue. Consider the simple program describing Simple Namespacing.

```
<?php  
  
namespace Handle;  
  
// app/models/another.php  
  
class add  
{
```

}

In this, add class with small change shows namespace directive. The line namespace Handle will tell PHP about Handle namespace. It says if classes created in such file will be there in 'Handle' namespace. So using Handle class once again.

&lt;?php

// app/routes.php

\$add = new add();

**Example:**

This\Namespace\And\Class\Combination\Is\Silly\But\Works

Example to use namespace

&lt;?php

namespace app\a{

class one{

public static function \_1(){

echo 'a one \_1&lt;br&gt;';

}

}

}

namespace app\b{

class one{

public static function \_2(){

echo 'b one \_2&lt;br&gt;';

}

}

}

namespace app{

echo a\one::\_1();

echo b\one::\_2();

OO and file  
handling in PHP

```
        echo a\two::_1());  
    }  
  
    namespace app\a{  
        class two{  
            public static function _1(){  
                echo 'a two _1<br>';  
            }  
        }  
    }  
}  
  
prints  
  
a one _1  
b one _2  
a two _1
```

### Check your progress 15

1. In PHP two classes will have:
  - a. similar names
  - b. no similar names
  - c. both classes will have single alphabet
  - d. none of above

---

## 1.17 Let Us Sum Up

---

In this unit we have learnt that Object-Oriented Programming is a type of programming which can be modular and reusable web applications. Object oriented programming methodology frames an application which could be any type as web based or windows based applications. In OOP, all will be around objects and class.

Object-oriented programming serves as unique coding style which allows developers to assemble similar work into classes. While defining a class in PHP,

you can add properties in it. It is hard to imagine that to create native class having regular non-typed public PHP properties.

In PHP, static methods and properties is an important feature which is directly accessible without creating object of class. PHP provides a way for objects to be defined so it is possible to iterate through a list of items, with, for example for each statement. An object copy is created by using the clone keyword (which calls the object's `__clone()` method if possible). An object's `__clone()` method cannot be called directly.

---

## 1.18 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-c)

**Check your progress 2**

**Answers:** (1-c)

**Check your progress 3**

**Answers:** (1-c), (2-d)

**Check your progress 4**

**Answers:** (1-b), (2-d)

**Check your progress 5**

**Answers:** (1-d), (2-d)

**Check your progress 6**

**Answers:** (1-c)

**Check your progress 7**

**Answers:** (1-a)

**Check your progress 8**

**Answers:** (1-a)

**Check your progress 9**

**Answers:** (1-d)

**Check your progress 10**

**Answers:** (1-d)

**Check your progress 11**

**Answers:** (1-c)

**Check your progress 12**

**Answers:** (1-b)

**Check your progress 13**

**Answers:** (1-b)

**Check your progress 14**

**Answers:** (1-c)

**Check your progress 15**

**Answers:** (1 –b)

---

## 1.19 Glossary

---

1. **Public** - It is a property or method which can be worked upon from anywhere on the script.
2. **Private** - It is a method which cannot be accessed from everywhere since used by class or object.
3. **Protected** - It is a property which can be used by code in class which is part of.
4. **Abstract** - It is a property that subclasses and not applied directly.

---

## 1.20 Assignment

---

What is Object-oriented programming?

---

## 1.21 Activities

---

What are Namespace in PHP?

---

## 1.22 Case Study

---

How to create Abstract Class in PHP?

---

## 1.23 Further Readings

---

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22
2. "Using PHP from command line", PHI, Dixit, 2009-09-11

---

## UNIT 2: FILE AND DIRECTORY HANDLING

---

### Unit Structure

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 Get a directory listing**
- 2.3 Read and write an entire file**
- 2.4 Read and write part of file**
- 2.5 Read and write CSV data**
- 2.6 Copy, rename and delete file**
- 2.7 File uploading**
- 2.8 Let Us Sum Up**
- 2.9 Answers for Check Your Progress**
- 2.10 Glossary**
- 2.11 Assignment**
- 2.12 Activities**
- 2.13 Case Study**
- 2.14 Further Readings**

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About directory listing.
- About Read and write file.
- About Read and write part of file.

---

### 2.1 Introduction

---

PHP carries full set of directory support functions which describes different functions that will read and operate directories and entries in it. With other



directory or file parts, it performs similar functions as with other languages and can do similar work with more simplified versions.

---

## 2.2 Get a directory listing

---

The directory listing in PHP is in shape of Directory Listing Script which can be configured easily and allows a programmer to simply in uploading of files on web directories that turns into formatted, mobile friendly directory browser. It is seen that with the ease of version 4, there appears new features in PHP directories which are:

- Support to mobile browser.
- Uploading of many files with certain restrictions on allowed file-types.
- Gives access to restricted access to script with the application of password or IP Address.
- Supports formation of new directories and sub-directories.
- Able to upload zip files and can able to extract them directly having an option to delete zip file once extracted.
- Will able to hide certain file types, names or extensions and directories.
- Ability to sort file listings by name, size or last modified date.

### Single Directory Listing

In order to start with single directory listing, consider a simple function which will return list of files, directories and properties from single directory:

OO and file  
handling in PHP

```
<?PHP
function getFileList($dir)
{
    // array to hold return value
    $retval = array();
    // add trailing slash if missing
    if(substr($dir, -1) != "/") $dir .= "/";
    // open pointer to directory and read list of files
    $d = @dir($dir) or die("getFileList: Failed opening directory $dir for reading");
    while(false !== ($entry = $d->read())) {
        // skip hidden files
        if($entry[0] == ".") continue;
        if(is_dir("$dir$entry")) {
            $retval[] = array(
                "name" => "$dir$entry/",
                "type" => filetype("$dir$entry"),
                "size" => 0,
                "lastmod" => filemtime("$dir$entry")
            );
        } elseif(is_readable("$dir$entry")) {
            $retval[] = array(
                "name" => "$dir$entry",
                "type" => mime_content_type("$dir$entry"),
                "size" => filesize("$dir$entry"),
                "lastmod" => filemtime("$dir$entry")
            );
        }
    }
    $d->close();
    return $retval;
}
```

?>

File and  
directory  
handling

You can use this function as follows:

```
<?PHP

// list files in the current directory
$dirlist = getFileList(".");
$dirlist = getFileList("./");

// a subdirectory of the current directory called images
$dirlist = getFileList("images");
$dirlist = getFileList("images/");
$dirlist = getFileList("./images");
$dirlist = getFileList("./images/");

// using an absolute path
$dirlist = getFileList("{$_SERVER[DOCUMENT_ROOT]}/images");
$dirlist = getFileList("$_SERVER[DOCUMENT_ROOT]/images");
```

File and  
directory  
handling

?>

**Example:**

```
<?PHP

$dirlist = getFileList("images");
echo "<pre>".print_r($dirlist)."</pre>";

/* sample output

Array
(
    [0] => Array
        (
```

OO and file  
handling in PHP

```
[name] => images/background0.jpg
[type] => image/jpeg
[size] => 86920
[lastmod] => 1077461701
)
[1] => ...
)
*/
?>
```

### Check your progress 1

1. In PHP, the directory feature will not:
  - a. support mobile browser
  - b. have restrictions on uploading file types
  - c. give access to restricted access
  - d. supports formation of new directories

---

## 2.3 Read and write an entire file

---

In PHP, you can easily read and write full of existing files and can be able to delete the file. We will initially start with reading a file as:

```
$myFile = "sampleFile.txt";
$fh = fopen($project, 'r');
$projectContents = fread($fh, 21);
fclose($fh);
echo $projectContents;
```

To do this, you need to initially set the file with file name which could be anything as \$project which can be set to "sampleFile.txt" as shown. Using link with \$myFileLink variable, set file name and set instruction for the file. Now see for line with carrying contents of file with variable \$projectContents. Now applying fread function, and pass variable link as first parameter followed by

number of characters which you want to read. You see that PHP will read the files in the same way as you read.

Just like reading the file, PHP will be able to write. With the `fread` function having file size that shows number of characters present in the file. Using this in `fread` function will tell PHP to read till the end of the file.

In PHP, writing to file is not hard as compared to reading of file. For this, consider an example shown:

```
$project2 = "testFolder/sampleFile2.txt";  
$projectLink2 = fopen($project2, 'w+') or die("Can't open file.");  
$newContents = "You wrote on me...";  
fwrite($projectLink2, $newContents);  
fclose($projectLink2);
```

It is noted that writing to file will vanish it immediately. On using write function you need to run it on some test content before taking all files out. For this, you need to have write permission and if it is not enabled, in such case server will tell PHP that it cannot write to file. On opening file link, we set variable `$newContents` to hold content that to be written to file. Finally, apply `fwrite` function in order to write `$newContents` to `$projectLink2`.

### Check your progress 2

1. In PHP, `fread` function will instruct to read:
  - a. first two lines
  - b. last two lines
  - c. half of the program
  - d. from top to bottom of file

---

## 2.4 Read and write part of file

---

You can read and write part of the file in PHP. The `fopen()` function in PHP is applied to open a file which needs two arguments that initially shows file name and mode of operation. You can make the change to opened file which will be

closed using `fclose()` function. The `fclose()` function needs file pointer as its argument which returns true when closure succeeds or false if it fails.

### Reading a file

Once a file is opened using `fopen()` function it can be read with a function called `fread()` that makes use of two arguments. These arguments should be file pointer having file length in bytes. It is noted that files length be found using `filesize()` function that takes file name as argument and returns file size expressed in bytes. Consider the steps shown in reading of file in PHP:

- Open a file using `fopen()` function.
- Have file's length using `filesize()` function.
- Read file's content using `fread()` function.
- Close file with `fclose()` function.

Consider an example that contains text file to a variable which shows required text on web page.

```
<html>

<head>

<title>Reading a file using PHP</title>

</head>

<body>

<?php

    $filename = "tmp.txt";
    $file = fopen( $filename, "r" );

    if( $file == false ) {

        echo ( "Error in opening file" );

        exit();

    }

    $filesize = filesize( $filename );
    $filetext = fread( $file, $filesize );
    fclose( $file );
    echo ( "File size : $filesize bytes" );
    echo ( "<pre>$filetext</pre>" );
```

```
?>
</body>
</html>
```

It will produce the following result:

File size : 278 bytes

### Writing a file

In order to write text in existing file with PHP `fwrite()` function, it is seen that this function uses two arguments that specifies file pointer and string of data which is to be written. You can also include third integer argument to specify length of data to write. The impact of third argument is that, it will stop writing after reaching to particular length. Consider an example showing text insertion in file by writing short text heading inside it. After closing this file its existence is confirmed using `file_exists()` function which takes file name as an argument

```
<?php
$filename = "/home/user/guest/newfile.txt";
$file = fopen( $filename, "w" );
if( $file == false ) {
    echo ( "Error in opening new file" );
    exit();
}

fwrite( $file, "This is a simple test\n" );
fclose( $file );
?>

<html>

<head>

<title>Writing a file using PHP</title>

</head>

<body>
```

OO and file  
handling in PHP

```
<?php
    $filename = "newfile.txt";

    $file = fopen( $filename, "r" );

    if( $file == false ) {
        echo ( "Error in opening file" );
        exit();
    }

    $filesize = filesize( $filename );
    $filetext = fread( $file, $filesize );
    fclose( $file );

    echo ( "File size : $filesize bytes" );
    echo ( "$filetext" );
    echo("file name: $filename");
?>
```

</body>

</html>

It will produce the following result:

Error in opening new file

### Check your progress 3

1. The PHP fopen() function uses \_\_\_\_\_arguments to open a file.
  - a. one
  - b. two
  - c. three
  - d. four
2. The fclose() function needs \_\_\_\_\_as its argument.
  - a. arrays
  - b. strings
  - c. file pointer
  - d. none of above



## 2.5 Read and write CSV data

CSV data in PHP is Comma Separated Values that are file that stores tabular data in plain text form with sequence of characters having no data that has to be interpreted as binary numbers. It carries any number of records, with line breaks having fields, separated by character or string with literal comma or tab. In this, the records have identical sequence of fields. The structure of CSV file:

one, two

example1, example2

data1, data2

test1, test2

You can read from a CSV file using php function "fgetcsv" which reads content of csv file as shown:

```
<?php
```

```
function readCSV($csvFile){
    $file_handle = fopen($csvFile, 'r');
    while (!feof($file_handle) ) {
        $line_of_text[] = fgetcsv($file_handle, 1024);
    }
    fclose($file_handle);
    return $line_of_text;
}
```

```
// Set path to CSV file
$csvFile = 'test_data.csv';
//calling the function
$csv = readCSV($csvFile);
if(!empty($csv)){
    foreach($csv as $file){
        //inserting into database
```

OO and file  
handling in PHP

```
$query_insert = "insert into csv_data_upload set
    name = ".$file[0].",
    value = ".$file[1]."";

echo $query_insert;

$insert = mysql_query($query_insert);
}
}else{
    echo 'Csv is empty';
}
?>
```

Once the code is executed, data gets stored in database.

To write into CSV File we require "fputcsv" which will format a line as csv and write it to particular file handle.

### Example

```
<?php

$list = array (
    array('dd', 'mm', 'yy','),
    array('12', '12', '2016'),
    array(""dd"", ""mm"")
);

$fp = fopen('file.csv', 'w');
foreach ($list as $fields) {
    fputcsv($fp, $fields);
}

fclose($fp);

?>
```

Following code will write into CSV File.

**Check your progress 4**

1. In PHP, fputcsv fuinction helps:
  - a. To read CSV
  - b. To write CSV file
  - c. To print CSV file
  - d. None of above

---

## 2.6 Copy, rename and delete file

---

### PHP copying a file

Simply, to copy a file, use copy() function which initially requires file type to copy that passes file path to first parameter of copy() function. Apart from this, you need to show file path so as to copy file. It is seen that copy() function will return true when file gets copied completely, else will returns false. In the following program, there are specific codes written which shows how to copy the test.txt file to test2.txt file.

```
<?php
```

```
    $fn = './test.txt';
```

```
    $newfn = './test2.txt';
```

```
    if(copy($fn,$newfn))
```

```
        echo 'The file was copied successfully';
```

```
    else
```

```
        echo 'An error occurred during copying the file';
```

```
<?php
```

```
    $fn = './test.txt';
```

```
    $newfn = './test2.txt';
```

```
    if(copy($fn,$newfn))
```

```
        echo 'The file was copied successfully';
```

```
    else
```

```
echo 'An error occurred during copying the file';
```

### PHP renaming a file

You can rename a file using rename() function which makes you to move a file from one directory to different directories. Consider the program where renaming of test.txt file to test.bak file exists with code:

```
<?php
$fn = './test.txt';
$newfn = './test.bak';
if(rename($fn,$newfn)){
    echo sprintf("%s was renamed to %s",$fn,$newfn);
}else{
    echo 'An error occurred during renaming the file';
}
<?php
$fn = './test.txt';
$newfn = './test.bak';
if(rename($fn,$newfn)){
    echo sprintf("%s was renamed to %s",$fn,$newfn);
}else{
    echo 'An error occurred during renaming the file';
}
```

We see that rename() function will return true when file gets successfully renamed, else returns false. In the following program, it describes how to apply rename() function so as to move test.bak to backup directory:

```
<?php
$fn = './test.bak';
$newfn = './backup/test.bak';
if(rename($fn,$newfn)){
    echo sprintf("%s was moved to %s",$fn,$newfn);
}else{
    echo 'An error occurred during moving the file';
}
```

```
}  
  
<?php  
$fn = './test.bak';  
$newfn = './backup/test.bak';  
if(rename($fn,$newfn)){  
    echo sprintf("%s was moved to %s", $fn,$newfn);  
}else{  
    echo 'An error occurred during moving the file';  
}
```

### PHP deleting a file

Similarly you can delete a file using unlink() function where on applying the particular file will result in deletion using unlink() function. Consider the program showing use of unlink() function in file deletion.

```
<?php  
$fn = './backup/test.bak';  
if(unlink($fn)){  
    echo sprintf("The file %s deleted successfully", $fn);  
}else{  
    echo sprintf("An error occurred deleting the file %s", $fn);  
}
```

```
<?php  
$fn = './backup/test.bak';  
if(unlink($fn)){  
    echo sprintf("The file %s deleted successfully", $fn);  
}else{  
    echo sprintf("An error occurred deleting the file %s", $fn);  
}
```

We see that copy(), rename() and unlink() functions raise warning-level errors if the file cannot be found therefore it is good practice to check the file exists using the file\_exists() function before copying, renaming or deleting it.

### **Check your progress 5**

1. The unlink() functions is used in:
  - a. read of file
  - b. writing of file
  - c. deleting of file
  - d. none of above

---

## **2.7 File uploading**

---

It is seen that normally files get uploaded in temporary directory which further will send across targeted destination with the help of PHP script. Information in phpinfo.php page will show temporary directory which is required for file uploading using upload\_tmp\_dir with max allowed size of files as upload\_max\_filesize. Such parameter gets set in PHP configuration file php.ini.

The process of uploading requires following steps –

- On opening HTML form having text files along with browse button and submit button.
- User on clicking browse button and choosing uploading of file from local PC.
- Complete path along with selected file will appear in text filed which the user will click on submit button.
- On pressing submit, selected file gets sent to temporary directory on server.
- The PHP script specified as form handler in form's action will check for arriving of file which further copies file in intended directory.
- The PHP script further will make sure about success to user.

### **Creating Upload File Script**

The "upload.php" file contains the code for uploading a file:

```
<?php
$target_dir = "uploads/";
$target_file = $target_dir . basename($_FILES["fileToUpload"]["name"]);
$uploadOk = 1;
$imageFileType = pathinfo($target_file,PATHINFO_EXTENSION);
// Check if image file is a actual image or fake image
if(isset($_POST["submit"])) {
    $check = getimagesize($_FILES["fileToUpload"]["tmp_name"]);
    if($check !== false) {
        echo "File is an image - " . $check["mime"] . ".";
        $uploadOk = 1;
    } else {
        echo "File is not an image.";
        $uploadOk = 0;
    }
}
?>
```

In this, we see that:

\$target\_dir = "uploads/": This will specify directory about the place of file

\$target\_file: It shows path of file which to be uploaded

\$imageFileType: It will keep file extension of file

### Check your progress 6

1. The correct form of uploading files in PHP is:

- a. upload.php
- b. upl.php
- c. uploa.php
- d. upload.php/

---

## 2.8 Let Us Sum Up

---

In this unit we have learnt that PHP carries full set of directory support functions which describes different functions that will read and operate directories and entries in it. In PHP, `fopen()` function is used to open a file. It requires two arguments stating first the file name and then mode in which to operate. After making a changes to the opened file it is important to close it with the `fclose()` function.

CSV data in PHP is Comma Separated Values that are file that stores tabular data in plain text form with sequence of characters having no data that has to be interpreted as binary numbers.

---

## 2.9 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-b)

**Check your progress 2**

**Answers:** (1-d)

**Check your progress 3**

**Answers:** (1-b), (2-c)

**Check your progress 4**

**Answers:** (1-b)

**Check your progress 5**

**Answers:** (1-c)

**Check your progress 6**

**Answers:** (1-a)



---

## 2.10 Glossary

---

1. **Class** - HTML elements can have one or more classes, separated by spaces. You can style elements using CSS by selecting them with their classes.
2. **Usage** - Almost all content belongs inside the body tag. The main exceptions are script and style tags, as well as the page title tag.
3. **Div** - A block level container for content with no semantic meaning.

---

## 2.11 Assignment

---

Explain the CSV data in PHP.

---

## 2.12 Activities

---

Write a program stated the application of PHP fopen() function.

---

## 2.13 Case Study

---

Compile and run the program in PHP which will upload files from computer?

---

## 2.14 Further Readings

---

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22.
2. "Using PHP from command line", PHI, Dixit, 2009-09-11.

---

## Block Summary

---

In this block, you will understand about the basic of Use the protected access modifier with knowledge about creating abstract classes and methods. The block gives an idea on creating final classes and methods along with concept of Interface of class are well detailed. The examples related to concept of Namespaces are also discussed.

In this block, you will understand about the basic of Copy, rename and deleting file along with its techniques. The concept related to reading and writing complete or part of file in PHP is well detailed. The student will be demonstrated practically about Comma Separated Values.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. What is directory listing in PHp?
2. What is object's \_\_clone() method?
3. What are the advantages and drawbacks of directories in PHP?
4. What is declaration of object in PHP?
5. Explain attribute with example?

### **Long Answer Questions**

1. Explain the steps involved in reading and writing part of file in PHp?
2. What are the features of CSV data files in PHP?
3. Explain Copy, rename and delete file in PHP?

Enrolment No.

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                   | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|-----------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                    | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Language and Style                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Illustration used (Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Conceptual Clarity                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Check your progress Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Feed back to CYP Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# Fundamentals of Computer Networking (FCN)

PGDCA 201

## BLOCK 2: CREATING WIRED AND WI-FI LAN

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**



# Fundamentals of Computer Networking (FCN)

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Mr. Sanjay Thapar

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Mr. Prashant Tikone

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'





## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# FUNDAMENTALS OF COMPUTER NETWORKING (FCN)

## **Contents**

### **BLOCK 1: NETWORKING CONCEPT**

#### **UNIT 1 INTRODUCTION AND NETWORKING BASICS**

Advantages of computer networking, computer networks and the Internet, WAN, LAN and PAN basics, Topologies, Connecting Media: Wired and Wireless and their characteristics, Introduction to NIDs and their specifications

#### **UNIT 2 NETWORK INTERFACE DEVICES**

Network Adaptor Cards (both wired and wireless), Hubs, Switches, Routers, Access Points (Wireless), Repeaters. Their basic architecture, working and use/application, understanding their technical specifications/data sheets.

---

### **BLOCK 2: CREATING WIRED AND WI-FI LAN**

#### **UNIT 1 CREATING A SWITCHED WIRED ETHERNET LAN**

Introduction to UTP CAT series cables, RJ-45 connectors, color coding scheme, crimping a UTP cable to RJ-45 connector, physically connecting individual nodes to the switch, selection of server machine, Windows 8.1 Server Installation and Configuration on Server Machine, Windows 8.1 Desktop installation and configuration on client nodes, checking connectivity, basic troubleshooting/diagnostic commands.

#### **UNIT 2 CREATING A WI-FI LAN:**

Introduction to Wi-Fi Technology, how to provide Wi-Fi capability to a PC, creating an ad-hoc Wi-Fi based LAN, creating an infrastructure based LAN using Wireless AP, configuration of AP and client Machines, accessing data from File Server through Wi-Fi Interface from client machine.

---



### **BLOCK 3: ADSL BROADBAND INTERNET AND WI-FI USB DONGLES**

#### **UNIT 1      ADSL BROADBAND INTERNET**

Introduction to ADSL broadband technology, motivation for ADSL Broadband, PSTN Basics, ADSL Modem basic architecture, working, standards, ADSL Wi-Fi Modem and Router, configuring a wired ADSL Modem for Internet Access, configuring a Wi-Fi ADSL modem/Router for Internet Access

#### **UNIT 2      WI-FI USB DONGLES**

Motivation and Need for Wi-Fi Dongles, basic architecture and working, connecting and configuring a Wi-Fi Dongle with a PC.

### **BLOCK 4:      INTERNET ACCESSING AND APPLICATION**

#### **UNIT 1      TETHERING FOR INTERNET ACCESS**

Need and Motivation for Tethering, Tethering with Wi-Fi, Tethering with Bluetooth, Tethering with USB Cable, Reverse Tethering

#### **UNIT 2      INTERNET/LAN APPLICATIONS**

Popular Browsers like Internet Explorer and Chrome, their configuration and settings, FileZilla File Transfer software, Team Viewer, Remote Desktop, Telnet, Microsoft Outlook Express.



Dr. Babasaheb  
Ambedkar  
Open University

PGDCA 201

## **Fundamentals of Computer Networking (FCN)**

### **BLOCK 2: CREATING WIRED AND WI-FI LAN**

---

#### **UNIT 1**

Creating a Switched Wired Ethernet LAN

02

#### **UNIT 2**

Creating a Wi-Fi LAN

26

---

# **BLOCK 2: CREATING WIRED AND WI-FI LAN**

---

## **Block Introduction**

The Ethernet cables are number sequence categories which support various specifications which are updated with certain testing standards. There are many types of cables which are used for various purposes. Category 5 cables was revised, and mostly replaced with, category 5 enhanced (Cat-5e) cables which did not change anything physically within the cable, but instead applied more demanding testing standards for crosstalk.

In this block, we will detail about the basic of RJ connector and its working techniques. The block will focus on architecture and distribution of cross-over cable applied to connect router to computer or ethernet switch along with their characteristics. The concept of Wi-Fi networks with infrastructure mode and working characteristics are also explained.

In this block, you will make to learn and understand about the basic of ad-hoc network and its techniques. The concept related to Wi-Fi networks and its working features are explained that will help you in learning more about networks. You will be demonstrated practically about various types of USB connectors.

## **Block Objective**

**After learning this block, you will be able to understand:**

- The basic of UTP CAT cables.
- Features of RJ-45 connectors.
- Idea about crimping of UTP cable with RJ-45 connector.
- Features of server machine.
- Characteristics of Windows 8.1 Server Installation and Configuration.
- Features about Wi-Fi Technology.
- Characteristics about Wi-Fi capability to PC.
- Configuration of AP and client Machines.

Creating Wired and  
Wi-Fi LAN

## **Block Structure**

**Unit 1:**     **Creating a Switched Wired Ethernet LAN**

**Unit 2:**     **Creating a Wi-Fi LAN**



---

# **UNIT 1: CREATING A SWITCHED WIRED ETHERNET LAN**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction to UTP CAT series cables**
- 1.2 RJ-45 connectors**
- 1.3 Color coding scheme**
- 1.4 Crimping a UTP cable to RJ-45 connector**
- 1.5 Physically connecting individual nodes to the switch**
- 1.6 Selection of server machine**
- 1.7 Windows 8.1 Server Installation and Configuration on Server Machine**
- 1.8 Windows 8.1 Desktop installation and configuration on client nodes**
- 1.9 Checking connectivity**
- 1.10 Basic troubleshooting/diagnostic commands**
- 1.11 Let Us Sum Up**
- 1.12 Answers for Check Your Progress**
- 1.13 Glossary**
- 1.14 Assignment**
- 1.15 Activities**
- 1.16 Case Study**
- 1.17 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About UTP CAT series cables
- About crimping UTP cable
- About server machine

- About Windows 8.1 Server

---

## 1.1 Introduction to UTP CAT series cables

---

Ethernet cables are grouped in sequence numbered categories supported different specifications; typically the category is updated with further clarification or testing standards. These categories are how we can simply know what type of cable we need for a particular application.

It is noticed that Category 5 cable was revised and upgraded with category 5 enhanced (Cat-5e) cables with similar physical look inside and outside the cable, with required testing standards.

Category 6 was revised with greater than before category 6 (Cat-6a) which shows testing for 500 MHz communication. It is noted that higher communication frequency will eliminate alien crosstalk (AXT) that allow for longer range upto 10 Gb/s.

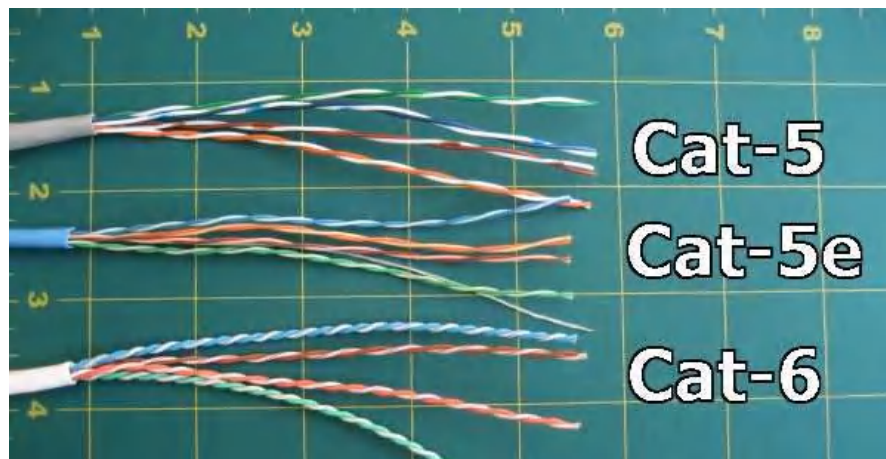


**Fig 1.1 Types of CAT cable**

It is found that certain Cat-6 cables will carry out nylon spline that will help to remove crosstalk. It is noted that mostly spline is not required in Cat-5 cable. In Cat-6 cable, spline is not required either the cable is long or short and as per standard. Fig 1.1 shows Cat-5e cable with spline.

Generally, Ethernet cables are twisted pair where manufactures uses shielding so as to safeguard from interference, while unshielded twisted pair also applied for cables among computer and wall which uses shielding cable for areas having high interference and running cables outdoors or inside walls.



**Fig 1.2 UTP and STP Cable**

There are different ways to shield an Ethernet cable, but typically it involves putting a shield around each pair of wire in the cable. This protects the pairs from crosstalk internally. It is found that manufactures protect the cables from foreign identities simply by screening UTP or STP cables.

---

## 1.2 RJ-45 connectors

---

RJ connector is registered jack (RJ) standardized physical network interface that connects telecommunications or data equipment. RJ45 could be a standard type of connector for network cables. RJ45 connectors are most commonly seen with Ethernet cables and networks. It's an eight pins to which the wire strands of a cable interface electrically. Standard RJ-45 pin outs define the arrangement of the individual wires needed when attaching connectors to a cable.

Several other kinds of connectors closely resemble RJ45 and might be easily confused for each other. The RJ-11 connectors used with telephone cables, for example, are only slightly smaller (narrower) than RJ-45 connectors.



**Fig 1.3 RJ Connector**

It is noted that connectors with required jacks are mostly modular connector which can be 50-pin miniature ribbon connector type. Such types are most common twisted pair connector types which is 8-position, 8-contact (8P8C) modular plug and jack called as RJ45 connector.

### **Check your progress 1**

1. Which of the following connector is used with telephone cable?
  - a. RJ45
  - b. Ethernet
  - c. RJ 11
  - d. None of these

---

## **1.3 Color coding scheme**

---

RJ-45 conductor data cable consists of 4 pairs of wires each consists of a solid colored wire and a strip of the same colour. There are two wiring standards for RJ-45 wiring:

- T-568A
- T-568B

There are 4 pairs of wires; 10BaseT/100BaseT Ethernet uses only 2 pairs:

- Orange
- Green

Apart from this, other two colors are:

- Blue
- Brown

This is used for second Ethernet line or for phone connections. The two wiring standards are used to create a cross-over cable:

- T-568A on one end
- T-568B on other end

Straight-through cable:

- T-568B
- T-568A

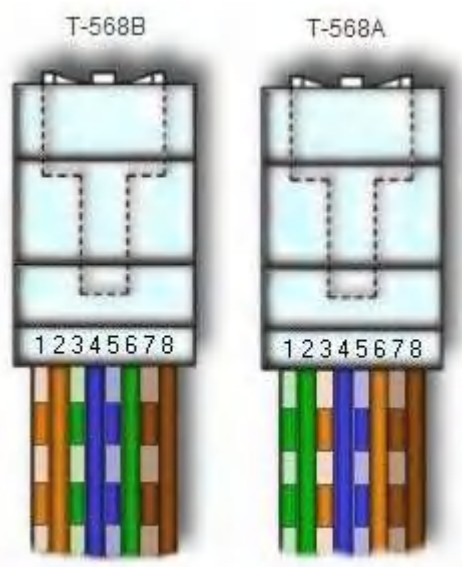


**Fig 1.4 Cable T568B and T568A**

The RJ45 data cable is used to connect computers to Ethernet switch by straight-through cables. The RJ45 cable uses only 2-pairs of wires:

- Orange (pins 1 & 2)
- Green (pins 3 & 6)
- Pins 4, 5 (Blue) and 7, 8 (Brown) are NOT used

Straight-through cable connects:

- pin 1 to pin 1
- pin 2 to pin 2
- pin 3 to pin 3
- pin 6 to pin 6

Cross-over cables are used to connect:

- TX+ to RX+
- TX- to RX-

That connects pin 1 to pin 3, pin 2 to pin 6, pin 3 to pin 1 and pin 6 to pin 2. The unused pins are generally connected straight-through in both straight-through and cross-over cables.

To network two computers without a hub, a cross-over cable is used. Cross-over cable is also used to connect a router to a computer, or Ethernet switch (hub) to another Ethernet switch without an uplink. Most Ethernet switches today provide an uplink port, which prevents a use of cross-over cable to daisy chain

another Ethernet switch. Straight-through cables are used to connect a computer to an Ethernet switch, or a router to an Ethernet switch.

It is noted that RJ45 cables carries 8 color coded wires, and the plugs have 8 pins and conductors. In this eight wires are used as 4 pairs, each representing positive and negative polarity.

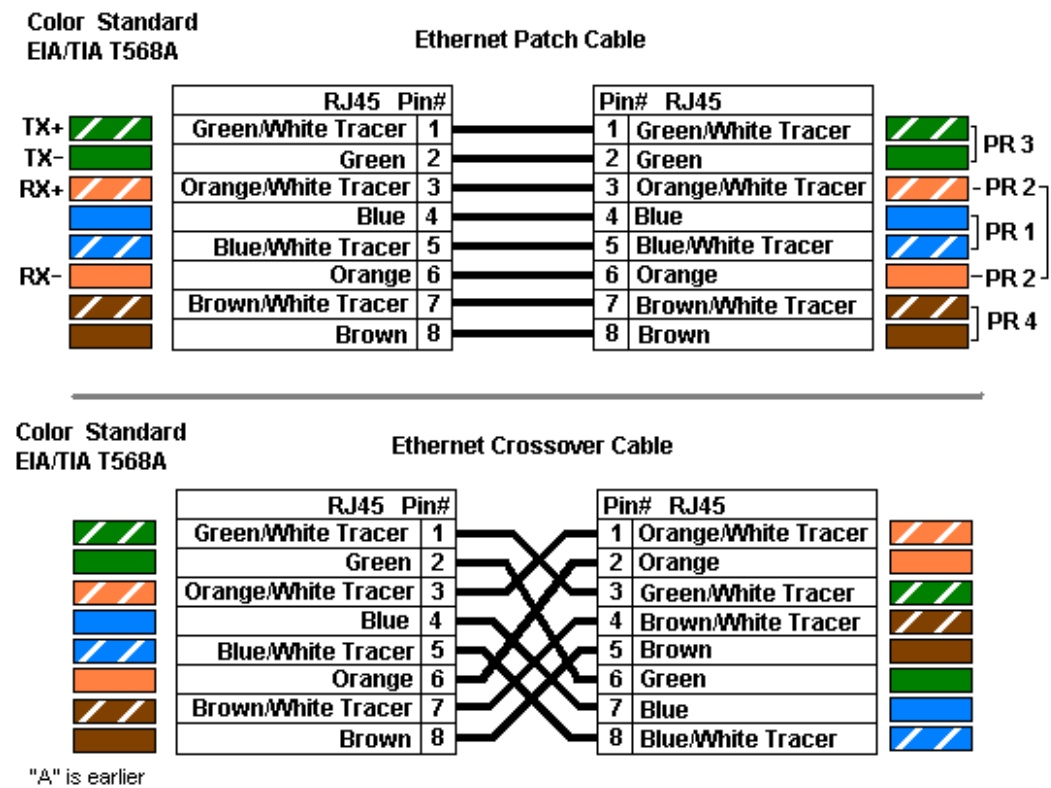


Fig 1.5 Colour Standard of T568A

Color Standard  
EIA/TIA T568B

Ethernet Patch Cable



Color Standard  
EIA/TIA T568B

Ethernet Crossover Cable

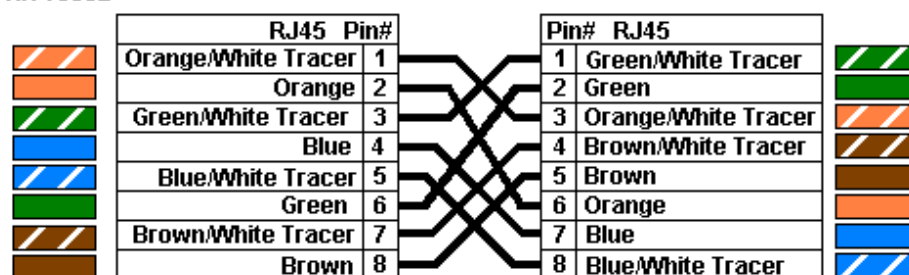


Fig 1.6 Colour Standard of T568B

The most commonly used wiring standard for 100baseT is T-586B standards. Prior to EIA 568A and 568B standards, the color-coded scheme was used to wire RJ45 cables. The table below shows pin and color schemes used in various setup.

| Pin | Colored Scheme  | T-568B (Common) | T-568A        |
|-----|-----------------|-----------------|---------------|
| 1   | Blue            | Orange Stripe   | Green Stripe  |
| 2   | Orange          | Orange          | Green         |
| 3   | Black           | Green Stripe    | Orange Stripe |
| 4   | Red             | Blue            | Blue          |
| 5   | Green           | Blue Stripe     | Blue Stripe   |
| 6   | Yellow          | Green           | Orange        |
| 7   | Brown           | Brown Stripe    | Brown Stripe  |
| 8   | White (or Grey) | Brown           | Brown         |

Table 1.1 Pin and Colour Scheme of RJ45 Connector

## Check your progress 2

1. RJ-45 conductor data cable consists of \_\_\_\_\_ of wires
  - a. 4 pairs
  - b. 2 pairs
  - c. 3 pairs
  - d. 6 pairs

---

## 1.4 Crimping a UTP cable to RJ-45 connector

---

A UTP cable is one of the most popular LAN cables which consist of 4 twisted pairs of metal wires. Adding RJ45 connectors at both the ends of the UTP cable will allow the cable to work in LAN network system. There are some steps that to be followed to crimp UTP cable into RJ 45 connector:

Step 1: Initially cut the plastic sheath about 1 inch from end of cut cable with the use of razor blade located in crimping tool.

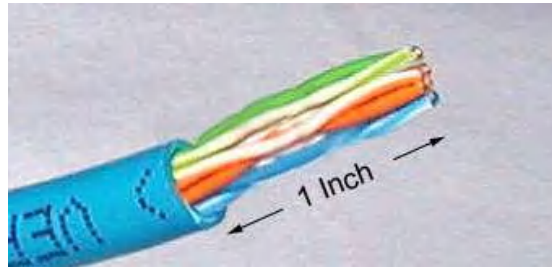


**Fig 1.7 Cutting of Cable sheath**

Step 2: Now unwind and pair the similar colors as shown in fig.

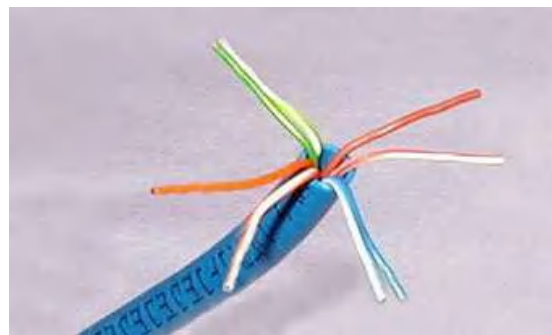


**Fig 1.8 Unwind the coloured wires**

Step 3: Keep the wires in between the fingers and make the wires straight with required colour orders.



**Fig 1.9 Straighten wires with fingers**



Step 4: Use scissors to make a straight cut across 8 wires to make them short by 1/2 Inch from cut sleeve to wire end.



**Fig 1.10 Scissor**

Step 5: Now insert 8 colour wires in RJ 45 connector by considering proper position of blue plastic sleeve.



**Fig 1.11 Wires in RJ 45 Connector**

Step 6: Now crimp the cable carefully by putting RJ 45 connector in Ethernet Crimper and push on handles tightly. You will find that copper splicing tabs on connector will cut into each of eight wires.



**Fig 1.12 Crimping of wire**

After crimping UTP cable in RJ 45 connector, you find the cable as shown in fig 1.13.

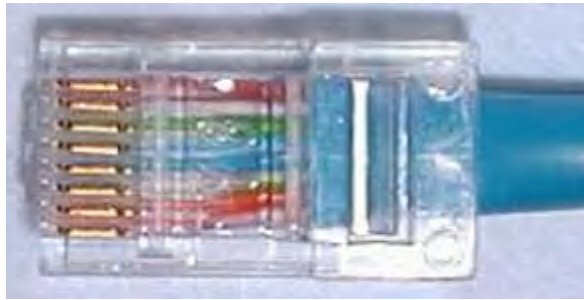


**Fig 1.13 Cable in RJ 45 Connector**

### **Check your progress 3**

1. Full name of UTP is\_\_\_\_\_.
- a. Universal Transport Port
- b. Unshielded Transport port
- c. Unshielded Twisted Pair
- d. None of these

---

## **1.5 Physically connecting individual nodes to the switch**

---

Switches are often a valuable asset to networking. Overall, they can increase the capacity and speed of your network. However, switching shouldn't be seen as a cure-all for network problems.

Switches occupy the same place in the network as hubs. Unlike hubs, switches examine each packet and process it accordingly rather than simply repeating the signal to all ports. Switches map the Ethernet addresses of the nodes residing on each network segment then allow only the required traffic to pass through the switch. Once a packet is received by the switch, the switch examines the destination and source hardware addresses and compares them to a table of network segments and addresses.

.

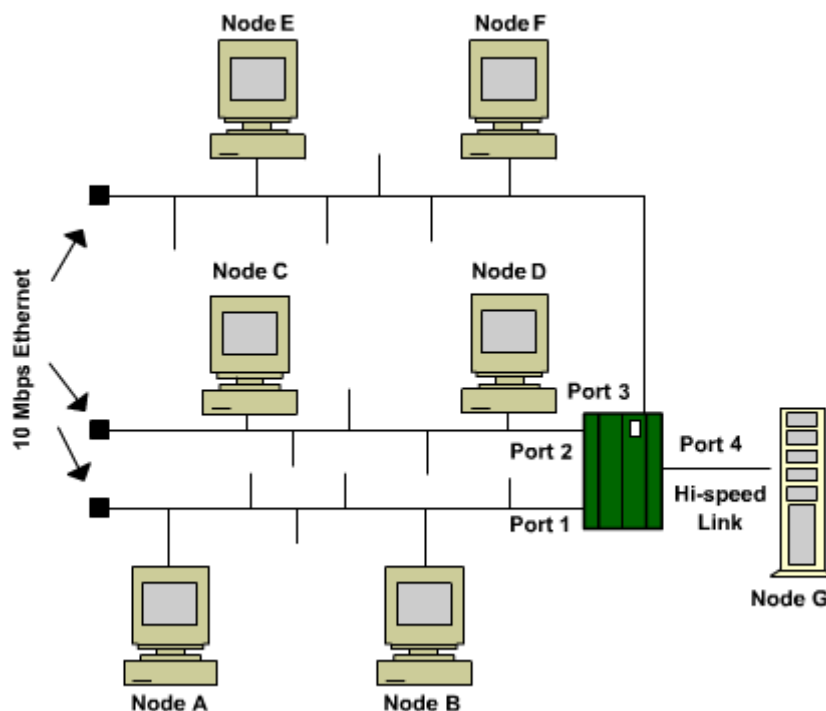


**Fig 1.14 Nodes in network**

Just like bridge, a switch does more of its job in high-speed hardware by providing performance closer to single-LAN performance than bridged-LAN performance.

Also unlike a bridge, which shares the LAN bandwidth among all of its ports, a switch dedicates the entire LAN media bandwidth, such as 10-Mbps Ethernet, to each port-to-port frame transmission. In this way, a switch easily multiplies the amount of effective network bandwidth.

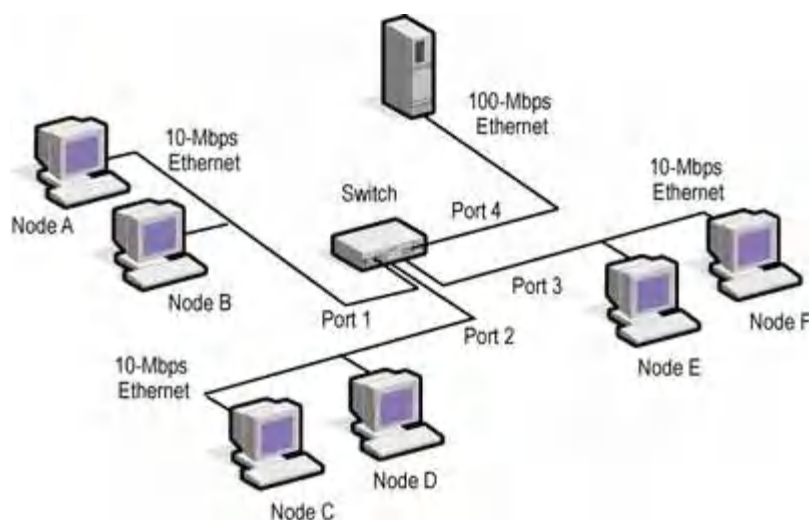


**Fig 1.15 Connecting nodes with switches**

### **Check your progress 4**

1. Switches operates at\_\_\_\_\_.
- a. Network layer
- b. Data Link layer
- c. Transport layer
- d. None of these

---

## **1.6 Selection of server machine**

---

It is found that small server look not different from high end desktop computer which is designed for easy operating system which can run certain desktop applications such as:

- word processor
- spreadsheet
- email client
- web browser

A server runs a specialised operating system designed to support several users. It's engineered to run multiuser applications like email, messaging, and print servers; shared calendar programs; databases; and enterprise resource planning and customer relationship management software.

A server additionally makes it easy for your employees to share information and collaborate, since it operates as a central repository for all of your documents, images, contacts, and other necessary files. It can host a company intranet, for sharing information with your employees quickly and economically. Set up a virtual private network, and you and your employees will access the data on the server remotely from anywhere you have internet access. On top of that, a server can automatically back up your desktop and laptop systems; therefore you'll never lose critical data if one machine fails or is lost or stolen. Servers are designed to be reliable, secure, and fault-tolerant, with redundant storage options.

When selecting correct server it is noted that it depends on large measure on applications which you want to work on. In case of file sharing, automated client backup, and light-duty remote access for PCs, we need to think for NAS or Windows Home Server. In a business having more than 10 employees are using

computers, then if you wish to use an email or print server, or you want to handle complex database, or run sophisticated server-based applications, or you have to store large information, or large-scale virtualization, in such cases there are more option such as tower, rack or blade server.

Virtualization enables one server to behave as several servers, each with its own operating system and unique set of applications. A virtual machine consists solely of software, yet it has all the components of a physical machine: it's a motherboard, a CPU, a hard disk, a network controller, and so on. The operating system and other applications run on a virtual machine just as they would on a physical machine--they see no difference between the 2 environments.

In virtualization, a program referred to as a hypervisor places an abstraction layer between the operating systems and therefore the hardware. The hypervisor will operate multiple virtual machines with the same OS or different OSs on the same physical server. Microsoft, Oracle, and VMware are among the top virtual-machine developers.

### **Check your progress 5**

1. A server runs a specialised \_\_\_\_\_ system designed to support several users.
  - a. Operating
  - b. Hardware
  - c. OS and Software
  - d. None of these

---

## **1.7 Windows 8.1 Server Installation and Configuration on Server Machine**

---

Windows 8.1 is an upgraded version of Windows 8 which is also computer operating system by Microsoft. It is part of Windows 8's support lifecycle and on installation maintains access to support. According to Microsoft, the following actions need to be performed before the installation of Windows Server:

Unplug of UPS system: Any power backup system needs to be unplugged or removed before installing Windows Server 8.1.

**Back up data:** It is highly recommended that you should take complete backup of configuration information for servers which include booting and system partitions.

**Running of Windows Memory Diagnostic tool:** Such type of testing procedure will tests the capacity and features of computer's RAM.

**Use of mass storage drivers:** You need to save driver file to appropriate media so that you can provide it during setup.

**Default action of Windows Firewall:** Server applications that require inbound connections will fail until you create inbound firewall rules that allow these connections.

**Prepare your Active Directory environment for Windows Server 2012 R2:** Before adding a Windows Server 2012 R2 domain controller or updating an existing domain controller to Windows Server 2012 R2, prepare the domain and forest by running Adprep.exe.

### **Check your progress 6**

1. Why backing up of data is required?
  - a. To save configuration information
  - b. To save System data
  - c. To be secure from data loss in case of system failure
  - d. All of these

---

## **1.8 Windows 8.1 Desktop installation and configuration on client nodes**

---

Client deployment refers to the planning, installation, and management of System Center 2012 Configuration Manager Client computers and mobile devices in your enterprise. The types of devices that you have, your business requirements, and your preferences, determine the methods that you use to manage computers and mobile devices.

For installation of Windows 8.1 Desktop version, the table 1.1 shows the required parameters:

| Component            | Minimum Requirement         | Microsoft Recommended  |
|----------------------|-----------------------------|------------------------|
| Processor            | 1.4 GHz                     | 2 GHz or faster        |
| Memory               | 512 MB RAM                  | 2 GB RAM or greater    |
| Available Disk Space | 32 GB                       | 40 GB or greater       |
| Optical Drive        | DVD-ROM drive               | DVD-ROM drive          |
| Display              | Super VGA (800x600) monitor | XGA (1024x768) monitor |

**Table 1.1 Parameters required for installation of Windows**

### Check your progress 7

1. What is meant by client deployment?
  - a. delivering software to client
  - b. user testing
  - c. It is the planning, installation, and management of System
  - d. All of these

## 1.9 Checking connectivity

The next target is to check the connectivity of modem or routers which can be configured initially so as to use the Internet while preventing other services from Internet. If you are not confident of your network whether it has configured or not, in such case you have to contact network administrator. If your network is not configured to block services, but some Internet applications work and others do not, then the issue is probably not related to your Wi-Fi network. There are certain steps you need to follow:

Step 1: Make sure that Wi-Fi device is ON. Computer carry inbuilt card that gets ON when turning it ON for use. When the Wi-Fi interface is ON and connected to Wi-Fi network, then Wi-Fi menu gets bold which is at top right corner of screen.

When the menu becomes dark, in such case your computer gets connected to Wi-Fi network.

Step 2: If Wi-Fi is off, choose Turn Wi-Fi On from the menu.

When you do find menu bar, then in such case select System Preferences from Apple menu and click Network icon and after that select Wi-Fi. Now click on check box located next to Show Wi-Fi status in menu bar as shown:



**Fig 1.16 Checking connectivity**

If your Wi-Fi interface does not appear in System Preferences, then you'll need to make assure that Wi-Fi card gets identified by computer. In such case load using CD or from Recovery HD if OS X Lion is installed. Your computer should be able to access available networks.

Step 3: After all when you are unable to get it online, then in such case make sure that your computer has connected with right Wi-Fi network as per the listing from the Wi-Fi menu as shown in figure:





**Fig 1.17 Setting of Wi-Fi**

Select your network if it is not chosen. If your Wi-Fi network is secured using a password, you will be prompted to enter a password as shown below.



**Fig 1.18 Using Password**

In this, you have to insert password. If you do not know your network password, then in such case, you have to contact the administrator of Wi-Fi network. If you are administrator/owner of network, you should configure router to define password for network.

Your Wi-Fi network may not be visible in the list. If the network is closed, it will not broadcast its network name. In order to join the Wi-Fi network, choose Join Other Network from the Wi-Fi menu. You will be prompted for the network name and security setting.



**Fig 1.19 Network name**

In this, enter the name of network and select Security of your network uses.

When the network is not seen in your Wi-Fi network list, then it means that such network is not applicable. To check the standards, you have use Network Utility, where you should set network interface to Wi-Fi and study about information listed after Model:

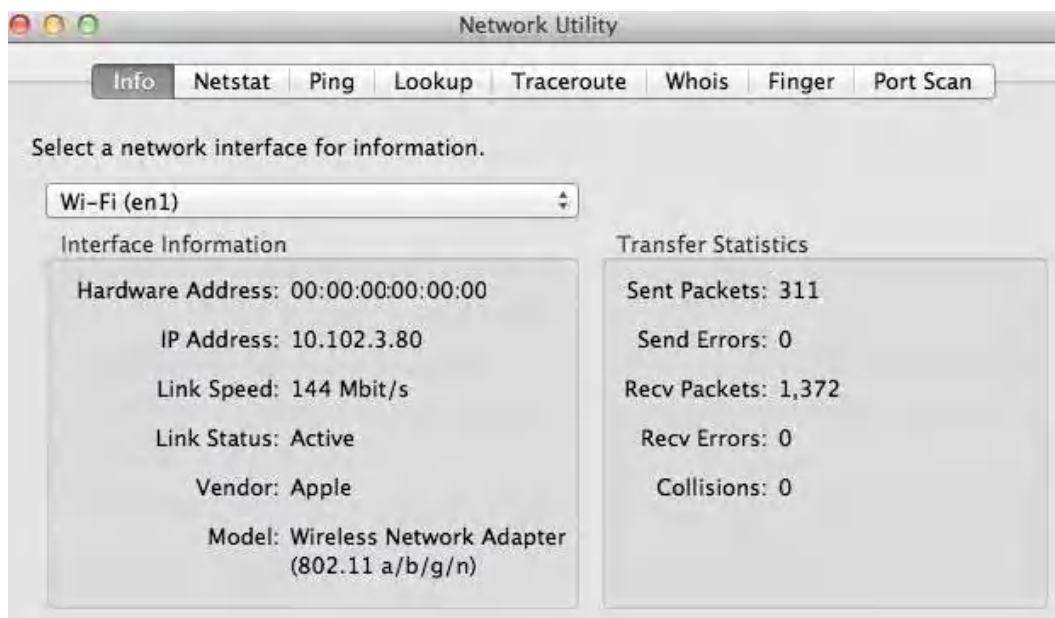


**Fig 1.20 Network Utility**

In case of correct Wi-Fi network, but still can't find online, then you need to check TCP/IP settings which is available in Network pane of System Preferences.

- Choose System Preferences from the Apple menu.
- Choose Network from the View menu.

- Select Wi-Fi, and then click the advanced button in the lower-left hand corner of the screen.
- Select the TCP/IP tab from the top of the screen.

After doing all setting your window will look like:



**Fig 1.20 Network TCP/IP**

If you see no IPv4 address, or if IP address starts "169.254.xxx.xxx", in such case you need to click on Renew DHCP Lease. Without the correct TCP/IP settings, your computer will not be able to get online.

If your TCP/IP settings appear to be correct, and your computer still cannot access the Internet, check the DNS tab. See below for this tab. DNS is an Internet service that translates IP addresses into URLs and vice-versa. A correct DNS configuration allows your computer to connect to [www.apple.com](http://www.apple.com) without having to enter the specific IP address of the Apple servers.

### **Check your progress 8**

1. Network configuration cannot be done:
  - a. using CD
  - b. online
  - c. pen drive
  - d. hard disk

---

## **1.10 Basic troubleshooting/diagnostic commands**

---

Local area networks (LAN) are integral to the operation of many businesses today. The most common LANs use Ethernet, a data link layer protocol, and Internet Protocol (IP), a network layer protocol.

**Root causes of network troubleshooting problems are frequently caused by one of these three sources:**

### **1. Physical layer: copper, fibres or wireless**

Possible causes:

- Damaged or dirty cabling or terminations
- Excessive signal attenuation
- Insufficient cable bandwidth
- Wireless interference

### **2. Network Layer: Ethernet and IP**

Possible causes:

- Damaged networking devices
- Incorrect or sub-optimal device configurations
- Authentication and association issues
- Insufficient network bandwidth

### **3. Switches and VLANs**

Possible causes:

- Excessive utilization
- Too many errors

- Incorrectly assigned VLAN membership
- Traffic priority (CoS/QoS) issues

### Check your progress 9

1. What is the cause of trouble at network layer?
  - a. Excessive utilization
  - b. Damaged or dirty cabling or terminations
  - c. Incorrect or sub-optimal device configurations
  - d. Excessive signal attenuation

---

## 1.11 Let Us Sum Up

---

In this unit we have learnt that RJ connector is registered jack standardized physical network interface that connects telecommunications or data equipment. RJ45 could be a standard type of connector for network cables. To network two computers without a hub, a cross-over cable is used. Cross-over cable is also used to connect a router to a computer, or Ethernet switch (hub) to another Ethernet switch without an uplink.

Switches are often a valuable asset to networking. Overall, they can increase the capacity and speed of your network. However, switching shouldn't be seen as a cure-all for network problems. Client deployment refers to the planning, installation, and management of System Center 2012 Configuration Manager Client computers and mobile devices in your enterprise.

---

## 1.12 Answers for Check Your Progress

---

### Check your progress 1

Answers: (1 –c)

### Check your progress 2

Answers: (1 -a)

**Check your progress 3**

**Answers:** (1 –c)

**Check your progress 4**

**Answers:** (1 –b)

**Check your progress 5**

**Answers:** (1 -a)

**Check your progress 6**

**Answers:** (1 –d)

**Check your progress 7**

**Answers:** (1 –c)

**Check your progress 8**

**Answers:** (1 –b)

**Check your progress 9**

**Answers:** (1 –c)

---

## 1.13 Glossary

---

1. **Structured P2P** - where the nodes are arranged having a particular distributed data structure.
2. **Unstructured P2P** - where the nodes have arbitrarily selected other close nodes.
3. **Hybrid P2P** - where some nodes are presented as special functions in a good organized manner.
4. **Workstation-server Model** - Workstation may be a standalone system or a part of a network.
5. **Processor-pool Model** - Provides processing power on a demand basis.
6. **Integrated Hybrid Model** - Workstations used as processor pools.

---

## **1.14 Assignment**

---

Try to do Windows 8.1 Server Installation.

---

## **1.15 Activities**

---

Crimp a UTP cable into RJ 45 connector.

---

## **1.16 Case Study**

---

Study the LAN network of your college.

---

## **1.17 Further Readings**

---

1. Distributed Systems, Principles and Paradigms by Tanenbaum.
2. Distributed Systems, Concepts and Design by Coulouris, Dollimore, Kindberg.

---

## UNIT 2: CREATING A WI-FI LAN

---

### Unit Structure

#### 2.0 Learning Objectives

#### 2.1 Introduction to Wi-Fi Technology

#### 2.2 How to Provide Wi-Fi capability to a PC

#### 2.3 Creating an ad-hoc Wi-Fi based LAN

#### 2.4 Creating an infrastructure based LAN using Wireless AP

#### 2.5 Configuration of AP and client Machines

#### 2.6 Accessing data from File Server through Wi-Fi Interface from client machine

#### 2.7 Let Us Sum Up

#### 2.8 Answers for Check Your Progress

#### 2.9 Glossary

#### 2.10 Assignment

#### 2.11 Activities

#### 2.12 Case Study

#### 2.13 Further Readings

---

### 2.0 Learning Objectives

---

After learning this unit, you will be able to understand:

- About Wi-Fi Technology.
- About Wi-Fi based LAN.
- About AP and client Machines.
- About data from File Server.

---

### 2.1 Introduction to Wi-Fi Technology

---

Wi-Fi is a technology that has revolutionized the way we tend to network computers and electronic devices together, making wired connections



unnecessary. Wi-Fi allows networking of computers and digital devices without the need for wires. Data is transferred over radio frequencies, allowing Wi-Fi capable devices to receive and transmit data after they are in range of a Wi-Fi network. The widespread use of the technology and its availability in both residential homes and public places – as well as parks, gathering spots, and coffee shops – have made it one of the most popular information transmission technologies out there nowadays.

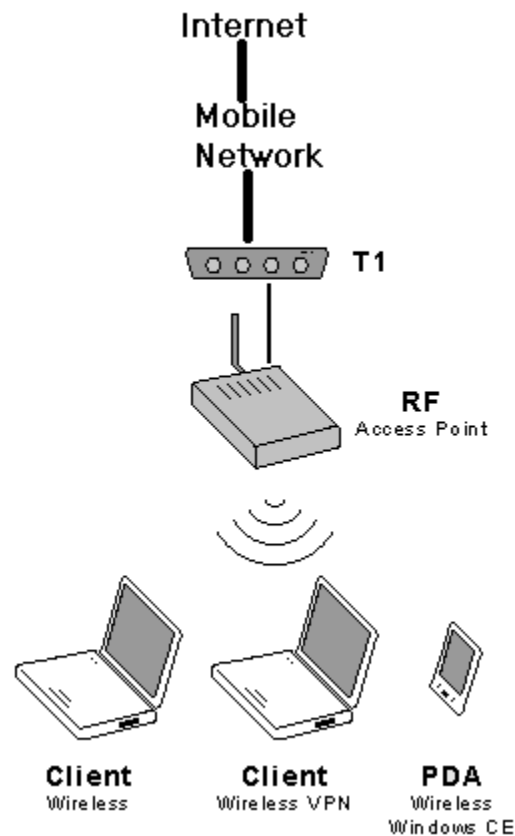


**Fig 2.1 WiFi arrangement**

Currently there are four major types of Wi-Fi which are:

- 802.11a
- 802.11b
- 802.11g
- 802.11n

Out of this the two most common and oldest types of Wi-Fi are:

- 802.11b
- 802.11g

Which operate at frequency of 2.4GHz.

It is seen that Wi-Fi 802.11b has theoretical maximum transmission speed of about 11Mbps while 802.11g will be able to transmit data at speeds up to 54Mbps. 802.11a was the next version of Wi-Fi developed, and it operated on a frequency of 5GHz and allowed data transmission at speeds of up to 54Mbps. It is not backwards compatible with 802.11b or g, due to its operation on a different frequency, thus limiting its use. 802.11n is the newest version of the technology and it is backwards compatible with devices running 802.11b or g. It operates at speeds up to 450Mbps on either 2.4GHz or 5GHz, either on a single channel or two channels. On dual channel devices, data transmission can theoretically exceed 450Mbps limit.

---

## 2.2 How to Provide Wi-Fi capability to a PC

---

Generally, PCs do not usually come with built-in Wi-Fi, especially older models. So if you need to get wireless connectivity you have options like- you can use either USB Wi-Fi adapter, a PCI-E Wi-Fi card, a new motherboard with built-in Wi-Fi.

**USB Wi-Fi Adapters:** It is plug and play device. The plug-and-play convenience means that you can remove it if it isn't needed, and use it in another PC.

**Wi-Fi PCI Cards:** If you have limited USB ports on your PC already, the Wi-Fi PCI card helps.

**Wi-Fi Enabled Motherboard:** If you're looking to upgrade your PC anyway, it might make more sense to upgrade your motherboard than buy an adapter.

### Check your progress 1

1. What is the full form of Wi-Fi?
  - a. Wired fidelity
  - b. Wireless Fidelity
  - c. Wireless files
  - d. None of these

---

## 2.3 Creating an ad-hoc Wi-Fi based LAN

---

An ad-hoc network is a local area network (LAN) that's built impromptu as devices connect. Rather than relying on a base station to coordinate the flow of messages to each node within the network, the individual network nodes forward packets to and from one another.

If you would like to share information stored on your pc with other people nearby and everyone's pc has a wireless network adapter, an easy methodology of sharing is to set up an ad hoc wireless network. In spite of the fact that members should be within 30 feet of each other, this type of network presents lots of prospects. In the Windows operating system, ad-hoc is a communication mode that allows computers to directly communicate with each other without a router.

Ad hoc networks are by definition temporary; they cease to exist when members disconnect from them, or once the computer from which the network was established moves beyond the 30-foot effective range of the others. you can share an internet connection through an ad hoc network, however keep in mind that the internet connection is then available to anyone logging on to a computer that's connected to the network, and thus is probably going not very secure

To set up an ad hoc network, follow these steps:

1. On the Start menu, click Connect To.
2. In the Connect to a network window, click the Set up a connection or network task.
3. On the Choose a connection option page, click Set up a wireless ad hoc (computer-to-computer) network, and then click Next.
4. Read the ad hoc network information, and then click Next.
5. Provide a network name, select whether the network is open or requires authentication, provide a security phrase if necessary, and then click Next.

After Windows Vista sets up ad hoc network, you can share Internet connection. If you wish to disconnect your connection from an ad hoc network, display the Connect To A Network window, click the ad hoc network, and then click Disconnect.

### **Advantages:**

- It is easy to set up if you just want to connect two devices to each other without using central access point.

- It is possible to connect them directly with ad-hoc mode to form a temporary Wi-Fi network without router.
- The new Wi-Fi Direct standard of ad-hoc mode, allow devices to communicate directly over Wi-Fi signals.

### **Check your progress 2**

1. An Ad-hoc network is a \_\_\_\_\_.
  - a. LAN
  2. B.WAN
  3. MAN
  4. None of these

---

## **2.4 Creating an infrastructure based LAN using Wireless Access Point**

---

In the case of wireless networking in Infrastructure mode you're connecting your devices using a central device, namely a wireless access point. To affix the WLAN, the AP and all wireless clients should be configured to use the same SSID. The AP is then cabled with the wired network to allow wireless clients access to, for example, internet connections or printers.

Most Wi-Fi networks are deployed in infrastructure mode. In infrastructure mode, a base station acts as a wireless access point hub, and nodes communicate through the hub. The hub typically, but not always, has a wired or fiber network connection, and will have permanent wireless connections to other nodes. Wireless access points are typically fastened, and provide service to their client nodes among range. Wireless clients, like laptops, smart phones etc. connect to the access point to join the network. Typically a network can have a multiple access points, with the same 'SSID' and security arrangement. In that case connecting to any access point on that network joins the client to the network. In this case, the client software can try and choose the access point to do to give the best service, like the access point with the strongest signal.

Infrastructure mode wireless networking is the mode that you most often encounter in your work as networking professional supporting networks for clients or in an exceedingly corporate environment.

At a minimum, the sole network infrastructure component that's required for Infrastructure mode is an access point, however if an AP is all you have, you have no more than you'd have had once using ad hoc mode. However, most Infrastructure mode implementations include other components from your traditional network infrastructure.

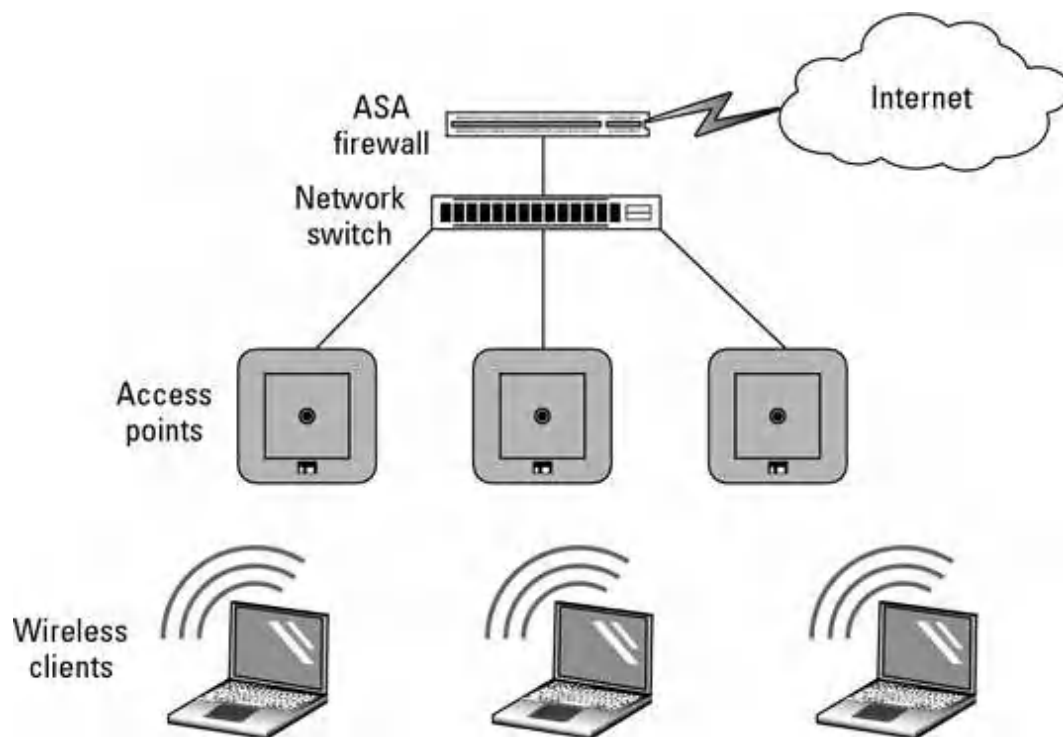


**Fig 2.2 network infrastructure component**

### Check your progress 3

1. What are wireless access points?
  - a. Connection points
  - b. It is a is a networking hardware device
  - c. Ports
  - d. None of these

---

## 2.5 Configuration of Access Point and client Machines

---

You can configure the access by using the following steps:

Step 1 - Connect your computer to other LAN port by Ethernet cable and login to TP-LINK web interface by IP address on bottom label.

Step 2 - Go to Network ->LAN on the left side menu, change the LAN IP address of TP-LINK router to the same segment of the main router

Step 3 - Configure the wireless

Go to Wireless->Wireless Settings page, configure the SSID (Network name) and Channel. Then Click Save button.

Step 4 - Go to DHCP on the left side menu, disable the DHCP Server and click **Save** button.

Step 5 - Go to System Tools->Reboot page, click Reboot button to reboot the device

Step 6 - Connect the main router to the LAN port on TP-Link router through Ethernet cable.

### Check your progress 4

1. Configuration of access point is a \_\_\_\_\_ installation.
  - a. LAN
  - b. WAN
  - c. WLAN
  - d. None of these

---

## 2.6 Accessing data from File Server through Wi-Fi Interface from client machine

---

You easily transfer information from file server using Wi-Fi technology from client machine. It is easy to use external hard drive and the router will create network storage accessibility for anyone on network. On setting up, any files stored on particular drive gets accessed by devices that are on particular network. Here's how to set it up on Windows.

### Step 1: Choose your drive

The drive which you want to share is small enough or can be large as TB and put it.

### Step 2: Enable USB drive sharing

On computer having Wi-Fi network, you can launch the browser and navigate your router's IP address which is at back of it or you can find it online. Now when the page gets loaded, you will be asked to log in with username and password.

It is noted that all router's interface are different, but if it is with USB port, then you can find a link written as File Sharing under Administrator settings. Here you have to enable file sharing server. Apply the settings, and exit.

### Step 3: Access the drive

Now after this, you are ready to read and write the attached hard drive by opening a file explorer window and navigating on to Network folder located at left sidebar. After this, clear the path at top of explorer window and enter \\[your IP address].



**Fig 2.3 File accessing**

You will find that your drive will appear. In this, open the contents which you want to transfer. If you're asked to log in, simply use your router's login details. If you wish to leave the attached hard drive so as to access from any Explorer window, you have to right-click on drive and select Map network drive.

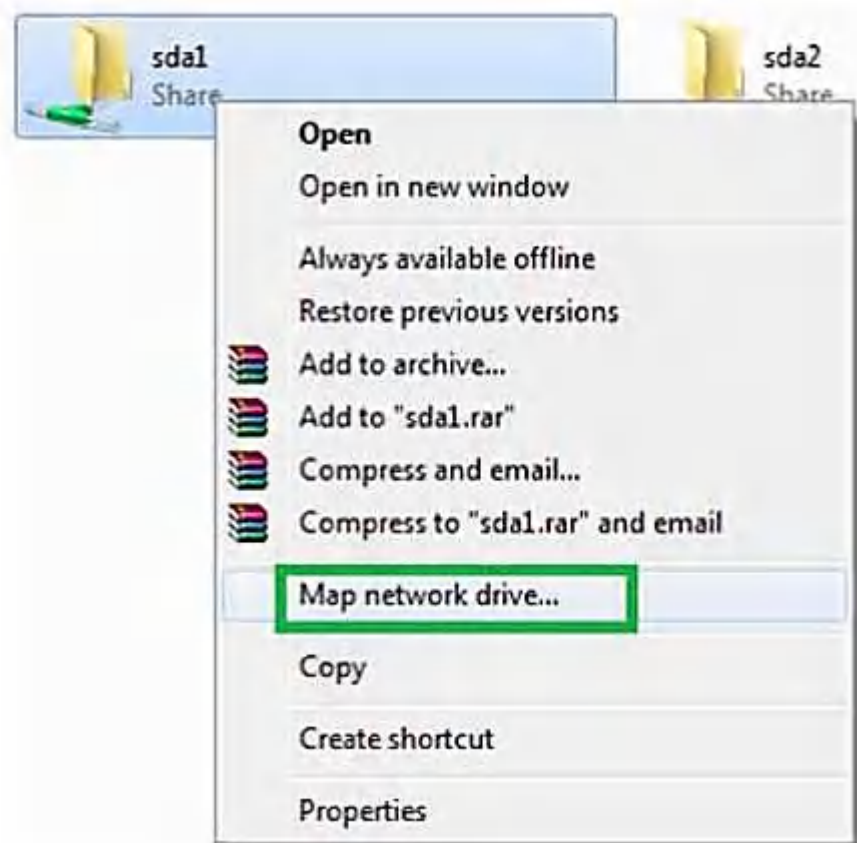


Fig 2.4. File transfer

### Check your progress 5

1. Transferring of data from client machine can be done by:
  - a. Configuring The Machine
  - b. Configuring The Data
  - c. Configuring The Files
  - d. All Of Above

---

## 2.7 Let Us Sum Up

In this unit we have learnt that Wi-Fi is a technology that has revolutionized the way we tend to network computers and electronic devices together, making wired connections unnecessary. An ad-hoc network is a local area network (LAN) that's built impromptu as devices connect. Rather than relying on a base station to coordinate the flow of messages to each node within the network, the individual network nodes forward packets to and from one another.



Wi-Fi networks are deployed in infrastructure mode. In infrastructure mode, a base station acts as a wireless access point hub, and nodes communicate through the hub.

---

## 2.8 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1 –b)

**Check your progress 2**

**Answers:** (1 –a)

**Check your progress 3**

**Answers:** (1 –b)

**Check your progress 4**

**Answers:** (1 –c)

**Check your progress 5**

**Answers:** (1-a)

---

## 2.9 Glossary

---

1. **Wi-Fi** - A network technology used in computing devices to connect to internet.
2. **Ad-hoc network** - It is a LAN network which is built in and depends on base station to arrange message flow.

---

## 2.10 Assignment

---

Explain infrastructure based LAN using Wireless Access Point

---

## **2.11 Activities**

---

Create an ad-hoc Wi-Fi based LAN for your college

---

## **2.12 Case Study**

---

How does Wi-Fi play an important role in accessing internet?

---

## **2.13 Further Readings**

---

1. Distributed Systems, Principles and Paradigms by Tanenbaum.
2. Distributed Systems, Concepts and Design by Coulouris, Dollimore, Kindberg.

---

## **Block Summary**

---

In this block, you will understand about the basic of client deployment features with various working characteristics related to planning, installation and management of System Center 2012 Configuration Manager are well detailed. The block gives an idea on architecture and distribution of various forms of RJ45 USB connectors with certain characteristics features. The examples related to concept of different RJ cables along with working characteristics are well detailed.

In this block, you will understand about the basic of Wi-Fi is a technology and its techniques. The concept related to Ethernet switch and hub along with working features is detailed. You will be demonstrated practically about RJ45 crimping technology.

---

## Block Assignment

---

### Short Answer Questions

1. What is Wi-Fi and why it is used?
2. What are Ethernet cables?
3. What are access points and why are they important?
4. How can we create an Ad- hoc Wi-Fi based LAN?

### Long Answer Questions

1. Explain in detail the steps to configure access points in client side.
2. How can we provide Wi-Fi capabilities to a PC?
3. What is the color-coding scheme for Ethernet?

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                   | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|-----------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                    | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Language and Style                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Illustration used (Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Conceptual Clarity                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Check your progress Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Feed back to CYP Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# OBJECT ORIENTED CONCEPTS

PGDCA-102

## BLOCK 2: JAVA CONTROL STATEMENTS, OPERATORS AND CLASSES



**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**

# OBJECT ORIENTED CONCEPTS

---



Knowledge Management and  
Research Organization  
Pune





## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Tejashree Bhosale

Mr. Kiran Shinde

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



## **OBJECT ORIENTED CONCEPTS**

### **Contents**

#### **BLOCK 1: INTRODUCTION TO PROGRAMMING WITH JAVA**

##### **UNIT 1 PROBLEM SOLVING WITH COMPUTERS**

Need of programming, Algorithms, Flowcharts, Pseudo-Code, Programming Languages Overview, and Overview of Object Oriented Concepts, Skills check, Exercises

##### **UNIT 2 INTRODUCTION TO JAVA**

History, Overview of JVM & Bytecode, Java versions (J2me etc.), Java class library, JDK, Your First Java Application, Skills check, Exercises

##### **UNIT 2 BEGINNING WITH JAVA PROGRAMMING**

Java Keywords, Comments in Java, Variables and Assignments, Strings and Characters, Arithmetic Operators and Expressions, Type Conversion in Expressions, Type Conversion, Skills check, Exercises

---

#### **BLOCK 2: JAVA CONTROL STATEMENTS, OPERATORS AND CLASSES**

##### **UNIT 1 JAVA CONTROL STATEMENTS**

The if statement, The if-else statement, Blocks of Code, The for statement, Increment & Decrement operators, Backslash codes, Relational and Boolean logical operators, Ternary operators, Skills check, Exercises

##### **UNIT 2 MORE ABOUT CONTROL STATEMENTS AND OPERATORS**

Nested if statement, Variations of the for loop, The while loop, The do loop, Nested loops, The break statement, The continue statement, The switch statement, The bitwise operators, Skills check, Exercises



### **UNIT 3**

### **CREATING CLASSES**

The General form of a class, Creating Simple Classes, Adding Constructors, Constructor Overloading, The this keyword, Instance variables and methods, Static variables and methods, Local variables and its scope, Method overloading, Argument Passing, Wrapper Classes, System Class, Garbage Collection, Skills check, Exercises

---

## **BLOCK 3: INHERITANCE, INTERFACE, PACKAGES AND EXCEPTIONS IN JAVA**

### **UNIT 1 INHERITANCE**

Overview of Re-usability, Subclasses, Inheritance and Variables, Method Overriding, Inheritance and Methods, Inheritance and Constructors, Class Modifiers, Variable Modifiers, Constructor Modifiers, Method Modifiers, Object and Class classes, Skills Check, Exercises

### **UNIT 2 INTERFACES AND PACKAGES**

Interfaces, Interface References, Interface Inheritance, The instanceof Operator, Packages, The import statement, Access control and Packages, Skills Check, Exercises

### **UNIT 3 EXCEPTIONS**

Overview of Exceptions, Exception Handling, Catch Block and searches, The throw statement, Exception and Error classes, The throws clause, Custom exceptions, Skills check, Exercises

## **BLOCK 4: JAVA CLASS LIBRARY, FILE HANDLING AND GUI**

### **UNIT 1 INTRODUCTION TO JAVA CLASS LIBRARY**

Classes of java.util package, Classes of java.net package, Classes of java.lang package, Overview of Collection Framework, Skills check, Exercises



## **UNIT 2**

### **FILE HANDLING IN JAVA**

Overview of File Handling, File Class in Java, Using Character Stream Classes, Using Byte Stream Classes, Using Scanner and Console Classes in Java, Skills check, Exercises

## **UNIT 3**

### **BUILDING GRAPHICAL USER INTERFACE (SWING)**

Building applications using classes related to Graphical Interface, Creating Layouts in GUI, Using Events in GUI applications, Skills check, Exercises



Dr. Babasaheb  
Ambedkar  
Open University

**PGDCA-102**

## **OBJECT ORIENTED CONCEPTS**

---

### **BLOCK 2: JAVA CONTROL STATEMENTS, OPERATORS AND CLASSES**

---

#### **UNIT 1**

|                         |    |
|-------------------------|----|
| JAVA CONTROL STATEMENTS | 03 |
|-------------------------|----|

#### **UNIT 2**

|                                             |    |
|---------------------------------------------|----|
| MORE ABOUT CONTROL STATEMENTS AND OPERATORS | 22 |
|---------------------------------------------|----|

#### **UNIT 3**

|                  |    |
|------------------|----|
| CREATING CLASSES | 43 |
|------------------|----|



---

# **BLOCK 2:    JAVA CONTROL STATEMENTS, OPERATORS AND CLASSES**

---

## **Block Introduction**

In Java, when any statement happens to skip or executed more than once, then it needs a control. The operators are used to carry out primitive data types which can be categorised as unary, binary or ternary.

In this block we will learn and study about concept involved in studying and working of control structures with stress on their syntaxes. The outline of these structures with their programming techniques is well explained. The knowledge about different types of operators with their features are helps the students to grasp more. The detailed about loops and statements with several illustrations are provided which could be of any future use to students.

After completing this block, students will be able to know more about control structures and its features with basic operations about operators and statements. The concept of constructor and garbage collector will allow him to work on certain Java control platforms. After this block, students will get knowledge and confidence about working with loops, operators, structures and statements.

## **Block Objective**

**After learning this block, you will be able to understand:**

- Control structures
- Study the syntax of IF structure
- Basic of IF-Else statement
- Concept of FOR statement
- Idea about prefix increment/decrement operators
- Detail about newline character
- Idea about ternary operator
- Structure of Nested IF statement

Java Control  
Statements,  
Operators and  
Classes

- Idea about variation in FOR loop
- Detail about condition in while loop
- Idea about working of do..while loop
- Knowledge about processing of Nested loop
- Study the functions about Break Statement
- Idea about continue statement
- Basic about working of Switch statement
- The basic of three classes
- Knowledge about role of constructor
- Idea about constructor overloading
- Features of garbage collector

## **Block Structure**

**Unit 1: Java Control Statements**

**Unit 2: More about Control Statements and Operators**

**Unit 3: Creating Classes**

---

## **UNIT 1:      JAVA CONTROL STATEMENTS**

---

### **Unit Structure**

- 1.0   Learning Objectives**
- 1.1   Introduction**
- 1.2   The if Statement**
- 1.3   The if-else Statement**
- 1.4   Blocks of Code**
- 1.5   The for Statement**
- 1.6   Increment and Decrement Operators**
- 1.7   Backslash Codes**
- 1.8   Relational and Boolean Logical Operators**
- 1.9   Ternary Operators**
- 1.10   Skills Check**
- 1.11   Exercises**
- 1.12   Let Us Sum Up**
- 1.13   Answers for Check Your Progress**
- 1.14   Glossary**
- 1.15   Assignment**
- 1.16   Activities**
- 1.17   Case Study**
- 1.18   Further Readings**

---

### **1.0      Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Study of if statement
- Study of if-else statement
- Study of Blocks of Code
- Study of Increment & Decrement operators

- Study of Backslash codes

---

## 1.1 Introduction

---

The programs that were written and designed up till now, carries a sequential flow of control. It means that the statements initially were executed line by line from top to bottom in a particular order. It is found that when any statement happens to skip or executed more than once, then it needs a control. In this unit we will see that such can be done using control structures. The control structure carries three groups:

- Decision making statements
- Repetition statements
- Branching statements

Decision making is type of control statements that will decide whether or not a given statement or group of statements be worked out or not. Repetition statements are such where the statements are carried out more than ones. Branching statements on the other hand transfers the control to another line in a given code.

---

## 1.2 The if Statement

---

It is seen that the if statement is applied to particular statement or block of statements to carried out. The if statement carries a condition that will judge whether the statements will worked out or not. It is found that when the condition in parentheses works to be true, then statements gets executed, otherwise, it will be skipped off. Conditions are created using conditional and relational operators. Following is syntax of if statement as:

```
if (condition) {  
    // code  
}
```

There is another code shown which displays the message "Mohit" when the value of boolean variable becomes true.

```
if (wish == true) {  
    System.out.println("Nishit");  
}
```

In the code, the braces get erased. These are not used in case of block consists of only single statement. Also, the condition 'wish==true' can be described as 'wish'.

### Check your progress 1

1. Which among the following acts as a selective statement?

- |         |            |
|---------|------------|
| a. if   | c. for     |
| b. goto | d. else if |

---

## 1.3 The if-else Statement

---

It is studied that if statement is called as single selection structure where only single statement can be worked out. Under such condition, no alternative statement will work when the condition appears to be false. For this, we need to have if else statement which is a double selection structure statement. In this when the condition becomes true, and then if block will worked out, otherwise, else block will work. The syntax of if else structure is shown:

```
If (condition ){  
    // code  
} else {  
    //code  
}
```

In an example showing whether the number is even an odd number, it is found that an even number is that which is divisible by 0 and leaves a remainder of 0. The remainder when divides number by 2 can be obtained with the help of modulo (%) mathematical operator. Consider the syntax:

```
If (num % 2 == 0 ){  
    System.out.println("Number is even");  
} else {  
    System.out.println("Number is odd");  
}
```

In this, we see that an if else statement is nested where block of if or else carries another if else structure. Here, an else is always linked with previous else. Such type of concepts forms a chained if else structure where only single several alternatives worked. Consider the code shown where if else structures have been nested to obtain number stored in int num:

```
if(num=1){  
    System.out.println("One");  
} else {  
    if (num = 2) {  
        System.out.println("Two"); } else {  
            if (num = 3) {  
                System.out.println("Three"); } else {  
                    System.out.println("Numbers greater than three cannot be processed  
by this code");  
                }  
            }  
        }  
    }  
}
```

It is found that the complete block is attached with else even if no braces are used. On removing the braces, we are left with as shown below. We have also written the words if and else on the same line.

```
if (num = 1) {  
    System.out.println("One");  
} else if (num = 2) {  
    System.outprintln(Two");  
} else if (num = 3) {  
    System.out.println("Tbree");  
} else {  
    System.out.println("Numbers greater than three cannot be processed  
by this code");  
}
```

The above code shows the chained if else structure which is obtained using a number of if and else statements.

### Check your progress 2

1. What will be value obtained, if the program is executed?

```
Public class ControlStatement {  
    Public static void main (string [] args){  
        int a = 25;  
        if (~a>25)  
            a++;  
            a+=a;  
        system.out.print(a);  
    }  
}
```

a. -49

c. 48

b. 50

d. 65

---

## 1.4 Blocks of Code

---

In Java, there exist two or more statements that are grouped in blocks of code which is often called as code blocks that can be seen in enclosed statements among opening and closing curly braces. On forming such block of code, it becomes logical unit which are applied in case of single statement. Consider this if statement:

```
if (x < y){  
    x = y;  
    y = 0;  
}
```

In the above code we see that if x is < y, then both statements inside block will get executed. So the two statements in block generate a logical unit where

statement cannot be worked without others. Consider an example where a block of code as the target of for loop is shown:

```
/* Demonstrate a block of code.  
  
Call this file "Test.java"  
  
*/  
  
class Test {  
    public static void main (String args[]) {  
        int x, y;  
        y=20;  
        for ( x= 0; x<10; x++) {  
            System.out.println("This is x: " x);  
            System.out.println("This is y: " y);  
            Y = y-2;  
        }  
    }  
}
```

The above program will generate an output as shown:

```
This is x: 0  
This is y: 20  
This is x: 1  
This is y: 18  
This is x: 2  
This is y: 16  
This is x: 3  
This is y: 14  
This is x: 4  
This is y: 12  
This is x: 5  
This is y: 10
```



This is x: 6

This is y: 8

This is x: 7

This is y: 6

Here, we see that the target of for loop is block of code and not simply a single statement. So every time the loop will repeat three statements present inside block.

### Check your progress 3

1. It is seen that a compound statement in Java is enclosed in:
  - a. parenthesis
  - b. braces
  - c. square brackets
  - d. all of them

---

## 1.5 The for Statement

---

In Java, it is seen that a for loop runs with group of Java statements till boolean condition results as true. Such loop will combine three elements such as initialization statement, boolean expression and increment or decrement statement. The syntax of for loop is:

```
for (<initialization> ; <condition> ; <statement>) {  
    <Block of statements>;  
}
```

In this, we find that the initialization statement will get worked out before loop begins. Normally, it is applied to initialize loop variable. Here the condition statement will run prior to block of statements. In such case, the block of statement will work only when boolean condition becomes true. Hence, this is used to increment or decrement loop variable as shown in example.

```
for (int i = 0; i < 5 ; i++)  
{  
    System.out.println("i is : "+i);  
}
```

Here we find that it is possible to initialize various variables in initialization block of for loop with the help of comma shown below:

```
for (int i = 0, j = 5; i++ )
```

Here we can have more than one increment or decrement portion as shown:

```
for (int i = 0; i < 5; i++, j++ )
```

The flow chart of working of for statement in Java is shown in fig 1.1

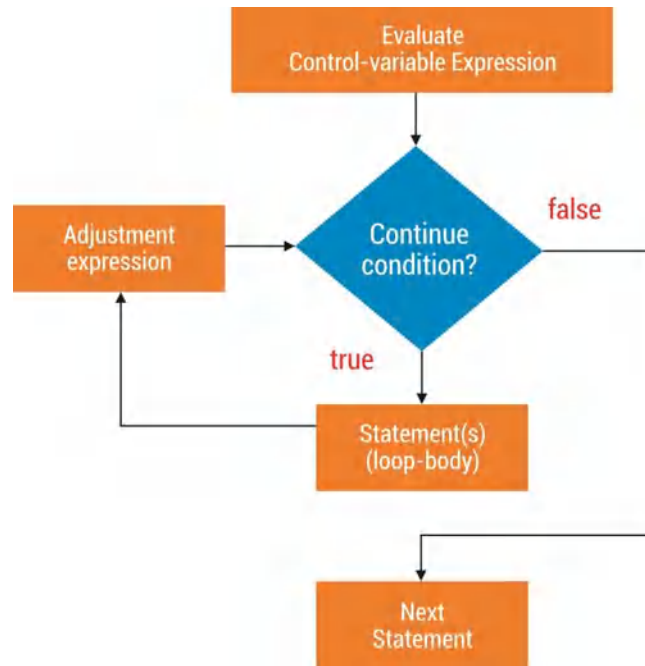


Fig 1.1 Flow chart of for statement

### Check your progress 4

1. What will the value obtained from the program?

```
Public class loop {  
    Public static void main (String[] args){  
        Integer a = 012, b;  
        for (b=0;b<a;b++)  
            System.out.print(b);  
    }  
}
```

a. 9

b. 11

c. 15

d. 18

---

## 1.6 Increment and Decrement Operators

---

In Java, we have noticed that there exists two type of Increment or decrement operators as ++ and --. These operators are exceptional as they can be written both prior to operand as applied and is known as prefix increment/decrement or when used after will know as postfix increment/decrement. Consider an example:

```
x = 1;

y = ++x;

System.out.println(y);

prints 2, but

x = 1;

y = x++;

System.out.println(y);

prints 1

Source Code

//Count to ten

class UptoTen {

    public static void main (String args[]) {

        int i;

        for (i=1; i 0; i++)

            System.out.println(i);

        }

    }
```

If we write i++, then it means  $i = i + 1$  and when i--, then  $i = i - 1$ . Here we will add and subtract one from number which is common operations which could be done by special increment and decrement operators. Also, for add and assign operation, we can use +=. Normally it is seen that we write as  $i += 11$ . It means that we want to count from 0 to 15 by two's as:

```
class CountToFifteen {

    public static void main (String args[]) {
```

```
int i;  
for (i=0; i <=15; i += 2) { //Note Increment Operator by 2  
    System.out.println(i);  
}  
}  
} //main ends here  
}
```

Consider an operator `--`, if we want to count from ten to zero by twos, then will write this as:

```
class CountToZero {  
    public static void main (String args[]) {  
        int I;  
        for (i=10; i >= 0; i-=2) { //Note Decrement Operator by 2  
            System.out.println(i);  
        }  
    }  
}
```

### Check your progress 5

1. What do you mean by variable++?
  - a. Adding 1 to variable.
  - b. Adding 1 to variable after using its current value.
  - c. Adding 1 to variable before using its value.
  - d. Double the value of variable.

---

## 1.7 Backslash Codes

---

These are characters which cannot be applied from keyboard and cannot be printed. These can be used anywhere as normal character and are termed as escape sequences. It is seen that a single most important backslash code is `\n`, which is called as newline character. We see that backslash codes are character constants

which can be assigned to character variable inside single quotes. It can be written as:

```
char = ch;
```

```
ch = '\u0009';
```

### Backslash types

| Backslash Code | Description                                                |
|----------------|------------------------------------------------------------|
| \t             | tab character ('\u0009')                                   |
| \n             | new line or line feed character ('\u000A')                 |
| \r             | carriage-return character ('\u000D')                       |
| \f             | form-feed character ('\u000C')                             |
| \a             | alert or bell character ('\u0007')                         |
| \e             | escape character ('\u001B')                                |
| \cx            | control character corresponding to x                       |
| \\             | backslash character                                        |
| \0n            | character with octal value 0n (0 <= n <= 7)                |
| \0nn           | character with octal value 0nn (0 <= n <= 7)               |
| \0mnn          | character with octal value 0mnn (0 <= m <= 3, 0 <= n <= 7) |
| \xhh           | character with hexadecimal value 0xhh                      |
| \uhhhh         | character with hexadecimal value 0xhhhh                    |

### Check your progress 6

1. What is \\?

- character with hexadecimal value
- escape character
- backslash character
- alert character

---

## 1.8 Relational and Boolean Logical Operators

---

### Relational operators

There are many types of relational operators such as <, <=, >, >=, !=, ==. These operators can be used and checked with the if statements as:

```
if (x == 2)
{
    if (y!=2)
    {
        System.out.println("Both conditions are true.");
    }
}
```

In Java, it is easier to handle multiple logic conditions with logic operators like &&, || and !.

It is found that && is logical and && combines two boolean values that returns boolean as true when only both of operands are true as shown:

boolean b;

```
b = 3 > 2 && 5 < 7; // b is true
```

```
b = 2 > 3 && 5 < 7; // b is now false
```

|| is logical or. || combines two boolean variables or expressions that returns result as true when either or both of operands are true as shown:

boolean b;

```
b = 3 > 2 || 5 < 7; // b is true
```

```
b = 2 > 3 || 5 < 7; // b is still true
```

```
b = 2 > 3 || 5 > 7; // now b is false
```

Here we find that the last logic operator is ! which states not. This will reverse the value of boolean expression. Now if b is true, then !b is false and if b is false, then !b is true as shown:

boolean b;

```
b = !(3 > 2); // b is false
```

```
b = !(2 > 3); // b is true
```

Such operators will test for multiple conditions easily as shown in earlier example which can now be written as:

```
if (x == 2 && y != 2)
{
    System.out.println("Both conditions are true.");
}
```

## Boolean logical operators

There are also many types of Boolean logical operators such as `|` , `&` , `^` , `!` , `||` , `&&` , `==` , `!=` . These operators will take the value to be true or false, and have default value as false. The main use of Boolean facilities is to work with expressions that controls if decisions and while loops. Such operators act on Boolean operands as per the table shown:

### Boolean operands

| A     | B     | A B   | A&B   | A^B   | !A    |
|-------|-------|-------|-------|-------|-------|
| false | false | false | false | false | true  |
| true  | false | true  | false | true  | false |
| false | true  | true  | false | true  | true  |
| true  | true  | true  | true  | false | false |

`|` the OR operator

`&` the AND operator

`^` the XOR operator

`!` the NOT operator

`||` the short-circuit OR operator

`&&` the short-circuit AND operator

`==` the EQUAL TO operator

`!=` the NOT EQUAL TO operator

## Boolean operators

| Operator                                  | Comments                                                                                                             |
|-------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| <code>==(equality)</code>                 | Returns true when both operands are equal.<br>The operands are converted to the same type before being compared.     |
| <code>!=(non-equality)</code>             | Returns true when both operands are not equal.<br>The operands are converted to the same type before being compared. |
| <code>===(equality)</code>                | Returns true if both operands are equal and of the same type.                                                        |
| <code>!=(non-equality)</code>             | Returns true if both operands are not equal and of the same type.                                                    |
| <code>&gt;(greater than)</code>           | Returns true if the left operand is greater than the right one.                                                      |
| <code>&gt;=(greater than or equal)</code> | Returns true if the left operand is greater than or equal to the right one.                                          |
| <code>&lt;(less than)</code>              | Returns true if the left operand is greater than the right one.                                                      |
| <code>&lt;=(less than or equal)</code>    | Returns true if the left operand is greater than or equal to the right one.                                          |

### Check your progress 7

1. A relational operation returns \_\_\_\_ value.
  - a. int
  - b. char
  - c. float
  - d. boolean
2. Which among the following returned by greater than, <, and equal to, ==, operator?
  - a. Integers
  - b. Floating point numbers
  - c. Boolean
  - d. None of these

---

## 1.9 Ternary operators

---

In Java, the ternary operator also shown as `?`, it is a type of if else statement which can be used to find an expression and return one of two operands as per the result of an expression.

```
boolean b = true;
```

```
String s = ( b == true ? "True" : "False" );
```

In this we see that the value of set of String s is set as per the value of boolean b, which can be represented by if else statement as:



```
boolean b = true;

String s;

if(b == true)

{

    s = "True";

}else{

    s = "False";

}
```

Such type of operator is useful in lowering number of lines of code. It can also be useful to test null values.

### Check your progress 8

1. What will be highest value among three integers as per ternary operations?

```
#include<stdio.h>

#include<conio.h>

void main()

{

    int 12, 6, 15, big ;

    clrscr() ;

    printf("Enter three numbers : ") ;

    scanf("%d %d %d", &a, &b, &c) ;

    big = a > b ? (a > c ? a : c) : (b > c ? b : c) ;

    printf("\nThe biggest number is : %d", big) ;

    getch() ;

}
```

- |       |         |
|-------|---------|
| a. 12 | c. 15   |
| b. 6  | d. none |

---

## 1.10 Skills Check

---

In Java, there are certain skills involved while analysing control structures. There are three such types of structures as Decision making, Repetition and Branching. The basic of control statement is if statement which is the single selection structure having one statement. You need to practice certain examples based on if statement in order to go for else if and nested if. There are two or more statements that are grouped in blocks of code which is often called as code blocks. Skills are involved in evaluating certain ternary operations which will clear the logic involve.

### Check your progress 9

1. Which is not a part of control structure?
  - a. Decision making
  - b. Repetition
  - c. Branching
  - d. Boolean

---

## 1.11 Exercises

---

### Example 1:

Consider a program which will show use of newline character.

```
class NewLineCharDemo
{
    public static void main (String args[])
    {
        System.out.print("This is first line. \n");
        System.out.print("This is second line.");
        System.out.print("This is third line.");
    }
}
```

**Output**

This is first line.

This is second line. This is third line.

---

## 1.12 Let Us Sum Up

---

In this unit we have learnt that, the control structure in Java carries three group statements such as Decision making statements, Repetition statements and Branching statements.

It is studied that if statement is called as single selection structure where only single statement can be worked out.

In Java, there exist two or more statements that are grouped in blocks of code which is often called as code blocks which is seen in enclosed statements among opening and closing curly braces.

In Java, the ternary operator also shown in shape of Question mark (?) is a type of if else statement which can be used to find an expression and return one of two operands as per the result of an expression

---

## 1.13 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-a)

**Check your progress 2**

**Answers:** (1-b)

**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-b)

**Check your progress 5**

**Answers:** (1-b)

**Check your progress 6**

**Answers:** (1-c)

**Check your progress 7**

**Answers:** (1-d), (2-c)

**Check your progress 8**

**Answers:** (1-c)

**Check your progress 9**

**Answers:** (1-d)

---

## 1.14 Glossary

---

1. **Modulus operator** - It is an operator which is shown with percent sign (%) that works on integers and gives a remainder when a number is divided by another number.
2. **Boolean expression** - It is an expression which shows the result as true or false.
3. **Comparison operators** - These are operators such as ==, !=, >, <, >=, and <= which compares two values.
4. **Logical operators** - These are type of operators like and, or, and not which combines with boolean expressions.
5. **Block** - It is a group of consecutive statements having similar indentation.

---

## 1.15 Assignment

---

Discuss the different types of Control structure statements in Java?

---

## 1.16 Activities

---

Calculate the output of the following program?

```
for(i=1,j=0;i<10;i++)  
    j+=I;  
System.out.println(i);
```

Java Control  
Statements

---

## 1.17 Case Study

---

Discuss various control structures in Java?

---

## 1.18 Further Readings

---

1. Learning Programming by Peter Norvig's
2. Approach programming with a more positive by P.Brian.Mackey

---

## **UNIT 2: MORE ABOUT CONTROL STATEMENTS AND OPERATORS**

---

### **Unit Structure**

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 Nested if Statement**
- 2.3 Variations of for Loop**
- 2.4 The While Loop**
- 2.5 The Do Loop**
- 2.6 Nested Loops**
- 2.7 The Break Statement**
- 2.8 The Continue Statement**
- 2.9 The Switch Statement**
- 2.10 The Bitwise Operators**
- 2.11 Skills Check**
- 2.12 Exercises**
- 2.13 Let Us Sum Up**
- 2.14 Answers for Check Your Progress**
- 2.15 Glossary**
- 2.16 Assignment**
- 2.17 Activities**
- 2.18 Case Study**
- 2.19 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Nested IF statement
- While do loops

- Switch and break statements
- Bitwise operators

---

## 2.1 Introduction

---

In Java operators are used to carry out primitive data types which can be categorised as unary, binary or ternary. These operators take one, two or three arguments respectively. It is found that a unary operator exists before or after an argument. As seen a binary or ternary operator will exist in between the arguments. Apart from these there are several divisions of such operators:

- assignment
- arithmetic
- relational
- logical
- bitwise
- compound assignment
- conditional
- type

---

## 2.2 Nested if Statement

---

When an if statement is having a target of another if or else statement, then it is said to be nested inside the outer if. This statement will put one IF Statement inside another.

The syntax of Nested IF statement is shown:

```
if (expression) statement;
if (expression) statement;
    |
    |
    |
    if (expression) statement;
```

Consider another form of an IF statement:

```
    if (expression) statement;  
    else  
        if (expression) statement;  
    else  
        if (expression)  
        |  
        |  
        |  
    else statement;
```

To know more about how this statement works in Java consider an example shown. Write a program using if statement which will check number as zero, positive or negative.

```
class CheckSignNumberDemo  
{  
    public static void main(String args[])  
    {  
        int x = 10;  
        if(x > -1)  
            if(x != 0)  
                if(x > 0)  
                    System.out.println("x is a positive number having value " + x);  
    }  
}
```

### **Output**

x is a positive number having value 10



### Check your progress 1

1. While executing a program having nested loops, which loop will be executed maximum number of times?
  - a. outermost loop
  - b. innermost loop
  - c. all loops
  - d. cannot determined

---

## 2.3 Variations of for Loop

---

In Java, it is seen that for is one of the most versatile statement as it allows a broad range of variations. Consider an example:

```
public class for loop
{
    public static void main (String [] args)
    {
        for (int i=5;i<10;i++)
        {
            System.out.println("i="+i);
        }
    }
}
```

If we run the above program, we will see that it will generate an output as:

```
i=5
i=6
i=7
i=8
i=9
```

In this program, we see that an initialization statement, condition statement and iteration statement are not compulsory and can be kept empty. If we include a variation, then the above program can be written as:

```
Public class for loop
```

```
{  
    public static void main (String [] args)  
    {  
        int i=5;  
        for (;i<10;) {  
            System.out.println("i="+i);  
            i++;  
        }  
        System.out.println("After for loop");  
    }  
}
```

It is seen in the above program, that initialization and iteration part remains empty. If we write an initialization statement before for loop, then increment the counter inside body of loop.

### Check your progress 2

1. What will be value of i in the program on execution?

```
public class Main {  
    public static void main(String args[]) {  
        int i = 0;  
        boolean done = false;  
        for (; !done;) {  
            System.out.println("i is " + i);  
            if (i == 10)  
                done = true;  
            i++;  
        }  
    }  
}
```

a. 0

b. 1

c. 2

d. all of these

## 2.4 The While Loop

In Java, while loop is applied to carry out statement(s) till condition remains to be true. First we see about the syntax of loop:

```
while (condition(s))
{
    // Body of loop
}
```

In the above we see that if condition remains true then body of loop will carry out soon after execution of loop body condition which is checked again. When the condition is true then body of loop gets carried out again and process will continue till the condition becomes false. In looping, condition should always worked out first which can be true or false, and when it is constant as while (c) { ...} where c is constant, then any non-zero value of c will be true and zero remains false.

Also, through testing of multiple conditions like:

```
While (a > b && c !=0)
{
    // Loop body
}
```

The loop body gets worked out till value of a is more than value of b where c is not equal to zero. It is seen that body of loop carries more than single statement. In case of multiple statements, it is required to put them in block having {} and when the body of loop has only single statement, then you can optionally use such bracket {}. To know more, consider an example where a program asks the user to input an integer and show it until user enters 0.

```
import java.util.Scanner;

class WhileLoop {
    public static void main(String[] args) {
        int n;

        Scanner input = new Scanner(System.in);

        System.out.println("Input an integer");

        while ((n = input.nextInt()) != 0) {
```

```
System.out.println("You entered " + n);  
System.out.println("Input an integer");  
}  
System.out.println("Out of loop");  
}  
}
```

You will find that the output of such program will show:

```
E:\Java>javac WhileLoop.java  
  
E:\Java>java WhileLoop  
Input an integer  
7  
You entered 7  
Input an integer  
-2  
You entered -2  
Input an integer  
9546  
You entered 9546  
Input an integer  
0  
Out of loop  
  
E:\Java>_
```

### Check your progress 3

1. In Java, the while loop is applied till condition remains:
  - a. false
  - b. true
  - c. imaginary
  - d. all of these

---

## 2.5 The Do Loop

---

A do...while loop is similar to a while loop, except that a do...while loop is guaranteed to execute at least one time.

Syntax:

The syntax of a do...while loop is:

```
do  
{
```

```
//Statements
}while(Boolean_expression);
```

Notice that the Boolean expression appears at the end of the loop, so the statements in the loop execute once before the Boolean is tested.

If the Boolean expression is true, the flow of control jumps back up to do, and the statements in the loop execute again. This process repeats until the Boolean expression is false.

Example:

```
public class Test {
    public static void main(String args[]){
        int x = 10;
        do{
            System.out.print("value of x : " + x );
            x++;
            System.out.print("\n");
        }while( x < 20 );
    }
}
```

This would produce the following result:

```
value of x : 10
value of x : 11
value of x : 12
value of x : 13
value of x : 14
value of x : 15
value of x : 16
value of x : 17
value of x : 18
value of x : 19
```

### Check your progress 4

1. In Java, do...while loop will execute:

- |                      |                   |
|----------------------|-------------------|
| a. at least one time | c. repeated twice |
| b. every time        | d. none of these  |

---

## 2.6 Nested loops

---

In Java, Nested loops are very helpful. The nested loops can be better understandable by considering example of pattern problems. In this, the integer value is taken as input from user with the result that following pattern gets printed based on input. Now, if we have input as 7, then we see that following 7 star pattern gets printed as:

```
*  
  
**  
  
***  
  
****  
  
*****  
  
*****
```

From this pattern we see that loops are used to print such pattern as we are not aware in advance of number of \*'s that to be printed. For doing this we require nested loops which is loop within another loop. Here the outer FOR loop keeps track of the line number while the inner FOR loop is used to keep track of the number of stars we are printing. It is analysed that number of stars on a particular line is equal to line number which are related to outer loop control variable. Keeping these things in mind, we can write the following code which prints this pattern. The number n is taken as an input from the user:

```
for (int i = 1; i <= n; i++) {  
    for (int j = 1; j <= i; j++) {  
        System.out.print("*");  
    }  
    System.out.println();  
}
```

It is seen that while using print() method and not println() to show star patterns similarly like \*'s as displayed on same line. Once the inner loop is executed, we can move to next line with the help of println() statement.

### Check your progress 5

1. In Nested loop, the inner loop is:
  - a. while loop
  - b. for loop
  - c. do..while loop
  - d. none of these

## 2.7 The Break Statement

In Java, the Break Statement applies to bring about loop control statements. Such type statement will break for Loop, while Loop and do-while loop. This statement will skip the remaining statements and will execute that statement which is after loop.

The Break statement is used when condition is satisfied inside loop. It is applied to make looping statements more flexible and powerful. The syntax of the Break Statement is break;

### Different Ways of Using Break Statement in Java:

#### Break Statement used in For Loop

The break statement will take control out of the loop where all statements gets carried out. As shown, statement 1, statement 2.... will go on till it reach to break statement where loop will break and takes the control outside the loop.

```

for ( initialization; condition; increment )
{
    Statement 1;
    Statement 2;
    Statement 3;
    .....
    .....
    .....
    break;
    Statement N-1;
    Statement N;
}
OutsideStatement 1;
  
```

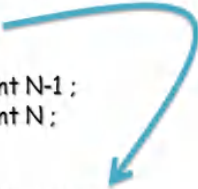


Fig 2.1 Break statement used in For loop

It is seen that in above arrangement of break statement, it is found that the control will take place outside the loop.

### Break Statement used in While Loop

```
Initialization;  
while (condition)  
{  
    Statement 1;  
    Statement 2;  
    Statement 3;  
    .....  
    .....  
    if ( If Condition)  
        break;  
    Statement N-1;  
    Statement N;  
    Increment;  
}  
OutsideStatement 1;
```

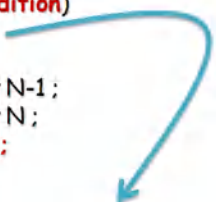


Fig 2.2 Break Statement used in While loop

Here we see that the break statement is present inside the while loop. When such statement will work out and break, then it will move the control outside the while loop.

### Check your progress 6

1. The Break Statement in Java will break:

- |               |                  |
|---------------|------------------|
| a. For loop   | c. Do while loop |
| b. While loop | d. all of these  |

---

## 2.8 The Continue Statement

---

In Java, it is seen that the Continue Statement will skip the part of loop. Similar like break statement, continue statement will not terminate the loop, but will skip the left over part of loop where the control goes back again to check the conditions. The syntax of Continue Statement is:

```
{  
//loop body  
.....
```



```
.....  
.....  
.....  
continue;  
.....  
.....  
}
```

It is studied that Continue Statement is also called as Jumping Statement which skips the Loop and Re-Executes Loop by considering new condition. This statement applies only in Loop Control Statements like For Loop, While Loop and do-While Loop. We can use such statements in many ways:

### Continue Statement Written Inside For Loop

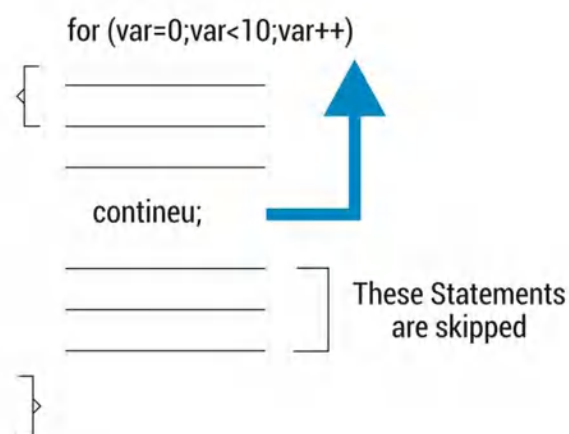


Fig 2.3 Continue statement written inside for loop

### Continue Statement Written Inside While Loop

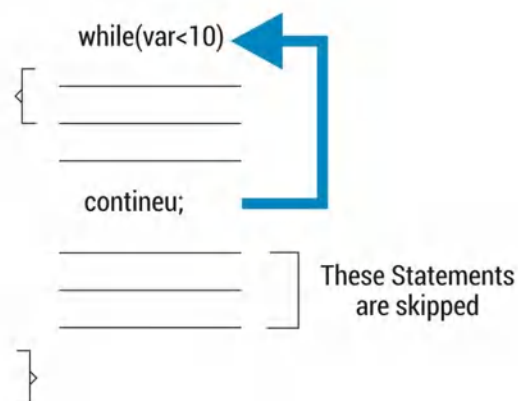


Fig 2.4 Continue Statement written inside while loop

### Check your progress 7

1. Which is not a Loop Control Statement?

- |                |                  |
|----------------|------------------|
| a. Nested loop | c. While Loop    |
| b. For Loop    | d. Do-While Loop |

---

## 2.9 The Switch Statement

---

Switch statement is used in Java where you require selecting one of several alternatives that is based on value of an integer, character, or string variable. The syntax of switch statement is:

```
switch (expression)
{
    case constant;
        statements;
        break;
    [ case constant – 2;
        statement;
        break; ] ...
    [ default;
        Statements;
        Break; ] ...
}
```

It is found that in a switch statement, the expression must evaluate to int, short, byte, or char. In this, each grouping of code lines begins with case keyword and ends with break statement. Here, coding of many case groups can be done where group starts with word case followed by constant and colon.

In such case, coding of more than one statement is possible when the value of switch expression equals to the value of constant. The last line of each case group is a break statement, which causes the entire switch statement to end. Consider an example:

```
Double commissionRate;
```

```
Switch (salesClass)
{
    Case 1:
        commissionRate = 0.02;
        break;
    Case 2:
        commissionRate = 0.035;
        break;
    Case 3:
        commissionRate = 0.05;
        break;
    default:
        commissionRate = 0.0;
        break;
}
```

It is found that a switch statement can find char data. In example shown, a char variable named sales Category is calculated for commission rates having sales categories as A, B or C.

```
double commissionRate;
switch (salesCategory)
{ case 'A': case 'a':
    commissionRate = 0.02;
    break;
case 'B':
case 'b':
    commissionRate = 0.035;
    break;
case 'C':
case c:
    commissionRate = 0.05;
    break;
default:
    commissionRate = 0.0;
    break;
}
```

### Check your progress 8

1. In switch statement, an expression evaluates to:

- a. int
- b. short
- c. byte
- d. all

## 2.10 The Bitwise Operators

In Java, it is seen that bitwise operators works on individual bits of integer that can be int value or long value. It is seen that when an operand is shorter than int, it is promoted to int before performing operations.

These operators allow us to know how integers are shown in terms of binary. In case of decimal number 3, it can be shown as 11 in binary, while in case of 5, it can be shown as 101 in binary. It is found that negative integers are stored in two's complement form.

| Operator | Name        | Example   | Result | Description                         |
|----------|-------------|-----------|--------|-------------------------------------|
| a. & b   | and         | 3 & 5     | 1      | 1 if both bits are 1.               |
| a   b    | or          | 3 5       | 7      | 1 if either bit is 1.               |
| a ^ b    | xor         | 3 ^ 5     | 6      | 1 if both bits are different.       |
| ~a       | not         | ~3        | -4     | Inverts the bits.                   |
| n<<p     | left shift  | 3<<<2     | 12     | Shifts bits of n left p positions.  |
| n>>p     | right shift | 5>> 2     | 1      | Shifts bits of n right p positions. |
| n>>>p    | right shift | -4 >>> 28 | 15     | Shifts bits of n right p            |

### Check your progress 9

1. Which among the following is not a bitwise operator?

- a. &
- b. &=
- c. |=
- d. <=

---

## 2.11 Skills Check

---

In Java skills are applied to work and solve problems related to looping. There are certain loop statements such as FOR loop, While loop, Do..While loop where knowledge is used to judge which loop will be executed first.

In programming, looping is common which will make us to reach on certain results. There is several loop statement which continuously works till reaches to particular solution.

### Check your progress 10

1. In programming, looping is common which will make us to reach on certain results.
  - a. True
  - b. False

---

## 2.12 Exercises

---

### Example 1:

Program to display triangle of \* using nested for loop

```
class NestedForLoopDemo
{
    public static void main(String args[])
    {
        for(int i = 1; i <=5 ; i++)
        {
            for(int j = 1; j <= i; j++)
            {
                System.out.println("* ");
            }
            System.out.println("");
        }
    }
}
```

```
    }  
    }  
  
Output
```

```
*  
  
* *  
  
* * *  
  
* * * *  
  
* * * * *
```

### Example 2 :

Program to print tables

```
class NestedWhileLoop
```

```
{
```

```
    public static void main(String args[])
```

```
    {
```

```
        int i=1, j=1;
```

```
        System.out.println("Tables");
```

```
  
        while(i <= 2) // change to 2 to 10 or 20 as many tables user want
```

```
        {
```

```
            while(j <= 10)
```

```
            {
```

```
                System.out.println(i + " * " + j + " = " + (i*j));
```

```
                j++;
```

```
            }
```

```
            i++;
```

```
            System.out.println("");
```

```
            System.out.println("");
```

```
        }
```

```
}  
}
```

Output

Tables

$$1 * 1 = 1$$

$$1 * 2 = 2$$

$$1 * 3 = 3$$

$$1 * 4 = 4$$

$$1 * 5 = 5$$

$$1 * 6 = 6$$

$$1 * 7 = 7$$

$$1 * 8 = 8$$

$$1 * 9 = 9$$

$$1 * 10 = 10$$

$$2 * 1 = 2$$

$$2 * 2 = 4$$

$$2 * 3 = 6$$

$$2 * 4 = 8$$

$$2 * 5 = 10$$

$$2 * 6 = 12$$

$$2 * 7 = 14$$

$$2 * 8 = 16$$

$$2 * 9 = 18$$

$$2 * 10 = 20$$

More about  
Control  
Statements  
and Operators

---

## 2.13 Let Us Sum Up

---

In this unit we have learnt that Java operators are used to carry out primitive data types that can be categorised as unary, binary or ternary.

It is seen that in case of looping, condition becomes worked out first which can be true or false, and when it is constant then any non-zero value of c will be true and zero remains false.

If the Boolean expression is true, the flow of control jumps back up to do, and the statements in the loop execute again.

In Java, the Break Statement applies to bring about loop control statements. Such type statement will break for Loop, while Loop and do-while loop.

Switch statement is used in Java where you require selecting one of several alternatives that is based on value of an integer, character, or string variable.

---

## 2.14 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-b)

**Check your progress 2**

**Answers:** (1-d)

**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-a)

**Check your progress 5**

**Answers:** (1-b)

**Check your progress 6**

**Answers:** (1-d)



**Check your progress 7**

**Answers:** (1-a)

**Check your progress 8**

**Answers:** (1-d)

**Check your progress 9**

**Answers:** (1-d)

**Check your progress 10**

**Answers:** (1-a)

---

## 2.15 Glossary

---

1. **Looping** - it is a length of line which goes again and again over making an opening
2. **Programming** - It is a technique of writing codes in order to work on computer.

---

## 2.16 Assignment

---

Compute the output of the program?

```
public class Test
{
    public static void main(String args[])
    {
        int i = 1, j = 0;
        switch(i)
        {
            case 2: j += 6;
            case 4: j += 1;
            default: j += 2;
            case 0: j += 4;
        }
        System.out.println(j + " ");
    }
}
```

```
    }  
}
```

---

## 2.17 Activities

---

Compare FOR and While..do loop according to their syntax.

---

## 2.18 Case Study

---

Design a nested if-else statement which will show character grade to percentage of mark as:

|         |              |
|---------|--------------|
| Grade A | 70 or above  |
| Grade B | 60-69        |
| Grade C | 50-59        |
| Grade D | 40-49        |
| Grade E | 30-39        |
| Grade F | less than 30 |

---

## 2.19 Further Readings

---

1. Learning Programming by Peter Norvig's
2. Approach programming with a more positive by P.Brian.Mackey

---

## **UNIT 3:      CREATING CLASSES**

---

### **Unit Structure**

- 3.0   Learning Objectives**
- 3.1   Introduction**
- 3.2   The General form of a Class**
- 3.3   Creating Simple Classes**
- 3.4   Adding Constructors**
- 3.5   Constructor Overloading**
- 3.6   The this Keyword**
- 3.7   Instance Variables and Methods**
- 3.8   Static Variables and Methods**
- 3.9   Local Variables and its Scope**
- 3.10   Method Overloading**
- 3.11   Argument Passing**
- 3.12   Wrapper Classes**
- 3.13   System Class**
- 3.14   Garbage Collection**
- 3.15   Skills Check**
- 3.16   Exercises**
- 3.17   Let Us Sum Up**
- 3.18   Answers for Check Your Progress**
- 3.19   Glossary**
- 3.20   Assignment**
- 3.21   Activities**
- 3.22   Case Study**
- 3.23   Further Readings**

---

## 3.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- Adding Constructors
- Constructor Overloading
- This keyword
- Instance variables methods
- Static variables methods
- Local variables

---

## 3.1 Introduction

---

After having basic knowledge about Java, you can now have basics of Java programming experience which you can apply to write your own classes. In this unit, you will receive detailed information about how you can write and define classes which could be in terms of declaring member variables, methods as well as constructors. Also this unit will make you learn to how to apply your classes to design and develop objects along with using certain objects what you can develop.

---

## 3.2 General form of a Class

---

In Java, classes are created using keyword class in which you can define objects. A class contains three types of items:

- Variables
- Methods
- Constructors

It is found that variables describe its state, whereas class carries static and instance variables. In this, method shows the logic that describes the behaviour of a class. As seen, class contains static and instance methods. Where it is seen that a constructor will initialize the state of new instance of class. We see that the general form of a Class in Java can be:

```
class clsName  
{
```

```
// instance variable declaration
type1 varName1 = value1;
type2 varName2 = value2;
:
:
typeN varNameN = valueN;
class clsName
{
// instance variable declaration
type1 varName1 = value1;
type2 varName2 = value2;
:
:
typeN varNameN = valueN;
// Constructors
clsName (cparam 1)
{
// body of constructor
}
:
:
clsName(cparamN)
{
// body of constructor
}
// Methods
rType1 methodName1(mParams1)
{
// body of method
```

```
}  
}
```

In the above syntax:

- Class shows declared class named `clsName` which follows java naming conventions for identifiers.
- Instance variables named `varName1` by `varNameN` as normal variable declaration syntax.
- Constructors carry similar name as class having no return values.
- Methods named `methName1` declares `methNameN` having return type methods as `rType1` by `rTypeN` and `mParamN`.

### Check your progress 1

1. In the example, the Stock is called as:

```
Class Stock {  
    Public commodity;  
    Public price;  
    Public void buy (int no_of commodity) {}  
    Public boolean sale () {}  
}
```

- |           |                  |
|-----------|------------------|
| a. Class  | c. Methods       |
| b. Fields | d. None of above |

---

## 3.3 Creating Simple Classes

---

We can write simple class in Java. Consider an example of simple class Java program which will calculate Area of Rectangle as shown:

```
Class Rectangle {  
  
    double length;  
  
    double breadth;  
  
}  
  
// This class declares an object of type Rectangle.
```

```
Class RectangleDemo {  
    Public static void main (String args[]) {  
        Rectangle myrect = new Rectangle();  
        double area;  
        // assign values to myrect's instance variables  
        myrect.length = 20;  
        myrect.breadth = 10;  
        // Compute Area of Rectangle  
        Area = myrect.length * myrect.breadth;  
        System.out.println("Area is "+area);  
    }  
}
```

If we run this program we will see that the output will be:

Area is 200.0

While designing such program, we have followed following steps:

**Step 1: Class Creation / Declaration:**

Consider following class –

```
Class Rectangle {  
    double length;  
    double breadth;  
}
```

- Class shows new type of data.
- Here rectangle is new data type.
- Rectangle is used to declare objects of type Rectangle.
- Class declaration creates template not actual object.

**Step 2: Creation of Object**

```
Rectangle myrect = new Rectangle();
```

Actually, the memory is allotted to object as soon as execution of statement that will create instance of class “Rectangle” having instance name as “myrect“. In fig 3.1, we see the use of Constructor

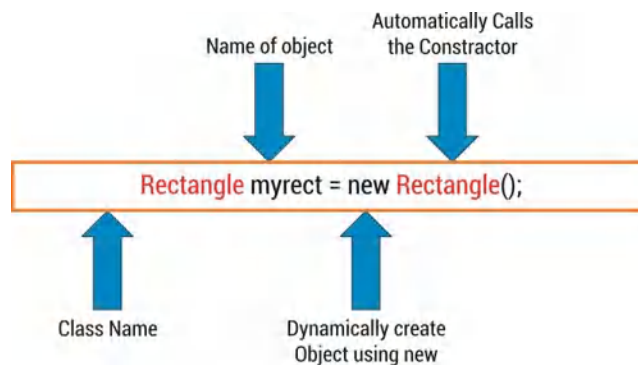


**Fig 3.1 Creating Objects**

**Step 3:** Accessing Instance Variables of Object/Instance with DOT Operator

```
myrect.length = 20;
```

```
myrect.breadth = 10;
```

It is found that each Instance/Object have its own copy of instance variables that can be length and breadth.

### Check your progress 2

1. While creating a simple class program, you have to consider:
  - a. Class creating
  - b. Object creation
  - c. Instance variable accessing
  - d. all of above

## 3.4 Adding Constructors

In Java, while discussing about classes, you need to involve with constructor which is present in a class. If you fail to write a constructor for class, then Java compiler will automatically create a default constructor for such class. It is seen that each time a new object is created. Constructor carries similar name as class, so a class can have more than one constructor. It is studied that constructors are methods in Java that are initialises objects. It has similar name as class they called when an object of class is created. Consider simple Java constructor class:

```
Class Programming {
```



```
// constructor method
Programming() {
    System.out.println("Constructor method called.");
}

Public static void main (String[] args) {
    Programming object = new Programming(); // creating object
}
}
```

On execution of this program we will see its output as:

E:\ Java>javac Programming.java

E:\ Java > java Programming

Constructor method called.

E:\ Java>

A simple example of a constructor is shown below:

```
Public class Puppy{
    Public Puppy() {
    }

    Public Puppy (String name) {
//This constructor has one parameter, name.
    }
}
```

### Check your progress 3

1. Identify the return type of Constructor from the following.

- a. int
- b. float
- c. void
- d. None of above

---

## 3.5 Constructor Overloading

---

Constructor overloading is a mechanism in Java where a class can have a number of constructors having different parameter lists. To distinguish, it is the duty of compiler to differentiate the constructors by considering number of parameters in list and its type. Similarly like other methods, java constructor also overloads as it develops several constructors in desired class. Consider a constructor overloading example:

```
Class Language {  
    String name;  
    Language() {  
        System.out.println("constructor method called.");  
    }  
    Language (String t) {  
        Name = t;  
    }  
    Public static void main (String[] args) {  
        Language cpp = new Language ();  
        Language java = new Language ("Java");  
        Cpp.setName("C++");  
        Java.getName();  
        Cpp.getName();  
    }  
    Void setName(String t) {  
        name = t;  
    }  
    Void getName() {  
        System.out.println("Language name: " +name);  
    }  
}
```

**The output of this program will be:**

```
E: \Java>javac Language.java  
E: \Java > java Language  
Constuctor method called.  
Language name: Java  
Language name: C++  
E: \Java  
Language name:
```

Consider another overloading program where you will find an overloading of constructor that generates an output.

```
class Student5{
    int id;
    String name;
    int age;
    Student5(int i:String n){
        id = i;
        name = n;
    }
    Student5(int i:String n:int a){
        id = i;
        name = n;
        age=a;
    }
    void display0{System.out.println(id+" "+name+" "=age);
    }
public static void main(String arg[]){
    Student5 s1 = new Student5(111,"Nishit");
    Student5 s2 = new Student5(222,"Rohit",25);
    s1.display();
    s2.display();
    }
}
```

If we run the above program we will get an output as:

```
111  Mohit      0
222  Rohit      25
```

### Check your progress 4

1. Which among the following can be overloaded?

- |            |                 |
|------------|-----------------|
| a. Methods | c. Constructors |
| b. Class   | d. Fields       |

---

## 3.6 The this Keyword

---

In Java, this keyword serves as a reference variable which mostly refers to present or current object. There are many usage of java this keyword as:

- It refers current class instance variable.
- This () calls upon current class constructor.
- It calls current class method.
- It passes an argument in method call.
- It passes an argument in constructor call.
- It returns current class instance.

Consider a program where you will find difference in outputs when this keyword is not used as shown:

```
class Student10 {  
    int id;  
    String name;  
    Student 10(int id, String name) {  
        id = id;  
        name = name;  
    }  
    void display() {System.out.println(id+" "-name);  
    }  
    public static void main(String args[]){  
        Student10 s1 = new Student10(111,"Nishit");  
        Student10 s2 = new Student10(321,"Rohit");  
        s1.display();  
        s2.display();  
    }  
}
```

The output will be as shown, when no "this keyword" are used:

0 null

0 null

The output will be as shown, when we use "this keyword":

111 Mohit

222 Rohit

### Check your progress 5

1. What will be the correct option, when "these keywords" are displayed?

```
Class Rectangle {  
    int length;  
    int breadth;  
    void setDiamentions (int ln, int br)  
    {  
        this.length = ln;  
        this.breadth = br;  
    }  
}  
  
Class RectangleDemo {  
    Public static void main(String args[]) {  
        Rectangle r1 = new Rectangle();  
        r1.setDiamentions(20,10);  
        System.out.println("Length of Rectangle :"+r1.length);  
        System.out.println("Breadth of Rectangle :"+r1.breadth);  
    }  
}
```

- |                             |                    |
|-----------------------------|--------------------|
| a. Length of Rectangle: 20  | c. Both a and b    |
| b. Breadth of Rectangle: 10 | d. Neither a nor b |

---

## 3.7 Instance Variables and Methods

---

Instance Variables in Java are variables that are defined without using static keyword. They are object specific and present outside any method declaration as their values are occurrence of any particular and not shared among instances. Consider an example shown:

Class Page

```
{  
    Public String page Name;  
    // instance variable with public access  
    Private int pageNumber;  
    // instance variable with private access  
}
```

### Rules for Instance variable

- Applies to any of four access level
- Marked as final
- Marked as transient
- Cant marked as abstract
- Cant marked as synchronized
- Cant marked as strict
- Cant marked as native
- Cant marked as static

### Instance Method

In Java, an instance method is linked with and works upon an object. With this result, it is necessary to develop an instance of that class so as to call upon such method. The syntax of instance method is shown:

```
reType funcName(paramList)  
{  
    //Body of this method  
}
```

Syntax to call instance method

```
objRef.funcName(args);
```

Here:

objRef: object reference variable

funcName: is name of method

args: optional arguments

**For example,**

```
int length()
String substring(int start)
```

Consider an example of instance method and instance variable as shown:

```
Class Circle
{
    //Instance Variables
    double x;
    double y;
    double radius;
    //Instance method
    void scale(double a)
    {
        radius *=a;
    }
}
```

In the above example, class circle will show instance variables and instance method.

### Check your progress 6

1. The instant variables can be marked as:

- |                 |              |
|-----------------|--------------|
| a. synchronized | c. native    |
| b. strict       | d. transient |

---

## 3.8 Static Variables and Methods

---

### Static Variables

Static variable is a variable which belongs to class and not to instance. It gets initialized first at the start before initialization of any instance variables. In this type of variable, only a single copy will be shared by every instances of class and is not accessed directly by class name and requires no object. The syntax of the Static variable is:

<class-name>.<variable-name>

Consider an example of static variable:

```
Public class MyStatic
{
    int var;
    static int var2;

    public static int methodStatic()
    {
        .....
    }
}
```

In the above program, 2 variables are present, 1 is instance and other is static. To obtain multiple objects of 'MyStatic' class, we see that instance variables are placed in separate space for storage whereas static variable remains common to every objects of MyStatic class with no separate space given. It is found that the java object belongs to MyStatic class will alter static variable, as there is no use to creating object of a class to modify static variables. It is found that you can alter the class name directly.

In java, every instance methods can directly be accessed using such static variables that are initialized first and then instance variables.

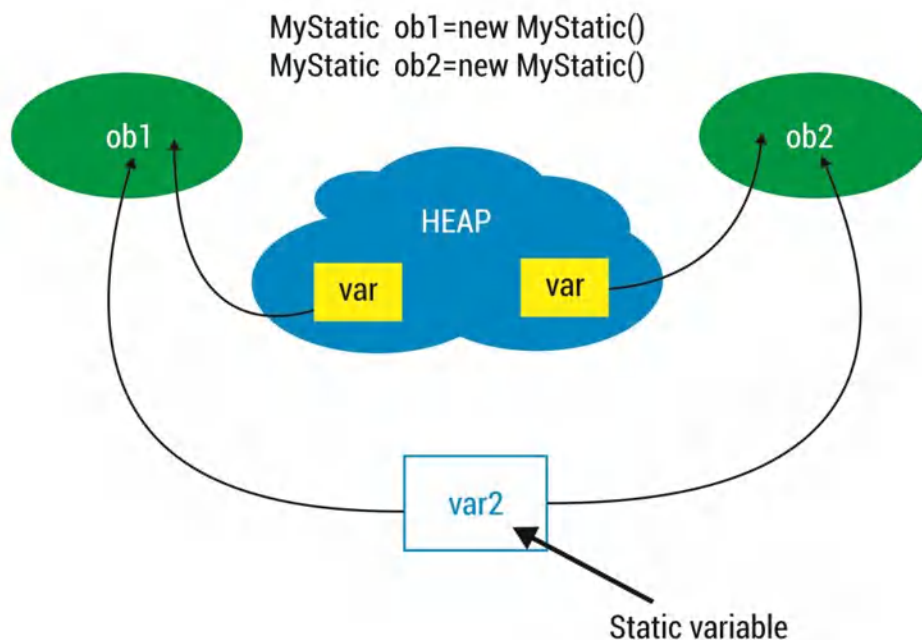


Fig 3.2 Static variable



## Static Methods

Static method is that which belongs to class and not to object and can access only static data without the need of instance variables. This method can only call other static methods and not non-static methods. Such type of method can access directly by class name and requires no object. The syntax of static method is:

`<class-name>.<method-name>`

In Static methods, you can call all static methods of java class by class name and

`MyStatic.methodStatic();`

It is seen that there is an requirement to create object for MyStatic class as you can call directly as shown in static variable. Generally, these static methods access static variables. It is seen that every instance methods of a class gets access such static methods as compared to static methods, that can't be access any instance types directly.

### Check your progress 7

1. Which is incorrect about static methods?

- |                            |                               |
|----------------------------|-------------------------------|
| a. It belongs to class     | c. It can access static data  |
| b. It belongs to an object | d. It uses instance variables |

---

## 3.9 Local Variables and its Scope

---

In explaining about a class, we see that a local variable is a technique in which the parameters obtained gets worked out that are used to store intermediate results. These variables are declared in a method and will be out at the time of its working. Such variables use different methods with similar name and have no relationship with each other. The syntax of Local Variable is:

`Type varName = value;`

Consider an example of a local variable which will describe its scope

Class Sample

{

    Static ctr = 0;

```
int i=100;
void display1()
{
    System.out.println("Before");
    System.out.println("ctr = " + ctr);
    System.out.println("i="+i);

    int ctr = 2, I =200;

    System.out.println(" ");
    System.out.println("After");
    System.out.println("ctr = "+ctr)
}
void display2()
{
    System.out.println("Another method");
    System.out.println("ctr = " +ctr);
    System.out.println("i = "+i );
}
}
class LocalVariables
{
    public static void main(String args[])
    {
        Sample sObj = new Sample();
        sObj.display1();
        System.out.println("");
        sObj.display2();
    }
}
```

From the output of this program we see that there are two results once before execution and second after execution.

### **Before**

ctr = 0

i = 100

### **After**

ctr = 2

**Check your progress 8**

1. Which is correct about local variables?
  - a. It is declared inside a method
  - b. It exist at the time of its execution
  - c. It carries no relationship among each other
  - d. All of these

---

### 3.10 Method Overloading

---

Method overload exists when a class having many methods with similar name but having different parameters, then under such situation the method will overload. Under this situation, if you have to do only single operation with similar name, then you will find that the method increases the readability of a program. If you have to perform addition with required numbers but you have n number of arguments and when you write method as a=int,int in case of two parameters and b=int,int,int in case of three parameters, then you will find difficulty and also programmers can't understand the behaviour of such method. This will happen because its name differs. Under such scenario, we do method overloading to have program quickly.

There are two ways in which you can overload the method:

- It can be done by changing number of arguments
- It can be done by changing data type

Consider two examples shown below where the Method of Overloading is explained by changing the number of arguments and by changing data type of argument.

**Example of Method Overloading by changing the no. of arguments**

Here you will find two overloaded methods, first sum method performing addition of two numbers and another is sum method performing addition of three numbers.

```
class Calculation{  
    void sum(int a,int b){System.out.println(a+b);}  
}
```

```
void sum(int a,int b,int c){System.out.println(a+b+c);}
public static void main(String args){
    Calculation obj=new Calculation();
    obj.sum(10,10,10);
    obj.sum(20,20);
}
```

If you run the above program you will find that its output will be:

30

40

### **Example of Method Overloading by changing data type of argument**

Here, you will see that two overloaded methods exist that differ in data type. First will be the sum method which gets two integer arguments and another will be the sum method that gets two double arguments.

```
class Calculation2{
    void sum(int a,int b){System.out.println(a+b);}
    void sum(double a,double b){System.out.println(a+b);}
    public static void main(String args()){
        Calculation2 obj=new Calculation2();
        obj.sum(10.5,10.5);
        obj.sum(20,20);
    }
}
```

When you run this program, you will find the output as:

21.0

40

**Check your progress 9**

1. What will be the output?

```
class OverloadingCalculation3 {
    void sum(int a,long b){System.out.println("a method invoked");}
    void sum(long a,int b){System.out.println("b method invoked");}
    public static void main(String args[]){
        OverloadingCalculation3 obj=new OverloadingCalculation3();
        obj.sum(20,20);//now ambiguity
    }
}
```

- |       |                       |
|-------|-----------------------|
| a. 20 | c. 10                 |
| b. 40 | d. Compile Time Error |

---

### 3.11 Argument Passing

---

In Java, arguments are the list of parameters which gets passed in Java Programme during the start of any programme. In this, if arguments.length > 0, then it will check if any arguments has been provided.

It is found that when you have pass-by-value parameter, then a copy of argument is kept safe in memory location which is given for formal parameter. Under such circumstances, any changes made to formal parameter in method have no effect on the value of argument which is applied back in calling method.

Also, when a parameter is pass-by-reference, then the memory address of an argument is passed to method which makes formal parameter for argument. Here the change made to formal parameter within the method gets reflected in value of an argument in case when a control is back to calling function.

Normally, it is found that all primitives are pass-by-value period into a method making a copy of primitive that carried out in such method. Hence, the value of copy will alter inside the method keeping the original value unchanged. Consider an example of the program:

```
public class TestPassPrimitive {
    static void doSomething(int m) {
        m=m+2;
        System.out.println("The new value is " +m ".");
    }
}
```

```
public static void main(String[] args) {  
    int m=5;  
    System.out.println("Before doSomething, m is " +m ".");  
    doSomething(m);  
    System.out.println("After doSomething, m is "+m+ ".");  
    }  
}
```

If we run such program, we see that the output will be shown as:

Before doSomething, m is 5.

The new value is 7.

After doSomething, m is 5.

### Check your progress 10

1. Arguments is a list of:

a. Parameters

c. Array

b. Programs

d. None of above

---

## 3.12 Wrapper Classes

---

The wrapper classes are part of java.lang package, which is imported by default into all Java programs. The wrapper classes in java serves two main functions:

- Providing mechanism to ‘wrap’ primitive values in an object to provide activities for objects.
- Providing assortment of utility functions for primitives such as converting primitive types to bases as binary, octal or hexadecimal

It is found that these statements will show difference among primitive data type and object of wrapper class:

```
int x = 35;
```

```
Integer y = new Integer(44);
```

Here, the statement will declares int variable named x and initializes value 35. Also, the object gets initialized with value 44 with reference to object assigned to object variable y.

It is seen that in Java, there appears to be a wrapper class for every primitive data

type. Such class will summarize particular value for primitive data type. In wrapper class we see that int is for Integer, float is for Float and so on.

### Types

| Primitive data type | Wrapper class |
|---------------------|---------------|
| byte                | Byte          |
| short               | Short         |
| int                 | Integer       |
| long                | Long          |
| float               | Float         |
| double              | Double        |
| char                | Character     |
| boolean             | Boolean       |

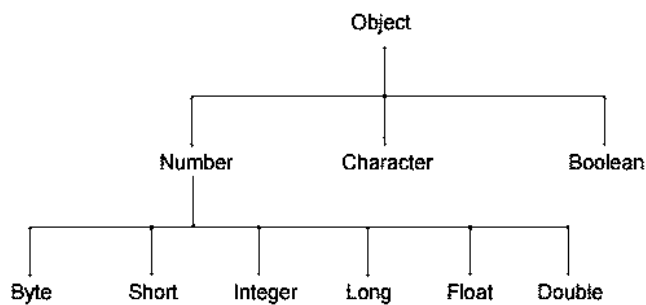


Fig 3.3 Wrapper Classes

### Check your progress 11

1. The wrapper classes in java are:

- |                 |              |
|-----------------|--------------|
| a. private      | c. immutable |
| b. serializable | d. finale    |

---

## 3.13 System Class

---

In Java, we see that a System Class:-

- Is a core class.
- Shows number of method and class fields.
- Is available in java.lang package.

System class carries Standard input, Standard output, error stream output, loading files and libraries that can be access to external properties and environment variables with utility method for quickly copying array.

### Unique Features in System Class:-

- It can directly interact with device
- It has all library in regards to input and output Stream
- It takes input and out specific buffer from Operating System.
- It carries reference of input and output stream which is static when system class loads into memory
- It interacts automatically with operating system

The internal structure of System class with method and variables:-

```
public final class System extends Object
{
    public static void void registerNative();
    static
    {
        registerNative();
    }
    public final static InputpuStream in=null;
    public final static PrintStream out=null;
    public final static PrintStream err = null;
    public static void setOut(PrintStream out){}
    public static void setErr(PrintStream err){}
    public static void setIn(InputStream in){}

    public static Console console() {}
    public static native long nanoTme();
}
```



During the class loading, the system class is loaded initially in static block where call register Natives() method will call to JNI(java Native Interaction). The JNI having inbuilt code will interact with Operating System and loads all library specified by JVM. Further, the JVM will receive all static reference variable such as public final static Print Stream out=null; The JVM will read the corresponding code and will instruct JNI to get Output Buffer from Operating System, With this, the JNI run its method and shows Output Stream Buffer

### Check your progress 12

1. Which is not a feature of Java System Class?
  - a. It will not interact directly with device
  - b. It carries all library having input and output Stream
  - c. It takes input/output specific buffer from Operating System
  - d. All of above

---

## 3.14 Garbage Collection

---

In Java, garbage collection refers to mechanism which describes about heap memory that identifies which object is in use and which not and further will delete unused objects. In used object, some part of program maintains a pointer to that object, whereas in case of unused object, there will be no referenced with any part of a program. Hence the memory used by unreferenced object gets broken.

In Java, it is seen that the mechanism of de-allocating of memory is handled directly by garbage collector who has certain process steps as shown:

### Step 1: Marking

The initial step is marking where garbage collector will decide which pieces of memory are in use and which are not.

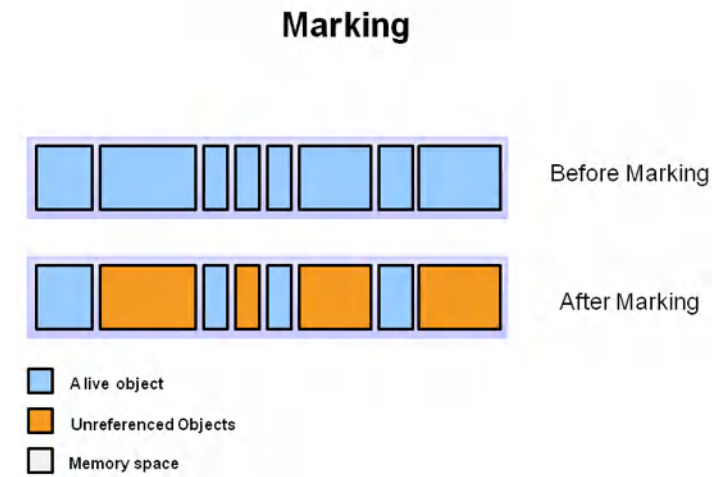


Fig 3.4 Marking process in Garbage

**Step 2: Normal Deletion**

Normal deletion will remove unreferenced objects and leaves behind referenced objects and pointers to free space.

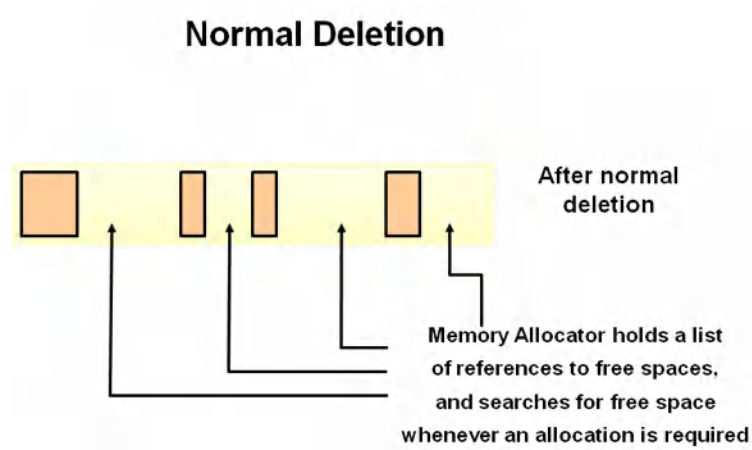


Fig 3.5 memory allocator

To improve the performance apart from adding deleting unreferenced objects, you can also compact left over part of referenced objects. With this, the new memory allocation becomes easier and faster as shown in fig 3.6.

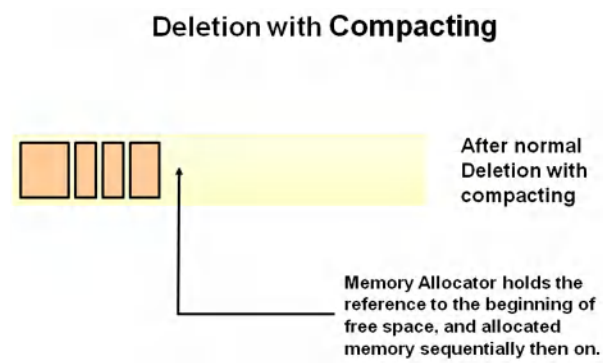


Fig 3.5 memory deletion

Now, we see that any new objects are allotted to eden space where both survivor spaces start out empty.

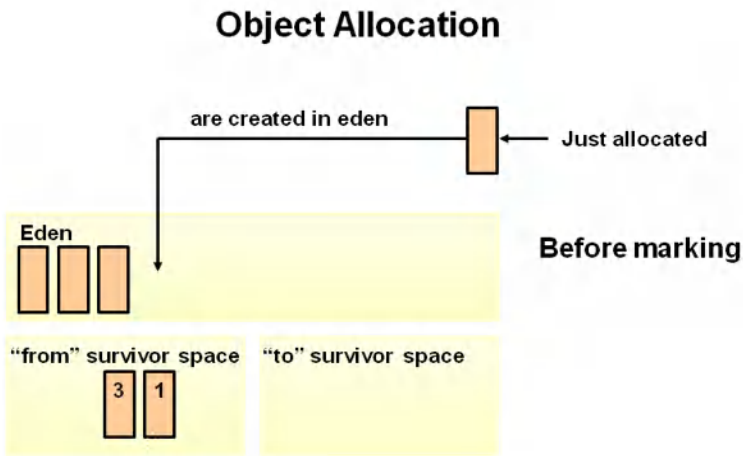


Fig 3.6 Object allocation

When the eden space fills up, a minor garbage collection is triggered.

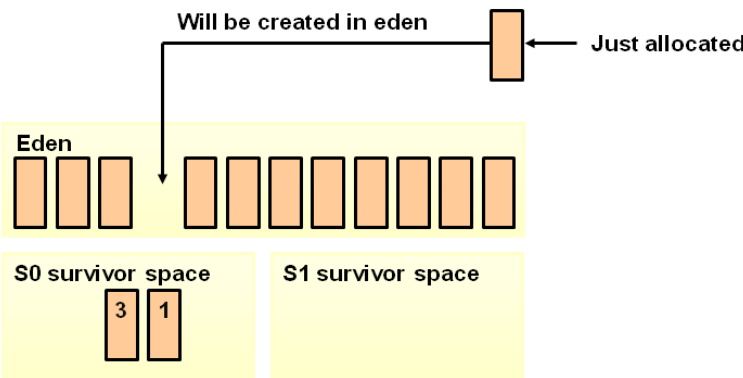


Fig 3.7 Filling of eden space

Now it is seen that the referenced objects are shifted to first survivor space. Here the unreferenced objects get deleted on clearing of eden space.

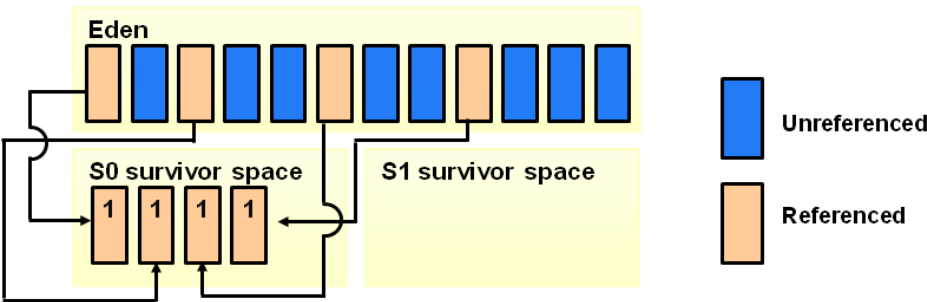


Fig 3.8 Copying of referenced objects

We see that after such operation, the unreferenced objects are deleted and referenced objects are moved to survivor space. Hence, they are moved to the

second survivor space (S1). It is seen that objects from last minor GC on first survivor space (S0) carries age incremented and moves toward S1.

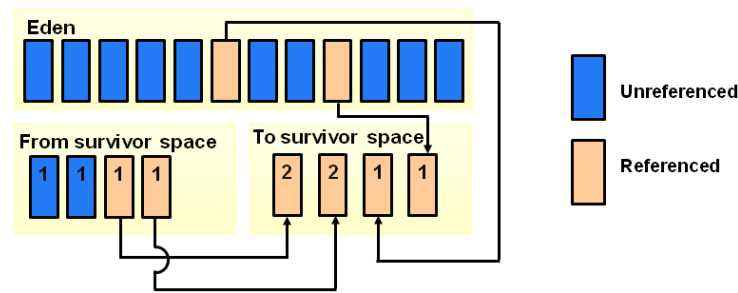


Fig 3.9 Object Aging

It is found that the survivor spaces switch and referenced objects are moved to S0. Eden and S1 are cleared.

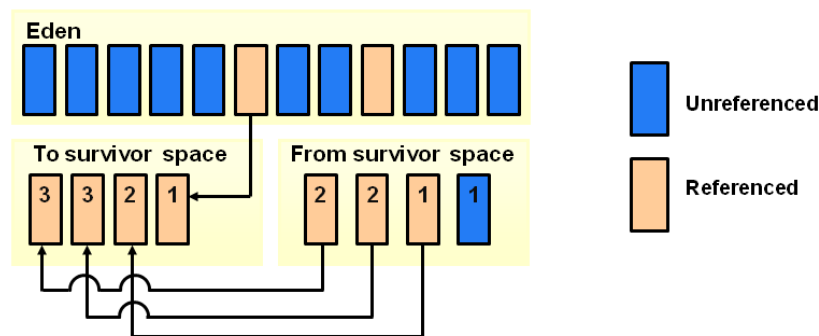


Fig 3.10 adding aging

After a minor GC, when aged objects reach a certain age threshold they are promoted from young generation to old generation.

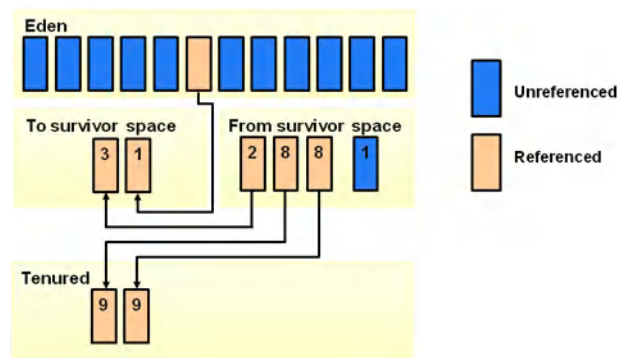


Fig 3.11 Promotion

With working of minor GCs to occur objects that continues to promote old generation space.

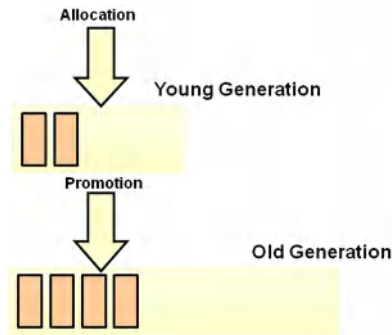


**Fig 3.12 Promotion**

### Check your progress 13

1. Which is correct in case of garbage collector?
  - a. It describes about heap memory
  - b. It describes about the status of object
  - c. It deletes unused objects
  - d. All of these

## 3.15 Skills Check

In Java lots of skills are used to study about Java classes which gets created using keyword. The concept of three classes as Variables, Methods and Constructors are easy to understand.

The skills used in studying simple class program can be applied to highlight class name, name of object, object creation using new and creating call using constructor.

The skills involved in solving and studying various constructor classes has similar name as class where class have more than one constructor. The skills in solving constructor overloading mechanism explains about number of constructors having different parameter lists which gets distinguishes by considering number of parameters in list and its type.

### Check your progress 14

1. In Java lots of skills are used to study about Java classes which gets created using keyword
  - a. True
  - b. False

---

## 3.16 Exercises

---

### Example 1:

Program to create a class named as Square having variables height and width, overloaded constructors assigning value to variables

```
class Square
{
    int height;
    int width;

    Square()
    {
        height = 0;
        width = 0;
    }
    Square(int side)
    {
        height = width = side;
    }

    Square(int sideh, int sidew)
    {
        height = sideh;
        width = sidew;
    }
}
```

```
class ImplSquare
{
```

```
public static void main(String args[])
{
    Square sObj1 = new Square();
    Square sObj2 = new Square(5);
    Square sObj3 = new Square(2,3);

    System.out.println("Variable values of object1 : ");
    System.out.println("Object1 height = " + sObj1.height);
    System.out.println("Object1 width = " + sObj1.width);
    System.out.println("");

    System.out.println("Variable values of object2 : ");
    System.out.println("Object2 height = " + sObj2.height);
    System.out.println("Object2 width = " + sObj2.width);
    System.out.println("");

    System.out.println("Variable values of object3 : ");
    System.out.println("Object3 height = " + sObj3.height);
    System.out.println("Object3 width = " + sObj3.width);
}
}
```

### Output

Variable values of object1:  
Object1 height = 0  
Object1 width = 0

Variable values of object2:  
Object height = 5  
Object width = 5

Variable values of object3:  
Object height = 2  
Object width = 3

---

### 3.17 Let Us Sum Up

---

In this unit we have seen that Java classes are created using keyword class where you can define objects. It is seen that a class contains three types of items Variables, Methods and Constructors.

It is studied that a simple class program can be designed where we will highlight class name, name of object, object creation using new and creating call using constructor.

It is found that constructor in class carries similar name as class, so class can have more than one constructor. It is basically methods in Java which initialises objects.

It is studied that constructor overloading is a mechanism where a class can have many number of constructors having different parameter lists that gets distinguishes constructors by considering number of parameters in list and its type.

---

### 3.18 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-a)

**Check your progress 2**

**Answers:** (1-d)

**Check your progress 3**

**Answers:** (1-d)

**Check your progress 4**

**Answers:** (1-c)

**Check your progress 5**

**Answers:** (1-c)



**Check your progress 6**

**Answers:** (1-d)

**Check your progress 7**

**Answers:** (1-b)

**Check your progress 8**

**Answers:** (1-d)

**Check your progress 9**

**Answers:** (1-d)

**Check your progress 10**

**Answers:** (1-a)

**Check your progress 11**

**Answers:** (1-d)

**Check your progress 12**

**Answers:** (1-a)

**Check your progress 13**

**Answers:** (1-d)

**Check your progress 14**

**Answers:** (1-a)

---

## 3.19 Glossary

---

1. **Block** - It is a group of consecutive statements having similar indentation.
2. **Body** - It is a block which is present in a compound statement that follows the header.

3. **Nesting** - In programming, it is the location of one program structure inside another
4. **Boolean expression** - It is an expression which shows the result as true or false.
5. **Comparison operators** - These are operators such as ==, !=, >, <, >=, and <= which compares two values.

---

### 3.20 Assignment

---

Explain the principle of constructor overloading?

---

### 3.21 Activities

---

Discuss in your group about garbage collection.

---

### 3.22 Case Study

---

In the program shown, can the program be there in any.java file? Comment.

```
class Test {  
    static int x;  
    int k;  
    // constructor with 2 args  
    public Test( int n, int m) {  
        x = n;  
        k = m;  
    }  
    public static void main(String[] args) {  
        Test t1 = new Test(10, 20);  
        Test t2 = new Test(30, 40);  
        System.out.print(t1.x + " ");  
        System.out.print(t1.k + " ");  
        System.out.print(t2.x + " ");  
        System.out.println(t2.k);  
    }  
}
```

---

## 3.23 Further Readings

---

1. Learning Programming by Peter Norvig's
2. Approach programming with a more positive by P.Brian.Mackey

---

## Block Summary

---

The block gives detailed knowledge to user or students in the areas of control structures and its features. The framing of control structures and statements with use of operators and loops will help to gain extra knowledge. Each topic is explained with examples and diagrams which made clear to students and gives lot of understanding about programming skills. The knowledge about different types of operators with their features and applications of control loop structures really allow students to work.

After completing this block, students will be able to know more about control structures and its features with basic operations about operators and statements. The concept of constructor and garbage collector will allow him to work on certain Java control platforms. After this block, students will get knowledge and confidence about working with loops, operators, structures and statements.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. Write short note on Java classes?
2. What is the syntax of While...do loop?
3. What is the use of this class in Java?
4. What is the idea of break statement?
5. What is FOR loop?

### **Long Answer Questions**

1. How to write a simple class program in Java?
2. Explain the features of garbage collector?
3. Classify different types of operators in Java?

Enrolment No.

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# RELATIONAL DATABASE MANAGEMENT

PGDCA 103

## BLOCK 2: DATABASE INTEGRITY AND NORMALIZATION

**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**





# RELATIONAL DATABASE MANAGEMENT

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Ms. Katherine Abraham

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



## **RELATIONAL DATABASE MANAGEMENT**

### **Contents**

#### **BLOCK 1: INTRODUCTION TO DATABASE MANAGEMENT SYSTEM AND CONCEPTUAL MODELLING**

##### **UNIT 1 INTRODUCTION TO DATABASE SYSTEM**

Basic Concepts of Data and Database, Database management system, Data Administrator and its functions, Advantages of DBMS over file systems, Applications of DBMS, Three-tier architecture of DBMS, Data Models, Components of DBMS, Overview of Languages of DBMS (DDL, DML, DCL)

##### **UNIT 2 CONCEPTUAL MODELLING**

Introduction, Basic Concept of Entity-Relationship Diagram, Relationship concept and its types (unary, binary and ternary), Cardinalities (one-one, one-many, many-one and many-many), Overview of Extended ER model, Case Study of Extended ER model

---

#### **BLOCK 2: DATABASE INTEGRITY AND NORMALIZATION**

##### **UNIT 1 DATABASE INTEGRITY CONCEPTS**

Introduction, Domain Integrity constraints, Entity Integrity constraints, Referential Integrity constraints

##### **UNIT 2 NORMALIZATION**

Introduction, Basic Concept of designing, Need of normalization, Decomposition (lossy and lossless), Functional Dependency, Full Functional Dependency, Armstrong Axioms of Functional Dependencies, Normalization Rules (1NF, 2NF, 3NF and BCNF), Examples based on normalization

---

**BLOCK 3: RELATIONAL ALGEBRA AND QUERY LANGUAGE**

**UNIT 1 RELATIONAL ALGEBRA**

Introduction, Basic concepts of relational algebra, symbols of relational algebra, operations on relational algebra, Examples based on relational algebra

**UNIT 2 QUERY LANGUAGE**

Introduction, Introduction to Structured Query Language, Data Definition Language (DDL), Data Manipulation Statement (DML), Data types in MySQL, Operators (Arithmetic, Comparison, Logical), SQL Functions, Group function (AVG, MAX, MIN, SUM and COUNT), COMMIT and ROLLBACK

**BLOCK 4: DATA RETRIEVAL SQL STATEMENT AND TYPES OF DATABASE SYSTEM**

**UNIT 1 DATA RETRIEVAL SQL STATEMENT**

Introduction, Single table query without condition, Single table query with condition, Group by clause, Orders by Clause, Self Join, Natural Join and Sub query, Examples based on SQL concepts

**UNIT 2 TYPES OF DATABASE SYSTEMS**

Introduction, Centralized Database System, Parallel Database Systems, Distributed Database Systems, Client-Server Database System



Dr. Babasaheb  
Ambedkar  
Open University

**PGDCA 103**

# **RELATIONAL DATABASE MANAGEMENT**

---

## **BLOCK 2: DATABASE INTEGRITY AND NORMALIZATION**

---

### **UNIT 1**

|                             |    |
|-----------------------------|----|
| DATABASE INTEGRITY CONCEPTS | 03 |
|-----------------------------|----|

### **UNIT 2**

|               |    |
|---------------|----|
| NORMALIZATION | 17 |
|---------------|----|



---

# **BLOCK 2: DATABASE INTEGRITY AND NORMALIZATION**

---

## **Block Introduction**

In DBMS, referential integrity explains about the database in total, where the things are arranged in way so that if a column is present in two or more tables inside the database, at that time, any change to a value in column in a single table will reflect in equivalent modification to facilitate value which exists in other tables. Normalization is a significant measurement of the database development process as; it frequently gives the appearance of real look of database and shows the working of how data are going to work together in database.

In this block, we will learn and study about Database Integrity Concepts and Normalization. We will cover the topics related to Domain Integrity constraints, Referential Integrity constraints, Entity Integrity constraints with concept and need of normalization. The student will be given with the knowledge about different rules and criteria's of normalization.

The block will focus on basic understanding about different integrity concepts and explains more on the concept of normalization and its requirement in making database. The students or programmers will get benefit while reading this block as it gives shortcuts and related examples that will clear all doubts.

## **Block Objective**

**After learning this block, you will be able to understand:**

- The concept of Domain Integrity constraints.
- Understanding Entity Integrity constraints.
- Detail about Referential Integrity constraints.
- Idea about normalization.
- Basic of Functional and Full Functional Dependency.
- Idea about Armstrong Axioms of Functional Dependencies.

Database Integrity  
and Concepts  
Normalization

## **Block Structure**

**Unit 1:     Database Integrity Concepts**

**Unit 2:     Normalization**

---

# **UNIT 1: DATABASE INTEGRITY CONCEPTS**

---

## **Unit Structure**

### **1.0 Learning Objectives**

#### **1.1 Introduction**

#### **1.2 Domain Integrity Constraints**

#### **1.3 Entity Integrity Constraints**

#### **1.4 Referential Integrity Constraints**

##### **1.4.1 Foreign Key**

##### **1.4.2 Candidate Key**

#### **1.5 Let Us Sum Up**

#### **1.6 Answers For Check Your Progress**

#### **1.7 Glossary**

#### **1.8 Assignment**

#### **1.9 Activities**

#### **1.10 Case Study**

#### **1.11 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About Integrity constraints
- Primary key in Entity Integrity constraints
- Basic of features of Referential Integrity constraints

---

## 1.1 Introduction

---

Earlier than the implementation of database tables, one ought to define the integrity restrictions. Integrity can be called as something just similar to be right as well as consistent. The data in a database should be obliged to be right in addition to behave in good condition. There are different integrities constraints like domain integrity, entity integrity, referential integrity as well as foreign key integrity.

The Entity integrity shows that in each table the primary key should have both the conditions:

1. Primary key is unique inside the table
2. Primary key column(s) has no null values.

It is seen that, referential integrity explains about the database in total, where the things are arranged in way so that if a column is present in two or more tables inside the database, at that time, any change to a value in column in a single table will reflect in equivalent modification to facilitate value which exists in other tables. It shows that in RDBMS, there should be an arrangement to take suitable performance in order to spread change in single table from a particular table to the other tables where changes occur.

---

## 1.2 Domain Integrity constraints

---

The main aim of constraints on a Database is basically to preserve the accuracy as well as integrity of data. Domain integrity explains the meaning of a suitable set of values for a feature.

In this, you define:

- data type
- length or size
- null value allowed
- value unique or not

For an attribute.

Many of DBMS integrity agree to show the output format and/or input mask for the attribute. Such explanation guarantees of a particular attribute that contains right as well as proper value in a database.

Moreover, a domain is set of probable values of an attribute. The Domain for the states as well as territories of India includes Delhi, Mumbai, Goa, Amritsar, Tibet, Northern Territory as well as New Delhi. In a database furthermore we may impede the approximations to be accessed as DL, MUM, GA, AMT, TIB, NT as well as ND. These approximations are following the set of acceptable approximations for the region.

It is observed that complete approximations accessed in the column should be from the identical region. If, the acknowledged STATE acquires region approximations as discriminated over and above the data attained into this column in the database may be solitary DL, MUM, GA, AMT, TIB, NT as well as ND. It is likely for an augmented one to accredit to allocate identical region. On account of INDIA\_theSTATE acquires equivalent region as STATE\_Name in a distinguishing database.

The barriers on a region conforming to its region name, explanation, data category, area, as well as set of appropriate values. Several regions restrictions will carry standard approximations, amount of decimal places, style as well as maximum along with minimum approximation amount. Every region is given with a single area constraint.

### Example:

A region constraint for Pune University database will show:

|                 |                                                           |
|-----------------|-----------------------------------------------------------|
| Region Name     | Person Name                                               |
| Explanation     | Name of person with First, Middle or Surname.             |
| Data Type       | Character                                                 |
| Total Length    | 40 characters                                             |
| Accepted Values | 'A'-'Z', 'a'-'z', and '-'                                 |
| Standard Value  | '' (null)                                                 |
| Style           | Xxxxxxxx (where initial letter of the Name being capital) |

In the above Pune University database, a Teacher entity contains two attributes: LName as well as FName, both sharing domain Person's Name.

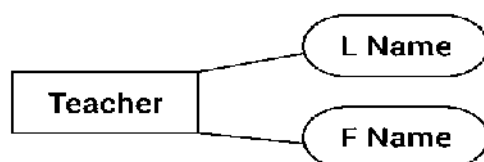


Fig 1.1domain constraint

|                  |                           |
|------------------|---------------------------|
| Attribute Name   | Lname                     |
| Domain Name      | Person's Name             |
| Description      | Last Name of Teacher      |
| Data Type        | Character                 |
| Max Length       | 40 characters             |
| Allowable Values | 'A'-'Z', 'a'-'z', and '-' |
| Default Value    | ' ' (null)                |
| Format           | Xxxxxxxx                  |

### **Not Null Constraint**

It is an SQL constraint which will make certain that every row in table will have standard finite approximation for particular column that is approximated as not null. This shows the no entry for null value whose syntax is:

[CONSTRAINT constraint name] NOT NULL

If we want to develop an employee table using Null values, then we will have:

```
CREATE TABLE employee
(id number (5),
name char (20) CONSTRAINT nm_nn NOT NULL,
dept char(10),
age number(2),
salary number(10),
location char(10)
);
```

### **Check constraint**

It is another SQL constraint which explains more of business rule placed on certain column. In this, every rows should clear using business rule. It is seen that such type of constraint is applicable for an individual column or many columns. The syntax of check constraint is:

[CONSTRAINT constraint name] CHECK (condition)

If you want to generate employee table, then you can check the constraint at column level as well as table level. While doing so you have to choose gender of a person which will make you to show as:

Check Constraint at column level:

```
CREATE TABLE employee
(id number (5) PRIMARY KEY
name char (20)
dept char(10),
age number(2),
gender char(1) CHECK (gender in ('M','F'))
salary number(10),
location char(10)
);
```

Check Constraint at table level:

```
CREATE TABLE employee
(id number (5) PRIMARY KEY
name char (20)
dept char(10),
age number(2),
gender char(1)
salary number(10),
location char(10)
CONSTRAINT gender_ck CHECK (gender in ('M','F'))
);
```

### Check your progress 1

1. The main view of constraints on Database is:

- a. save accurateness
- b. save integrity of data
- c. both a and b
- d. neither a nor b

---

## 1.3 Entity Integrity constraints

---

Entity integrity is related to ensuring every row of a table will carry a one of its kind plus non-null primary key value. It further says that every row in a table shows an individual instance of an entity type model by the table. A necessity of E F Codd in his determining paper is to show primary key of an entity or any part of it without taking null value.

Entity integrity constraint gives explanation in relation to primary keys which are not empty. Conveniently, there ought to be an appropriate value in primary key field. It turns out as primary key value which is engaged to obtain in dependent rows in a table. It seems that if null values exist for primary keys, at that time it gives you an idea that such type or rows could not be made available.

Similarly, there are definite null values apart from primary key fields which shows that an individual doesn't need to be up to date with the value meant for such field. Null value is different from zero value or space.

In a feasible design forum we are constraining to examine that Primary Keys monopoly has been assigning to every table as well as there are a set of secret approximations that can be developed to create the row. Additionally it brings about conclusive evidence that foreign keys will referral the row to affix cells for inquiries on the database.

The Entity completeness commands describes about every Entity that conveys a Primary Key which is described as abnormal identifiers at every case of the entity. Lately following the entity possesses altered to an association or table, the authentication mechanism desires to acquire every row that holds extraordinary approximation in the primary key column. No null approximations are dispensed in the Primary Key column.

### **Example:**

Consider an example of a Company that is involved in Car Rentals. We see that such company has a database where there exists a Car table where every car is assigned with proper as well as distinctive Registration \_No. The database has certain cars that have no rate because either they are broken or it is a new brand where the Rate field carries null values as described below.

The entity integrity constraints promise that a definite row in a table can be recognized.



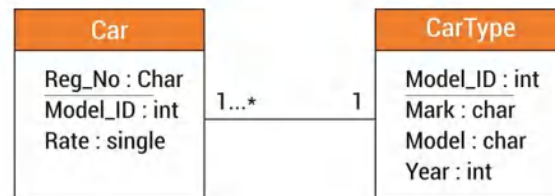


Fig 1.2 Example

| Car | reg_no  | model_id | rate    |
|-----|---------|----------|---------|
|     | ABC-112 | 1        | 45,00 € |
|     | ABC-122 | 1        | 45,00 € |
|     | ABC-123 | 1        | 47,00 € |
|     | ACC-223 | 6        | 65,00 € |
|     | ACC-224 | 6        | 65,00 € |
|     | ACC-667 | 2        | 57,00 € |
|     | BAA-441 | 5        | 35,00 € |
|     | BAA-442 | 5        | 35,00 € |
|     | BSA-224 | 3        | 45,00 € |
|     | CCE-325 | 4        |         |
|     | CCE-326 | 4        | 61,00 € |
|     | CCE-327 | 4        | 62,00 € |

← null value

| Car Type | model_id | mark    | model  | year |
|----------|----------|---------|--------|------|
|          | 1        | Ford    | Focus  | 2004 |
|          | 2        | Ford    | Mondeo | 2005 |
|          | 3        | Peugeot | 307    | 2004 |
|          | 4        | Peugeot | 407    | 2005 |
|          | 5        | Renault | Cilo   | 2004 |
|          | 6        | Renault | Laguna | 2003 |

Fig 1.3 null value

## Check your progress 2

1. Null values are:

- same as primary key fields
- same as secondary key fields
- not the primary key fields
- all

---

## 1.4 Referential Integrity constraints

---

Referential integrity is another type of database model that makes it sure to facilitate the interaction flanked by tables which remains steady. When a single table contains a foreign key to other table, then the idea of referential integrity explains with the aim that you might not put in a record inside the table which carries the foreign key except that there is an equivalent record in the linked table. Further, it is included that the performance which is called as cascading bring up to date further more cascading delete, which makes certain changes through to the linked table that are return in the primary table. The referential integrity constraint is particular among two tables in addition; it is applied so as used to preserve the steadiness that exists among rows flanked by the two tables. There are certain rules while applying, which are as follows:

### Rules:

- You cannot eliminate a record from primary table if similar records be near in a connected table.
- You cannot change a primary key value into primary table if that record has connected records.
- You cannot go through a value in foreign key field of connected table that doesn't carry on in primary key of primary table.
- On the other hand, you can place a Null value in foreign key, by representing that the records are not connected.

### 1.4.1 Foreign key

We see that an integrity constraint contains 2 foreign keys:

- cascade update related fields
- cascade delete related rows

Both such constraints affect the referential integrity constraint.

#### Cascade Update Related Fields

In such integrity constraint, you can alter primary key of row in primary table at any moment of time. Here the foreign key values are adjusted in the matching row which is held in related table. Such constraint will rule against rule 2 in the referential integrity constraints.

| Car | reg_no  | model_id | rate    |
|-----|---------|----------|---------|
|     | ABC-112 | 1        | 45,00 € |
|     | ABC-122 | 1        | 45,00 € |
|     | ABC-123 | 1        | 47,00 € |
|     | ACC-223 | 6        | 65,00 € |
|     | ACC-224 | 6        | 65,00 € |
|     | ACC-667 | 2        | 57,00 € |
|     | BAA-441 | 5        | 35,00 € |
|     | BAA-442 | 5        | 35,00 € |
|     | BSA-224 | 3        | 45,00 € |
|     | CCE-325 | 4        |         |
|     | CCE-326 | 4        | 61,00 € |
|     | CCE-327 | 4        | 62,00 € |

null value

| Car Type | model_id | mark    | model  | year |
|----------|----------|---------|--------|------|
|          | 1        | Ford    | Focus  | 2004 |
|          | 2        | Ford    | Mondeo | 2005 |
|          | 3        | Peugeot | 307    | 2004 |
|          | 4        | Peugeot | 407    | 2005 |
|          | 5        | Renault | Cilo   | 2004 |
|          | 6        | Renault | Laguna | 2003 |

Fig 1.4 referential integrity constraints

Figure 1.3, shows with an explanation about constraint along with relationship that can be available with tables where Car is marked and CarType is written. You can also introduced model\_id in this CarType table as shown above. You can alter the model\_id 1 as (Ford Focus) to model\_id 100 in CarType table, whereas the model\_ids in Car table will be altered from 1 to 100 as for cars ABC-112, ABC-122, ABC-123.

### Cascade Delete Related Rows

In such integrity constraint, you can remove a row located in the primary table anytime. In this, the matching rows are directly removed in the associated table. The result is that such constraint will rule against rule 1 placed in referential integrity constraints.

| Car | reg_no  | model_id | rate    |
|-----|---------|----------|---------|
|     | ABC-112 | 1        | 45,00 € |
|     | ABC-122 | 1        | 45,00 € |
|     | ABC-123 | 1        | 47,00 € |
|     | ACC-223 | 6        | 65,00 € |
|     | ACC-224 | 6        | 65,00 € |
|     | ACC-667 | 2        | 57,00 € |
|     | BAA-441 | 5        | 35,00 € |
|     | BAA-442 | 5        | 35,00 € |
|     | BSA-224 | 3        | 45,00 € |
|     | CCE-325 | 4        |         |
|     | CCE-326 | 4        | 61,00 € |
|     | CCE-327 | 4        | 62,00 € |

null value

| Car Type | model_id | mark    | model  | year |
|----------|----------|---------|--------|------|
|          | 1        | Ford    | Focus  | 2004 |
|          | 2        | Ford    | Mondeo | 2005 |
|          | 3        | Peugeot | 307    | 2004 |
|          | 4        | Peugeot | 407    | 2005 |
|          | 5        | Renault | Cilo   | 2004 |
|          | 6        | Renault | Laguna | 2003 |

Fig 1.5 referential integrity constraints

While explaining such constraint among the tables Car as well as CarType, then it becomes feasible to remove rows from CarType table. It is seen that when you remove the Ford Focus row from CarType table, then you will find that the cars ABC-112, ABC-122, ABC-123 will also be removed from Car table.

**You can apply following rules to solve such problems:**

- Rule 1: You will not be able to remove any of rows located in CarType table as every car types are used in Car table.
- Rule 2: You will not be able to remove any model\_ids in the CarType table in view of the fact that every car types are in use in Car table.
- Rule 3: Whatever values you enter in model\_id field placed in the Car table should present in model\_id field in CarType table.
- Rule 4: The model\_id field in Car table encompasses null value where car types of such car are not identified.

### 1.4.2 Candidate key

Conveniently, there should be at slightest one smallest subset of attributes in the relation, which can be recognizing a tuple exclusively. Such type of negligible subset of attributes is termed as a key for particular relation. When there is more than one smallest subsets, then such type of key is known as candidate keys.

#### Example:

The accounts department of a company has segregated the Employee details in one table and Salary in another table. Further in the Employee details table, there are two columns, the employee ID and employee name. Further in the salary table, they have two columns, the employee ID and salary for given ID.

| Employee details |               | Salary      |                     |
|------------------|---------------|-------------|---------------------|
| Employee ID      | Employee name | Employee ID | Salary for given ID |
| AS 101           | Rohit Mathur  | AS 101      | 25000               |
| AS 102           | Amit Charan   | AS 102      | 15000               |
| AS 103           | Anuj Charan   | AS 103      | 25000               |
| AS 104           | Sangeeta      | AS 104      | 15100               |
| AS 105           | Kashyap       | AS 105      | 12000               |

Fig 1.6 Employee table

If the company wishes to remove an employee as he no longer works with it, then you have to delete his entry from the Employee table, as his identity is also present in Salary table. To remove an employee from the Salary table manually is very troublesome. If that employee presence is there in Companies other table, then he would have to be deleted from those tables also.

We can use referential integrity to solve such problem, so that in future we will not delete any name manually from any table. We will:

- First define the employee ID column in Employee table as primary key.
- Secondly, define the employee ID column in Salary table as foreign key which points to primary key located in employee ID column in Employee table.
- Third, add a constraint to the Salary table, which will add in selected cascading delete, where an employee can be deleted from Employee table any time.

From the above we see that if any entries that employee carries in Salary table will automatically get removed from Salary table.

### Check your progress 3

1. A candidate key is also known as:
  - a. Secondary key
  - b. Alternate key
  - c. Composite key
  - d. Concatenate key
2. Which is not the rule of referential integrity constraint?
  - a. You cannot remove a record from primary table if matching records are there in associated table.
  - b. You can transform a primary key value in primary table if that record has associated records.
  - c. You cannot enter a value in foreign key field of associated table that doesn't survive in primary key of primary table.
  - d. You can put a Null value in foreign key, by indicating that records are not linked.

---

## 1.5 Let Us Sum Up

---

In this unit we have learnt that referential integrity is related to explanation concerning with database where things are arranged in certain ways where column present in two or more tables in database will alter value in column. We have studied that Entity integrity ensures every row of a table to carryout non null primary key value. In this every row in table shows individual instance of an entity type model by the table.

The Entity completeness describes about every Entity which is transferred as Primary Key showing abnormal identifiers at every case of the entity. Referential integrity is another type of database model that makes it sure to facilitate the interaction flanked by tables which remains steady. When a single table contains a foreign key to other table, then the idea of referential integrity

explains with the aim that you might not put in a record inside the table which carries the foreign key except that there is an equivalent record in the linked table.

It is found that the number of DBMS describes the output format and/or input mask for the attribute. The constraints on domain contain a domain name, description, data type, size and acceptable values. Entity integrity ensures all rows of a table as well as non-null primary key value.

---

## 1.6 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-c)

**Check your progress 2**

**Answers:** (1-c)

**Check your progress 3**

**Answers:** (1-b), (2-b)

---

## 1.7 Glossary

---

1. **Key** - It is a single or combination of many fields which access or retrieve data rows from table.
2. **Primary Key** - It is an attribute or combination of attributes having unique identity as row or record in relation.
3. **Secondary key** - It is a field or combination of fields which retrieves.
4. **Candidate Key** - It is a relation containing many fields or combination of fields used as primary key.
5. **Composite key** - It is a primary key containing two or more attributes.
6. **Sort key** - It is a field or combination of fields used to sequence the stored data.
7. **Foreign Key** - It is an attribute in relation having value that is matched with primary key in another relation.

---

## 1.8 Assignment

---

What is property in a table?

---

## 1.9 Activities

---

Can a composite key be act as candidate key?

---

## 1.10 Case Study

---

Can you use the same foreign key constraint in two different tables?

---

## 1.11 Further Readings

---

1. Dependency Structures of Data Base Relationships by W. W. Armstrong.
2. Cardinal numbers and Formalized Mathematics by Grzegorz Bancerek.
3. The fundamental properties of natural numbers by Grzegorz Bancerek.
4. Finite sequences and tuples of elements of a non-empty sets by Czesław Byliński.



---

## **UNIT 2: NORMALIZATION**

---

### **Unit Structure**

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 Basic Concept of designing**
- 2.3 Need of normalization**
- 2.4 Decomposition**
- 2.5 Functional Dependency**
- 2.6 Full Functional Dependency**
- 2.7 Armstrong Axioms of Functional Dependencies**
- 2.8 Normalization Rules**
- 2.9 Examples based on normalization**
- 2.10 Let Us Sum Up**
- 2.11 Answer for Check Your Progress**
- 2.12 Glossary**
- 2.13 Assignment**
- 2.14 Activities**
- 2.15 Case Study**
- 2.16 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About normalization
- Basic of Composite key
- Understanding of Normal Forms

---

## 2.1 Introduction

---

While designing a database, if its design is not just right, then it seems that it carries anomalies. Controlling a database with anomalies is highly impracticable. While for a designer, there exist three possibilities:

- **Updating anomalies** - If data items are spread as well as are not connected to each one properly, it could possibly lead to extraordinary situations. If we try to update one data piece having its copies spread over numerous places, the small number of instances gets modernized correctly at the same time few others are left by old values. Such examples will make the database in an inconsistent state.
- **Deleting anomalies** - When you make an effort to delete a record, however parts of record was remained undeleted as lack of knowledge, then the data at that time gets saved somewhere else.
- **Inserting anomalies** - When you try to place something in the data in record which is not available at all.

With this method, you can eliminate all anomalies as well as fetch the database to a steady state. By Normalization process, you can thoroughly inspect relations for anomalies furthermore, after detection; you can remove such anomalies by dividing relations into two new connected relations.

Normalization is a significant measurement of the database development process as; it frequently gives the appearance of real look of database and shows the working of how data are going to work together in database.

## 2.2 Basic Concept of Designing

The figure 2.1 shows the main phases of database design which is connected with application design.

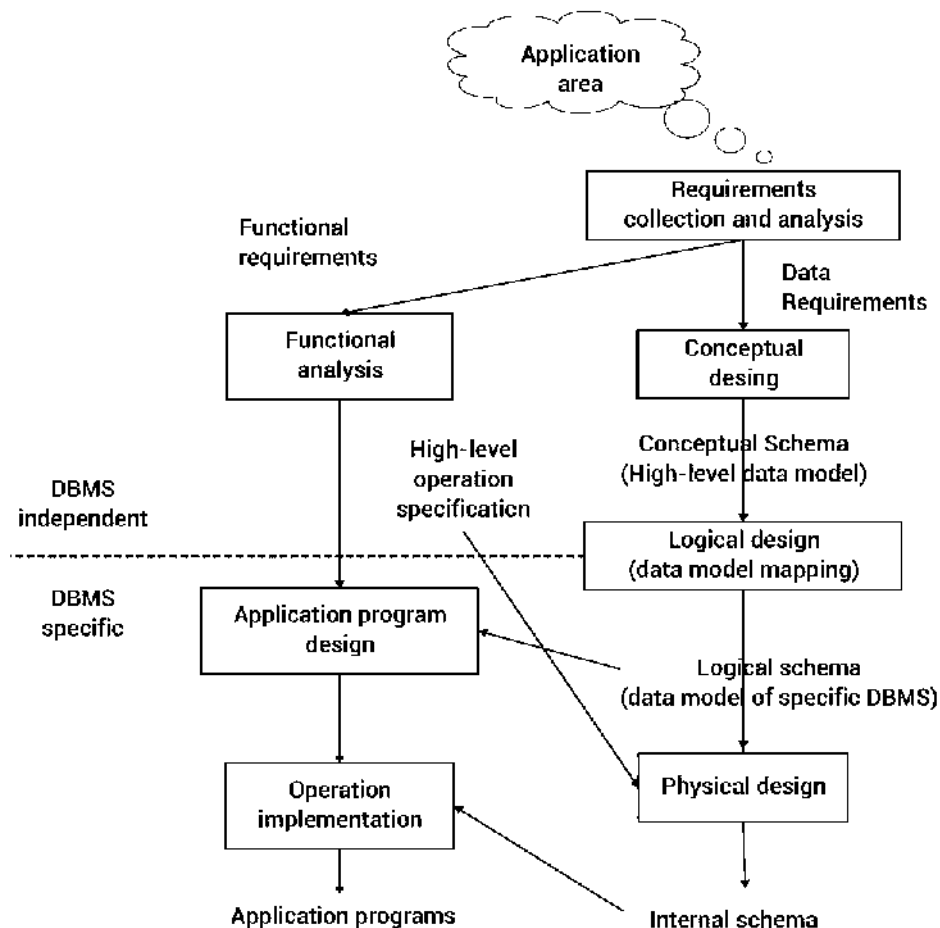


Fig 2.1 Database design

The needs as well as compilation analysis phase creates together the data needs as well as functional needs. The data needs works as a foundation of database design. The data need sought to be specified as detailed as well as total form as feasible.

In identifying data needs, it is helpful to identify the famous functional needs of the application which comprises of user defined process with the purpose to apply database. The practical needs are employed as a foundation of application software design. We see that a number of phases are database management system free as well as needy. The main view is to design initially the database without assess mention relation to definite database system which simply focus on the data.

After gathering all relevant information, the next action is to produce a conceptual plan for the database, by means of high level theoretical data model which relates to conceptual design.

This phase results in Entity-Relationship (ER) diagram or else UML class diagram. These are high-level data model which show how dissimilar entities are connected to each one. In these, the attributes carries each entity which carries all the explanation about concepts of application area.

All through the conceptual plan, the essential data model procedure can be used to identify the high-level user operations acknowledged all through the functional investigation.

It is further studied that normalization is the final portion of logical design. The objective of normalization is to get rid of redundancy as well as possibly update anomalies. Redundancy explains that similar data is kept more than once in a database. Bringing an anomaly up to date is an effect of laying-off. If a part of the data is put aside in additional one place, then the similar data is required to be updated in more than one place. By the method of normalization, you can alter the relation schema in order to decrease the laying-off. Every normalization period adds additional relations inside the database.

### Check your progress 1

1. The Entity-Relationship diagram is also called as:

- |                      |                      |
|----------------------|----------------------|
| a. UML class diagram | c. MLU class diagram |
| b. ULM class diagram | d. MUL class diagram |

---

## 2.3 Need of normalization

---

Normalization is a process where the table gets divided into various pieces also is applied to lower the redundancy in data that results in data inconsistency. If the tables are split into pieces, then to do manipulations on these, you will need joins as well as nested queries. Basically, database normalization is used to lower the redundancy as well as dependency. It significantly helps in dividing huge table into smaller ones meant for less redundancy as well as better relationship among data. Because of isolation of data, you can do additions, deletions as well as modifications of field in a single table plus spread throughout the rest of the

database by the use of defined relationships. So normalization eliminates redundancy as well as potentially keeps informed anomalies. While normalising a database you will accomplish four goals:

1. Arranging data in logical groups with each group explains about small portion of the whole.
2. Minimizing the quantity of reproduction data stored in a database.
3. Construction of database where you use as well as control the data rapidly as well as professionally with no compromise on integrity of data storage.
4. Arranging the data in such a way that on modification, you can do changes in simply one position.

### Check your progress 2

1. Normalization is used to:

- |                            |                          |
|----------------------------|--------------------------|
| a. reduce the table size   | c. expand the table size |
| b. increase the table size | d. all of these          |

---

## 2.4 Decomposition

---

Decomposition is basically changing a relation by means of collecting smaller relations. In database, the idea of decomposition results in breaking of tables into various tables and reaching to higher normal form. There are two characteristics of good decompositions:

- Lossless
- Preserve dependencies

### Lossless Decomposition

This shows the functioning without result in a loss. It retains everything. A good database should always carry such characteristics. We will explain this as:

- Assuming  $R$  to be a relation schema.
- Assuming  $F$  to be the set of functional dependencies on  $R$ .

Now we will form a decomposition of R. We see that the decomposition is lossless join decomposition of R, if one of following functional dependencies are in  $F^+$

- 1)  $R1 \cap R2 \text{ ----- } R1$
- 2)  $R1 \cap R2 \text{ ----- } R2$
- 3)  $R1 \cap R2 \text{ ----- } R1$
- 4)  $R1 \cap R2 \text{ ----- } R2$

We see that if R gets split into R1 as well as R2, then for lossless decomposition, at least one of two should hold true. Projecting on R1 and R2, and joining back, results in the relation you started with.

Now make sure that attributes is used in natural join ( $R1 \cap R2$ ) which is a candidate key for one of the two relations. It confirms that we will never have situation where false tuples originates, as for any value on join attributes, there will be a unique tuple in any relations. Further we see that decomposition is lossless if we can recover:

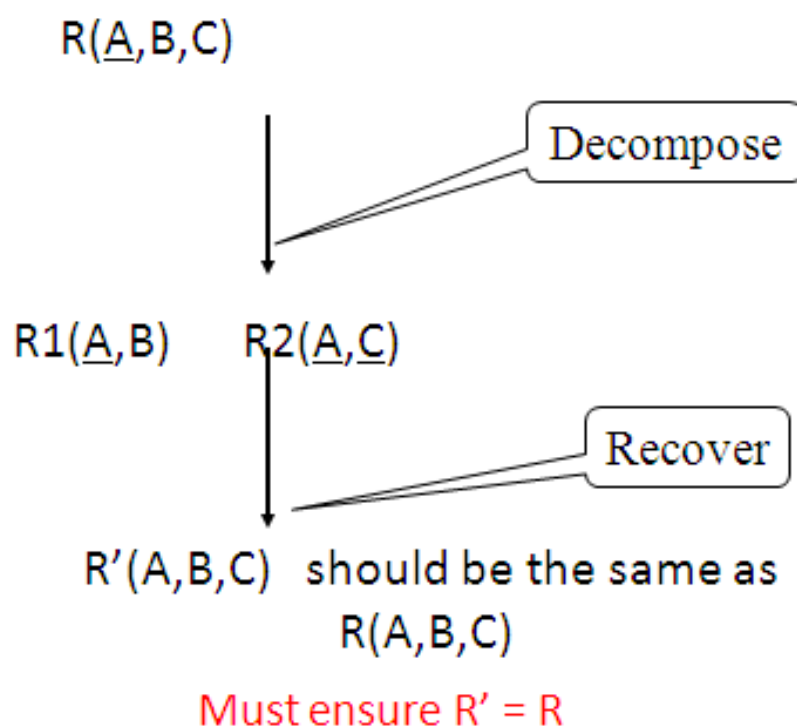
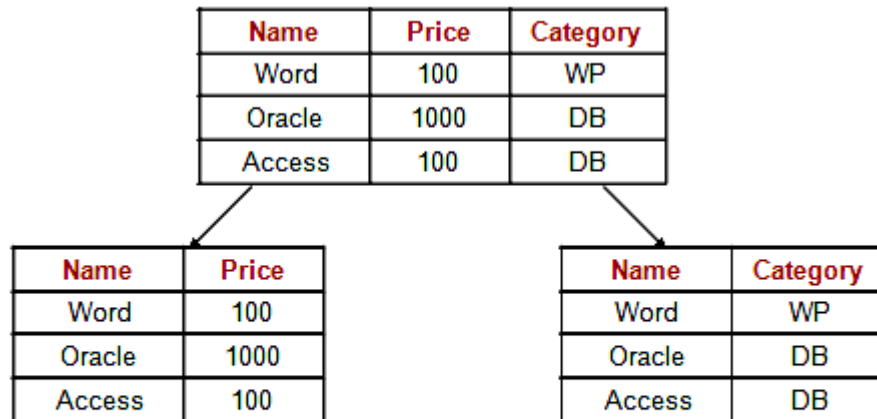


Fig 2.2 Lossless decomposition

Occasionally, the same set of data is replicated as:

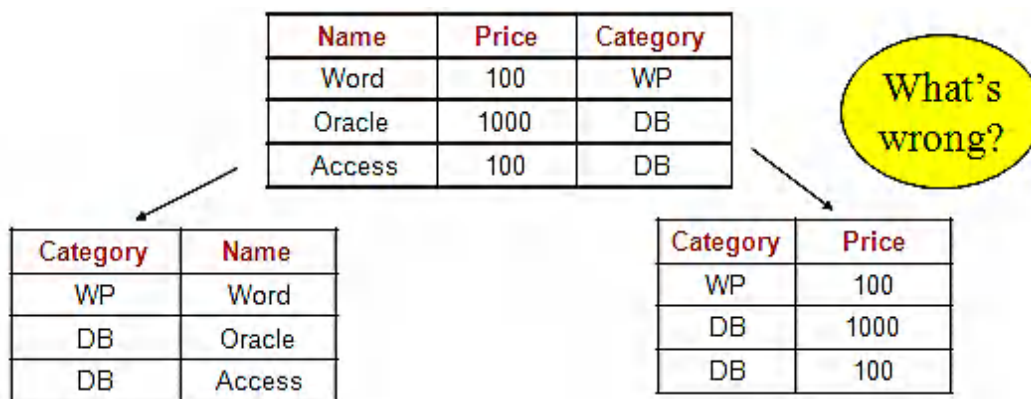
Normalization



$(\text{Word}, 100) + (\text{Word}, \text{WP}) \rightarrow (\text{Word}, 100, \text{WP})$   
 $(\text{Oracle}, 1000) + (\text{Oracle}, \text{DB}) \rightarrow (\text{Oracle}, 1000, \text{DB})$   
 $(\text{Access}, 100) + (\text{Access}, \text{DB}) \rightarrow (\text{Access}, 100, \text{DB})$

Fig 2.3 data replication set

Many times it will not work as shown:



$(\text{Word}, \text{WP}) + (100, \text{WP}) = (\text{Word}, 100, \text{WP})$   
 $(\text{Oracle}, \text{DB}) + (1000, \text{DB}) = (\text{Oracle}, 1000, \text{DB})$   
 $(\text{Oracle}, \text{DB}) + (100, \text{DB}) = (\text{Oracle}, \text{100}, \text{DB})$   
 $(\text{Access}, \text{DB}) + (1000, \text{DB}) = (\text{Access}, \text{1000}, \text{DB})$   
 $(\text{Access}, \text{DB}) + (100, \text{DB}) = (\text{Access}, 100, \text{DB})$

Fig 2.4 will not work data replication set

Make sure that lossless decomposition can only be when:

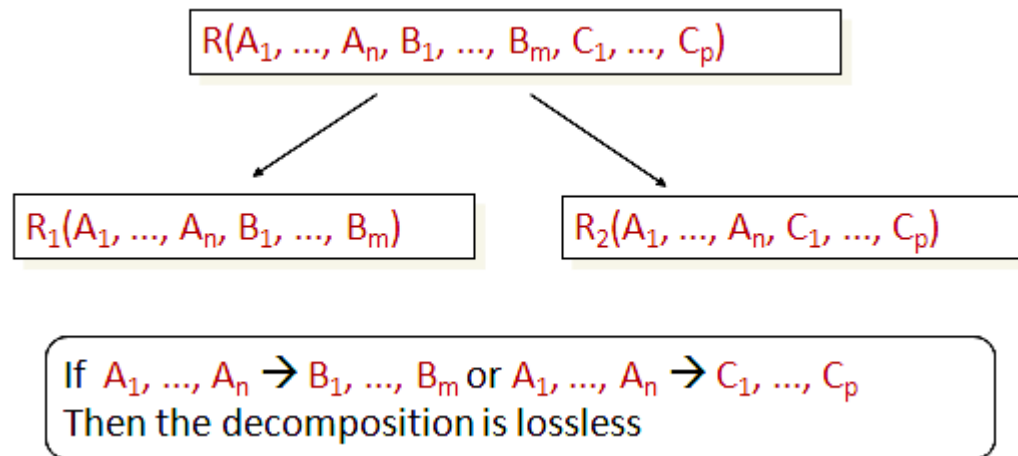


Fig 2.5 Lossless decomposition

### Identifying Lossless decomposition:

Initially prepare a table for sub schemas of R

Now complete the table by filling eminent variables (equivalent to sub schemas)

- If initial row is full with famous variables, then it is lossless
- If no single row gets full, then you have to add famous variables

You can add famous variables by following:

- 2 or more rows with famous variables on LHS
- 1 or more rows with famous variables on RHS
- 1 or more rows with non-famous variables on RHS

### Lossy Decomposition

In this type of decomposition, we lose the information not the record. Consider a relation R, which is decomposed into two relations R1 and R2.

We see that when relation  $R = (R1 \text{ JOIN } R2)$  is expressed, then we find this as lossless join decomposition which according to rule is accepted.

Also when relation R  $(R1 \text{ JOIN } R2)$  which acts as group then at that time, it will act as lossy join decomposition.



From the above we contemplate that, the lossy join decomposition, after combining R1 and R2 will generate extra records when it was compared with relation R. For this, we have consider a relation as shown R = (SN, Name, Address)

R1

| SN   | Name   | Address |
|------|--------|---------|
| 1111 | Rohit  | 1 Pine  |
| 2222 | Nishit | 2 Oak   |
| 3333 | Nishit | 2 Oak   |

| R1   |        | R2     |         |
|------|--------|--------|---------|
| SN   | Name   | Name   | Address |
| 1111 | Rohit  | Rohit  | 1 Pine  |
| 2222 | Nishit | Nishit | 2 Oak   |
| 3333 | Nishit | Nishit | 3Pine   |

If relations R1 and R2 are joined, then the result obtained is shown in the table:

R1 join R2

| SN   | Name   | Address |
|------|--------|---------|
| 1111 | Rohit  | 1 Pine  |
| 2222 | Nishit | 2 Oak   |
| 2222 | Nishit | 3 Pine  |
| 3333 | Nishit | 2 Oak   |
| 3333 | Nishit | 3 Pine  |

We see that the information lost here is the address for person 2222 and 3333. If you see the original relation R, then person 2222 lives at 2 Oak. When we join relations R1 and R2, then person 2222 shows that either he lives at 2 Oak or 3 Pine. With this, it seems that the extra information will result in lossy decomposition. Here, the records were not lost, however what is lost is the information regarding which the records were in original relation.

### Check your progress 3

1. Which is the popular property of decomposition?
  - a. Partition constraint
  - b. Dependency preservation
  - c. Redundancy
  - d. Security

---

## 2.5 Functional Dependency

---

Functional dependency is a connection which originates after one attribute exceptionally finds out one more attribute. It is an association that exists among two attributes having similar relational database table. In this, first attributes is called as determinant while second attribute is called as determined. It is visualised that for value of determinant there appears one value of determined.

If relation R with attributes X and Y, then functional dependency occurs among attributes is shown as  $X \rightarrow Y$ . It means that attribute Y is functionally dependent on attribute X. Now the attribute X is a determinant set furthermore Y is a dependent attribute. So every value of attribute X is connected accurately with single attribute Y value. So it is clear that the functional dependency in database provides a restriction among two sets of attributes.

It is seen that functional dependency explains about Boyce-Codd normal form as well as third normal form. This will conserve dependency among the attributes, which results in removing the replication of information. Functional dependency is connected to candidate key, which exceptionally recognize a tuple furthermore determines the value of the entire attributes in the relation. In several cases, functionally dependent sets will not get reduced if:

- Righth and set of functional dependency keeps only single attribute.
- Left and set of functional dependency not reduced as it alters the full content of set.
- Lowering any of accessible functional dependency that changes content of a set.

It is seen that there exists one of the famous property of functional dependency which is called as Armstrong's axiom that is utilised for database normalization. If R is the relation with three attributes X, Y, Z then Armstrong's axiom will result strong, when following situations are contented:

- Axiom of Transitivity: If  $X \rightarrow Y$  and  $Y \rightarrow Z$ , then  $X \rightarrow Z$
- Axiom of Reflexivity (Subset Property): If  $Y$  is a subset of  $X$ , then  $X \rightarrow Y$
- Axiom of Augmentation: If  $X \rightarrow Y$ , then  $XZ \rightarrow YZ$

### Example

From the supplier database shown below

| The Supplier Table |        |        |          |  |
|--------------------|--------|--------|----------|--|
| Sno.               | Sname  | Status | City     |  |
| S1                 | Suneet | 20     | Oadlan   |  |
| S2                 | Ankit  | 10     | Amritsar |  |
| S3                 | Amit   | 10     | Amritsar |  |

| The Part Table |       |       |        |          |
|----------------|-------|-------|--------|----------|
| Pno.           | Pname | Color | Weight | City     |
| P1             | Nut   | Red   | 12     | Oadlan   |
| P2             | Bolt  | Green | 17     | Amritsar |
| P3             | Screw | Blue  | 17     | Amritsar |
| P4             | Screw | Red   | 14     | Oadlan   |

| The Shipment Table |      |     |
|--------------------|------|-----|
| Sno.               | Pno. | Qty |
| S1                 | P1   | 270 |
| S1                 | P2   | 300 |
| S1                 | P3   | 700 |
| S2                 | P1   | 270 |
| S2                 | P2   | 700 |
| S3                 | P2   | 300 |

Fig 2.6 database table

Consider a Supplier table, where:

Sno: Serial number of Supplier

Sname: Supplier name

City: Place of supplier

Status: Shows the city grades as A grade cities or B grade cities.

We explained that Sname is FD on Sno. As Sname has individual approximation for particular value of Sno (S1), then there exists individual Sname for supplier number S1.

Here FD is shown as Sno a Sname

In this, a means Sname is functionally depends on Sno. Further, city as well as status is in addition FD on Sno, as for every value of Sno there will be single city along with status.

FD signifies:

Sno - City

Sno - Status

S. Sno – S (Sname, city, status)

Consider a shipment table, where:

S no: Number of Supplier

Pno: Part number of part

Qty: Quantity supplied for particular Part no

We see that Qty is FD on mixture of Sno, Pno for the reason that every grouping of Sno as well as Pno results simply for single Quantity.

SP (Sno, Pno) --> SP.QTY

In a dependency diagram we have attribute names as well as functional dependencies as shown.

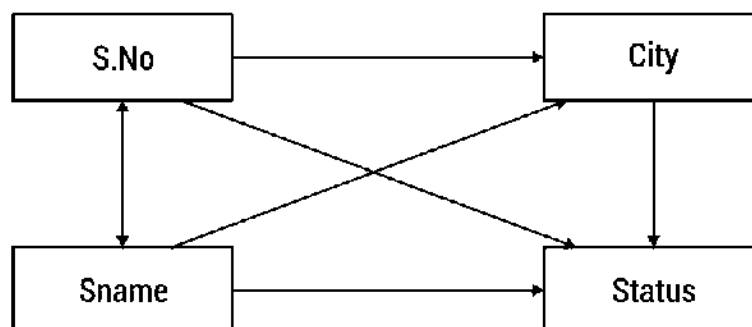


Fig 2.7 Dependency diagram

We see that from supplier table, we get following functional dependencies:

Sno: Sname

Sname: Sno

Sno: City

Sno: Status

Sname: City

Sname: Status

City: Status

Now the FD diagram of relation P can be shown as:

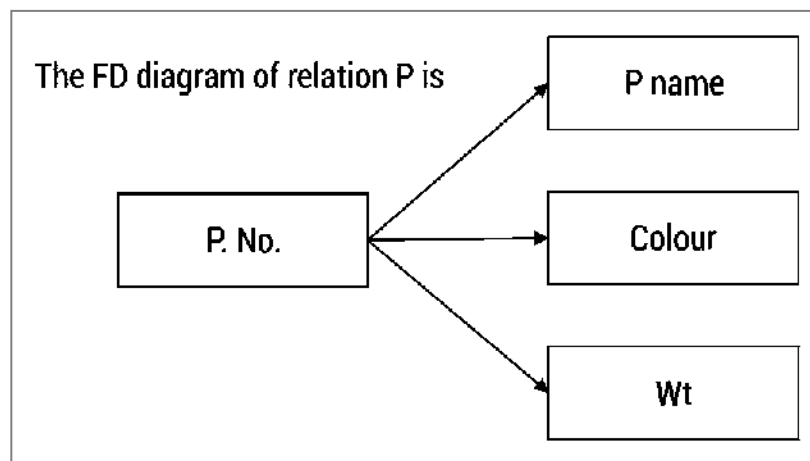


Fig 2.8 FD diagram

Now the functional dependencies for Part table:

Pno: Pname

Pno: Color

Pno: Wt

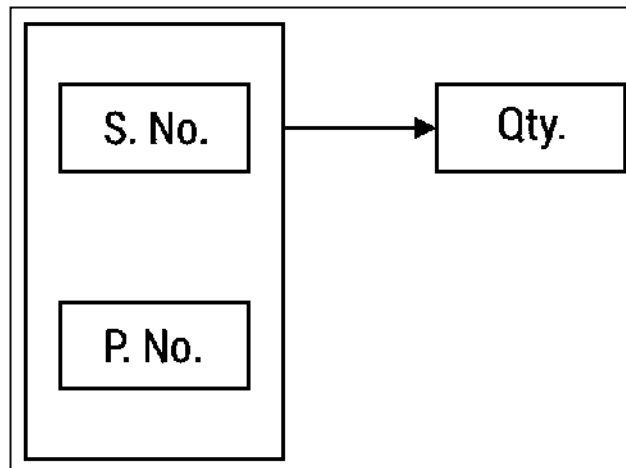


Fig 2.9 Functional dependencies

Finally we get functional dependencies the same as SP (Sno, Pno) - SP.QTY in Part table.

#### Check your progress 4

1. The functional dependency can be tested easily on the materialized view, using the constraints \_\_\_\_\_.
  - a. Primary key
  - b. Null
  - c. Unique
  - d. Both b and c
2. In a relational database table, the first attributes is:
  - a. determinant
  - b. determined
  - c. both a and b
  - d. neither a nor b

---

## 2.6 Full Functional Dependency

---

Full functional dependency take place when you previously meet the needs for a functional dependency along with the set of attributes lying on the left side of the functional dependency statement cannot be condensed beyond.

In a relation R in addition to Functional Dependency  $X \rightarrow Y$

We see that, Y is fully functionally dependent on X as well as there must not exist any  $Z \rightarrow Y$ , where Z is a appropriate subset of X.

Further, we see that if each and every non key attributes of the entity completely as well as functionally rely on key attribute of similar entity then such category of dependency is called as Full Functional Dependencies.

### Example

In a student table entity having 4 column attribute as:

- Sid
- Sname
- Add
- Course Name

The table Sname, Add and Course Name are non-key attribute that are fully relying on Sid(Key Attribute).

In this, to get any information about student, you have to use only Key Attribute i.e Sid. Apart from this, all other information can be obtained on the bases of key attribute. We see that all attribute (Sname,Add,CourseName) completely rely on key attribute. Hence because of this, it is called as Full Fuctional Dependencies.

### Check your progress 5

1. Consider a relation  $R(A,B,C,D,E)$  with functional dependencies as  $ABC \rightarrow DE$  and  $D \rightarrow AB$ . The number of super keys of R is:
 

|      |       |
|------|-------|
| a. 2 | c. 10 |
| b.7  | d. 12 |

---

## 2.7 Armstrong Axioms of Functional Dependencies

---

If F continues set relevantly functional dependencies that time the fastener of F, described as  $F^+$ , endures the set dependently all-inclusive functional dependencies feasibly intimated nearby F. Armstrong's codes are set of commands, when approached frequently brings about obstruction of functional dependencies.

Reflexive principle: If alpha endures a set of acknowledges as well as beta is\_subset\_of alpha, that time alpha accepts beta.

Augmentation principle: if  $a \rightarrow b$  assumes along with  $y$  endures acknowledge set, that time  $ay \rightarrow by$  by additionally assumes. It continues affixing which acknowledges in associations, operates and not alteration in elementary dependencies.

Transitivity principle: It is equivalent as transitive principle in algebra, if  $a \rightarrow b$ , we assumes along with  $b \rightarrow c$  that will affirms if  $a \rightarrow c$  that additionally gathers  $a \rightarrow b$  which endures identified appropriately as shown by b.

### Check your progress 6

1.The logically implied functional dependencies can be find by certain rules, such collection of rules is known as:

- |                       |              |
|-----------------------|--------------|
| a. Axioms             | c. Armstrong |
| b. Armstrong's axioms | d. Closure   |

---

## 2.8 Normalization Rules

---

Database Normalisation is a method of put in order the data in database. Basically normalization is a technique which was introduced for decomposing tables to keep away from data loss or unwanted features such as:

- Insertion
- Removing and updated Anamolies

It is a multi-step process that position data into table formation by keeping out double data from relation tables. The process of normalization applies mostly to:

- Get rid of useless data.
- Make certain about data dependencies.

Lacking of normalization will led to difficulty in handling and updating database, with no data loss. Insertion, Updation as well as Deletion Anomalies are very common if database is not normalized.



## Normalization Rule

There are 4 different rules of normalization such as:

- First Normal Form
- Second Normal Form
- Third Normal Form
- BCNF

### First Normal Form (1NF)

As per First Normal Form, no two Rows of data be required to hold repeating group of information. It seems that in FNF, every set of column should possess a unique value like multiple columns which cannot be used to carry out in similar row. Every table should be arranged in rows and every row should contain primary key that distinguishes it as only one of its kind.

The Primary key is more often than not a single column, but from time to time more than one column can be mutual to produce a single primary key. For instance consider a table which is not in First normal form.

Consider a table showing details of each student:

| <b>Student</b> | <b>Age</b> | <b>Subject</b> |
|----------------|------------|----------------|
| Nishit         | 15         | Biology, Maths |
| Rohit          | 14         | Maths          |
| Amit           | 17         | Maths          |

It is seen that in First Normal Form, in the least, the row ought to have a column where more values can be kept, by separating with commas. Rather than that, we must separate such data into multiple rows. In the student table subsequent 1NF will be seen as:

| <b>Student</b> | <b>Age</b> | <b>Subject</b> |
|----------------|------------|----------------|
| Nishit         | 15         | Biology        |
| Nishit         | 15         | Maths          |
| Rohit          | 14         | Maths          |
| Amit           | 17         | Maths          |

Applying First Normal Form, we see that data redundancy rises, since there are several columns having similar data in many rows having a unique identity.

### **Second Normal Form (2NF)**

The quality of the second normal form is that there should not partly dependency of column on main key. It is found that a table containing end to end joining of primary key will not be part of primary key and be kept complete end to end joining of key in order to work further. It is found that if any column having single end to end key, then during that time the table will not go after with Second normal form.

In illustration of First Normal Form, it is seen that there are 2 rows which covers many subjects that are taken by the students. At the same time, this is searchable, additionally pursue First normal form, so it is inefficient usage of space. In the Table, the First Normal Form, the candidate key is {Student, Subject}, Age of Student simply rely on Student column, which is not correct as per Second Normal Form. To get second normal form, it is advisable to break subjects into independent table as well as match them by student names with the help of foreign keys. Now the new student table of 2NF will show:

| <b>Student</b> | <b>Age</b> |
|----------------|------------|
|----------------|------------|

|        |    |
|--------|----|
| Nishit | 15 |
| Rohit  | 14 |
| Amit   | 17 |

It is seen that the candidate key is available in Student column, where Age is based on it. So we find another Subject Table for 2NF as:

| <b>Student</b> | <b>Subject</b> |
|----------------|----------------|
|----------------|----------------|

|        |         |
|--------|---------|
| Nishit | Biology |
| Nishit | Maths   |
| Amit   | Maths   |
| Rohit  | Maths   |

In the candidate column with having {Student and Subject, bot will meet for Second Normal Form that will continuously updated Anomalies.

### Third Normal Form (3NF)

Such type of normal form is good for every non-prime table features which depends on primary key. . It is seen that such type of transitive practical dependency should be avoided in the table and should favour for Second Normal form. To understand this, consider the following fields.

Student\_Detail Table:

| Student_id | Student_name | DOB | Street | City | State | Pincode |
|------------|--------------|-----|--------|------|-------|---------|
|------------|--------------|-----|--------|------|-------|---------|

We see that Student\_id is a Primary key whereas street, city and state has to depend on Pin code. So the dependency that exists among pin code with other fields are transitive. To have 3NF, we will re-shuffle the street, city and state with another table as shown:

New student\_Detail Table:

| Student_id | Student_name | DOB | Pincode |
|------------|--------------|-----|---------|
|------------|--------------|-----|---------|

Address Table

| Pincode | Street | City | State |
|---------|--------|------|-------|
|---------|--------|------|-------|

The benefit of moving out the transitive dependency is, Reducing of data duplication approximation.

Getting more data integrity

### Boyce and Codd Normal Form (BCNF)

Boyce with Codd Normal Form is assumed to be the best edition in terms of Third Normal form. In this, the appearance will result with anomaly that is not managed by 3NF. . It is found that a 3NF table that has no multiple overlapping candidate keys will remained in BCNF.

The following conditions should match for table to be in BCNF:

R to remain in 3rd Normal Form

For each functional dependency (  $X \rightarrow Y$  ), X to be super Key.

Consider the following relationship :  $R(A, B, C, D)$

and following dependencies :

$A \rightarrow BCD$

$BC \rightarrow AD$

$D \rightarrow B$

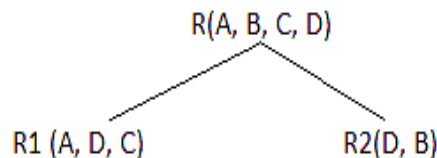
Above relationship is already in 3rd NF. Keys are  $A$  and  $BC$ .

Hence, in the functional dependency,  $A \rightarrow BCD$ ,  $A$  is the super key.

in second relation,  $BC \rightarrow AD$ ,  $BC$  is also a key.

but in,  $D \rightarrow B$ ,  $D$  is not a key.

Hence we can break our relationship  $R$  into two relationships  $R1$  and  $R2$ .



Breaking, table into two tables, one with  $A, D$  and  $C$  while the other with  $D$  and  $B$ .

Fig 2.10 Function dependency chart

### Check your progress 7

1.  $F1, F2, F3, F4$ , and  $F5$  are fields in table having functional dependencies as:

$F1 \rightarrow F3$

$F2 \rightarrow F4$

$(F1, F2) \rightarrow F5$

Then with respect to normalization, the table will be in:

a. 1NF

c. 3NF

b. 2NF

d. None of the mentioned

2. In a relational schema  $R(A, B, C, D, E, P, G)$ , following FDs are known:

$AB \rightarrow CD$

$DE \rightarrow P$

$C \rightarrow E$

$P \rightarrow C$

$B \rightarrow G$

Now the relation schema R is:

- a. in BCNF
  - b. in 3NF, but not in BCNF
  - c. in 2NF, but not in 3NF
  - d. not in 2NF
3. Which normal form is considered adequate for normal relational database design?
- a. 2NF
  - b. 5NF
  - c. 4NF
  - d. 3NF
4. When the attribute of composite key depends on other composite, then a relation will be \_\_\_\_.
- a. 2NF
  - b. 3NF
  - c. BCNF
  - d. 1NF

## 2.9 Examples based on normalization

### 1st Normal Form Example

To bring about un-normalized table in first normal form, consider an example shown:

Table: 1st Normal Form

| TABLE_PRODUCT |              |       |
|---------------|--------------|-------|
| Product ID    | Color        | Price |
| 1             | red, green   | 15.99 |
| 2             | yellow       | 23.99 |
| 3             | green        | 17.50 |
| 4             | yellow, blue | 9.99  |
| 5             | red          | 29.99 |

This table is not in first normal form as the [Color] column carries various values. If you see the first row that has values "red" and "green." Now to bring this table to first normal form, you have to distribute the table in two tables as:

Table Distributed tables

| TABLE_PRODUCT_PRICE |       | TABLE_PRODUCT_COLOR |        |
|---------------------|-------|---------------------|--------|
| Product ID          | Price | Product ID          | Color  |
| 1                   | 15.99 | 1                   | red    |
| 2                   | 23.99 | 1                   | green  |
| 3                   | 17.50 | 2                   | yellow |
| 4                   | 9.99  | 3                   | green  |
| 5                   | 29.99 | 4                   | yellow |
|                     |       | 4                   | blue   |
|                     |       | 5                   | red    |

Now first normal form is satisfied, as the columns on each table all hold just one value.

## 2nd Normal Form Example

Consider the following table:

Table: Normal Form

| TABLE_PURCHASE_DETAIL |          |                   |
|-----------------------|----------|-------------------|
| Customer ID           | Store ID | Purchase Location |
| 1                     | 1        | Los Angeles       |
| 1                     | 3        | San Francisco     |
| 2                     | 1        | Los Angeles       |
| 3                     | 2        | New York          |
| 4                     | 3        | San Francisco     |

This table has a compound primary key [Customer ID, Store ID]. The non-key characteristic is [Purchase Location]. In this situation, [Purchase Location] merely depends on [Store ID], which is no more than single part of primary key.

As a result, this table does not please second normal form. To carry this table to second normal form, we rupture the table into two tables, further more as:

Table: Data

| TABLE_PRODUCT |          | TABLE_STORE |                   |
|---------------|----------|-------------|-------------------|
| Customer ID   | Store ID | Store ID    | Purchase Location |
| 1             | 1        | 1           | Los Angeles       |
| 1             | 3        | 2           | New York          |
| 2             | 1        | 3           | San Francisco     |
| 3             | 2        |             |                   |
| 4             | 3        |             |                   |

What we have done is to take away the partial functional reliance that we originally have. At this time, in the table [TABLE\_STORE], the column [Purchase Location] is completely needy on the primary key of that table, which is [Store ID].

### Check your progress 8

1. In \_\_\_\_\_ normal form, a composite attribute is converted to individual attributes.
  - a. First
  - b. Second
  - c. Third
  - d. Fourth

## 2.10 Let Us Sum Up

In this unit we have studied that normalization is an important measurement of a database development process which shows a real look of database and displays working of data which will helped for particular database.

With the method of normalization we can change relation schema in order to decrease the laying-off. It is seen that all normalization period will give extra relations available in database.

It is studied that database normalization is used to lower the redundancy as well as dependency. It will probably help in dividing huge table into smaller ones so to possess low redundancy and to have better relationship among data.

In this unit, we have seen that decomposition is used to change relation by means of collecting smaller relations. As seen, use of decomposition in database results in breaking of tables into various tables and reaching to higher normal form.

Full functional dependency take place when you previously meet the needs for a functional dependency along with the set of attributes lying on the left side of the functional dependency statement cannot be condensed beyond

---

## 2.11 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-a)

**Check your progress 2**

**Answers:** (1-a)

**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-c), (2-a)

**Check your progress 5**

**Answers:** (1-c)

**Check your progress 6**

**Answers:** (1-b)

**Check your progress 7**

**Answers:** (1-a), (2-d), (3-d), (4-b)



Answers: (1-a)

---

## 2.12 Glossary

---

1. **Normalization** - A process where data in relational construct is organized to lower redundancy and non-relational constructs.
2. **Physical model** - Data modelling level where you add database and database management system (DBMS) specific modelling information such as tables, columns, and data types.
3. **Primary key** - It is an attribute or attributes that uniquely identify an instance of an entity.
4. **Referential integrity** - It is an assertion where a foreign key values in an instance of child entity corresponds to values in parent entity.

---

## 2.13 Assignment

---

Collect the information about the value of normalizing a table, if normalization is done from second normal form to third normal form?

---

## 2.14 Activities

---

Why do I need to use a subquery for this SQL query?

---

## 2.15 Case Study

---

In data mining and statistical data analysis, when do you need to normalize data and why is it important to do so?

---

## 2.16 Further Readings

---

1. Dependency Structures of Data Base Relationships by W. W. Armstrong.
2. Cardinal numbers and Formalized Mathematics by Grzegorz Bancerek.
3. The fundamental properties of natural numbers by Grzegorz Bancerek.
4. Finite sequences and tuples of elements of a non-empty sets by Czesław Byliński

---

## **Block Summary**

---

While studying this block, the user will achieve knowledge and understanding about Database Integrity Concepts and the idea of Normalization. The block explains about Domain Integrity constraints, Referential Integrity constraints and Entity Integrity constraints with certain examples that will help the user to grab the concept.

The block explains about basic understanding of various integrity concepts and ideas of normalization that will make effective for making database. The features of decomposition in database are well explained through diagrams and illustration. The students or programmers will get benefit while reading this block as it gives shortcuts and related examples that will clear all doubts

---

## Block Assignment

---

### Short Answer Questions

1. What are Domain Integrity constraints?
2. What is Entity Integrity constraint?
3. What is Functional Dependency?
4. Explain Entity Integrity constraints?
5. Is database normalization important?

### Long Answer Questions

1. Explain the rules of normalization?
2. Explain Primary key and unique key?
3. How do you standardize a monthly distribution to account for the different length of each month?

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                              | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|----------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| <b>Presentation Quality</b>                        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Language and Style</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Illustration used<br/>(Diagram, tables etc)</b> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Conceptual Clarity</b>                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Check your progress<br/>Quest</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| <b>Feed back to CYP<br/>Question</b>               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# FUNDAMENTALS OF OPERATING SYSTEM

PGDCA 104

## BLOCK 2: MEMORY MANAGEMENT AND PROCESS SCHEDULING

Dr. Babasaheb Ambedkar Open University



# FUNDAMENTALS OF OPERATING SYSTEM

---



Knowledge Management and  
Research Organization  
Pune





## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Prof. Jaipal Gaikwad

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)



## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



## **FUNDAMENTALS OF OPERATING SYSTEM**

### **Contents**

#### **BLOCK 1: INTRODUCTION TO OPERATING SYSTEMS**

##### **UNIT 1 BASICS OF OS**

Definition and Function of operating systems, Evolution of operating system, Operating system structure-monolithic layered, virtual machine and Client server

##### **UNIT 2 TYPES OF OPERATING SYSTEM**

Different types of operating system-real time systems, multi-user System, distributed system

##### **UNIT 3 BATCH OPERATING SYSTEM**

Introduction to basic terms and batch processing system: Jobs, Processes files, command interpreter

---

#### **BLOCK 2: MEMORY MANAGEMENT AND PROCESS SCHEDULING**

##### **UNIT 1 MEMORY MANAGEMENT**

Logical and Physical address protection, paging, and segmentation, Virtual memory, Page replacement algorithms, Cache memory, hierarchy of memory types, Associative memory

##### **UNIT 2 PROCESS SCHEDULING**

Process states, virtual processor, Interrupt mechanism, Scheduling algorithms Performance evaluation of scheduling algorithm, Threads

---



**BLOCK 3: FILE AND I/O MANAGEMENT**

**UNIT 1 FILE SYSTEM**

File systems-Partitions and Directory structure, Disk space allocation, Disk scheduling

**UNIT 2 I/O MANAGEMENT**

I/O Hardware, I/O Drivers, DMA controlled I/O and programmed I/O, I/O Supervisors

**BLOCK 4: BASICS OF DISTRIBUTED OPERATING SYSTEM**

**UNIT 1 DISTRIBUTED OPERATING SYSTEM**

Introduction and need for distributed OS, Architecture of Distributed OS, Models of distributed system

**UNIT 2 MORE ON OPERATING SYSTEM**

Remote procedure Calls, Distributed shared memory, Unix Operating System: Case Studies



Dr. Babasaheb  
Ambedkar  
Open University

**PGDCA 104**

# **FUNDAMENTALS OF OPERATING SYSTEM**

---

## **BLOCK 2: MEMORY MANAGEMENT AND PROCESS SCHEDULING**

---

### **UNIT 1**

MEMORY MANAGEMENT

### **UNIT 2**

PROCESS SCHEDULING

---

# **BLOCK 2: MEMORY MANAGEMENT AND PROCESS SCHEDULING**

---

## **Block Introduction**

An operating system is important software which makes the computer to run. It handles all the computer processes and runs the hardware. It makes you to communicate with computer without having command on its language. It is seen that your computer operating system manages all software and hardware functions. The main idea of operating system is to coordinate will all processes and links these processes with central processing unit (CPU), memory and storage.

In this block, we will discuss detail about the basic of memory management and process scheduling of Operating System. The block will focus on the study and concept of virtual memory, paging and segmentation. The students will give with the idea about Catch memory and virtual processor.

In this block, the student will make to learn and understand about the basic of memory management techniques and its techniques. The concept related to memory hierarchy, process state and interrupt mechanism will also be explained to the students. The student will be demonstrated practically about the working of page replace algorithm and its technique.

## **Block Objective**

**After learning this block, you will be able to understand:**

- About Logical and Physical address protection of memory.
- Study about paging and segmentation.
- Idea about virtual memory.
- Detailed on page replace algorithms.
- Knowledge about different type of memory hierarchy.
- Concept of process states.
- Generalization of virtual processor.



Memory  
Management and  
Process  
Scheduling

- Basic of interrupt mechanism.

## **Block Structure**

**Unit 1:     Memory Management**

**Unit 2:     Process Scheduling**

---

# **UNIT 1: MEMORY MANAGEMENT**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Logical and Physical Address Protection**
- 1.3 Paging and Segmentation**
- 1.4 Virtual Memory**
- 1.5 Page Replacement Algorithms**
- 1.6 Cache Memory**
- 1.7 Hierarchy of Memory Types**
- 1.8 Associative Memory**
- 1.9 Let Us Sum Up**
- 1.10 Answers for Check Your Progress**
- 1.11 Glossary**
- 1.12 Assignment**
- 1.13 Activities**
- 1.14 Case Study**
- 1.15 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Memory Management Unit
- External and internal fragmentation
- Virtual page number
- Paging address Translation
- Virtual and physical memory
- Importance of Cache memory
- Associative memory

---

## 1.1 Introduction

---

Memory management is a type of subsystem which is an important part of an operating system. During the computing period there was continuously need of more memory in computer systems. Strategies have been developed to overcome this limitation and the most successful of these is virtual memory. Virtual memory makes the system appear to have more memory than it actually has by sharing it between competing processes as they need it. With earlier computing it is found that:

- Program should be carryout into the memory and is kept inside a process for further working.
- Input queue which is collecting of processes information on the disk are brought into the memory for implementation.
- The single process implementation will do from input queue which is loaded inside the memory for implementation.
- Once the implementation is done, then the implementation memory space will become free.
- In computers, the address space starts with 00000, which was first address of the user process that cannot be all 0.

---

## 1.2 Logical and Physical Address Protection

---

In the meantime, the computer acquires inter communicated through logical as well as physical addressing in order to map its memory. The logical approach is developed by the processor which is additionally designated as virtual address. The program observes this address space. Whereas the physical address exists the real address which continues assumed by the computer hardware like as memory unit. It is determined that the logical to physical address analysis occurrence acted or conveyed off is experienced by the Operating System. Here, virtual memory designates to the absence of estrangement of logical memory which is glanced nearby the process from physical memory which is inspected nearby the processor. On approximate of this separation, the computer programmer desires to be careful of about logical memory space while the operating system affirms two or more levels of physical memory space.

It is found that during the compile time and load time, the address binding schemes makes these two tend similar but they differ in execution time address

binding scheme and MMU (Memory Management Unit) that caters the translation of such addresses.

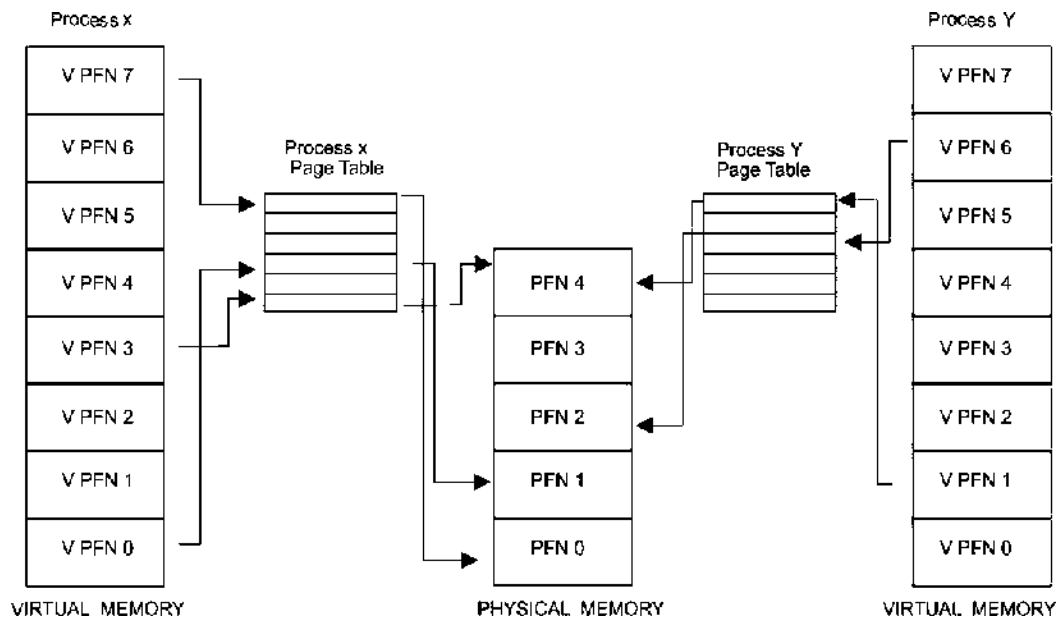


**Fig 1.1 Virtual to Physical address mapping**

In fig 1.1, it is looked that every procedure in the system comprises its own virtual address space. Similarly virtual address spaces are entirely alienated from each other furthermore on account of a process bounding one application cannot authorize another. Additionally, the hardware virtual memory approaches assign regions of memory to be maintained across writing. This conserves code as well as data from being overwritten by miscreant applications.

In the virtual to physical address mapping demonstrated in fig 1.1, as the processor deliver out the program it understands an instruction from memory furthermore analyse it. While the processor interprets the instruction, it desires to fetch or store the contents of a position in the memory. Following that, the processor will deliver out the instruction furthermore actuate onto the following instruction in the program. With this concept, the processor is consecutively approaching memory to deliver out the instructions or to acquire as well as accumulate the data.

In rack relevantly virtual memory system, complete similar addresses are virtual addresses in addition are not physical addresses. The virtual addresses acquires altered into physical addresses with the support of a processor based on

information that had been conveyed in a set of tables to preserve safe the operating system.

To simplify this, virtual as well as physical memory are allotted as handy sized chunks recognized as pages. Similarly chunks are of identical size where it serves difficult for the system to monitor as Linux on Alpha AXP systems utilizes 8 KB pages as well as on Intel x86 systems it applies 4 KB pages. Every chunk of pages are allotted with a unique number as page frame number (PFN).

Under such model, a virtual address is made of two parts:

- Offset
- Virtual page frame number

If the page size is 4 KB, then the bit ratio will be 11:0 of virtual address which has offset and bits number 12 and over this are virtual page frame number. Every time the processor comes across a virtual address which takes the offset and the virtual page frame number. For this, the processor should translate virtual page frame number into physical frame number and contact the location for correct offset into physical page by using page tables.

It is found that the virtual memory will allow process to be of virtual address spaces, so that there are times when you need processes to share such memory. Now the processor uses virtual page frame number as an index into the processes page table to get back its page table entry. If the entry is valid, then the processor will carry out the physical page frame number from such entry. If the entry is not a valid entry, then the process will try to come out with a non-existing area of its virtual memory. Under such conditions, the processor cannot solve the address and will pass the control to the operating system which is a permanent process.

The concept of logical address space that is bound to a separate physical address space is central to proper memory management.

- Logical address – generated by the CPU; also referred to as virtual address
- Physical address – address seen by the memory unit

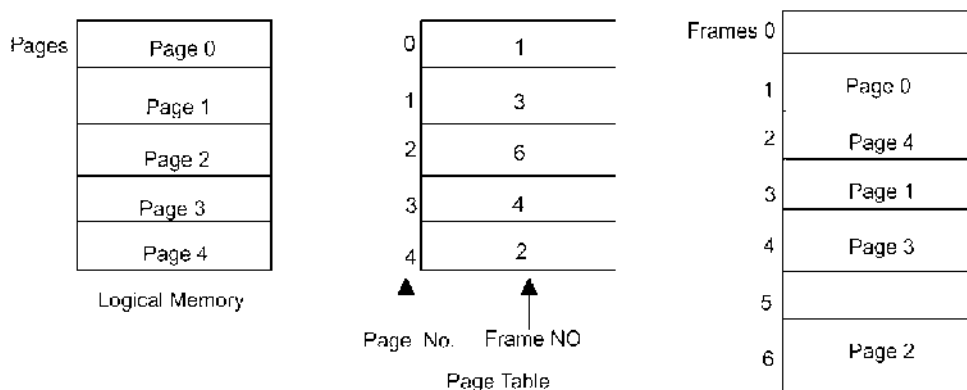
Logical and physical addresses are the same in compile-time and load-time address-binding schemes; logical (virtual) and physical addresses differ in execution-time address-binding scheme

**Check your progress 1**

- In logical address protection, the logical address is generated by the \_\_\_\_\_.  
 a. memory  
 b. processor  
 c. memory address  
 d. none of these
- Input queue is a collection of process \_\_\_\_\_ on the disk  
 a. data  
 b. information  
 c. both a and b  
 d. neither a nor b
- Linux on Alpha AXP systems utilizes \_\_\_\_KB pages  
 a. 8  
 b. 16  
 c. 32  
 d. 64

**1.3 Paging and Segmentation****Paging**

Paging is a process that will help in solving the problem that was seen in case of variable sized partitions such as external fragmentation. In paged system, the logical memory is sliced into number of constant sizes chunks called as pages. Further, the physical memory is pre-divided in certain constant sized blocks which are known as page frames. The page sizes or the frame sizes will be of power 2, and fluctuates between 512 bytes to 8192 bytes per page. They have certain bytes per page because of the implementation of paging mechanism with page number and page offset.

**Fig 1.2 Paging operation**

In fig 1.2, the process page gets loaded to particular memory frame. Such pages will further be loaded into neighboring frames or in non-neighboring memory as highlighted in the figure 1.2 It is seen that the outside fragmentation gets improved because the processes gets inside in a separate holes.

### Page Allocation

With variable sized partitioning of memory, it is seen that every time a process of n size is loaded which is the best location from the list of available/free holes. Such type of dynamic storage allocation is required as it increases the efficiency and throughput of system. This type of selection can be done by using:

- 1) Best-fit Policy: It allocates the hole where the process is tight as the difference between whole size and process size is lowest.
- 2) First-fit Policy: This will allocates the initial found hole that can be big enough to fit in the new process.
- 3) Worst-fit Policy: It allocates the maximum size hole which leaves the full amount of unused space.

From the above three listed strategies it seems that the strategy best-fit and first-fit are better as compared to the worst-fit. Both best-fit and first-fit strategies are efficient in terms of time and storage capacity. In case of best-fit strategy, minimum leftover space is seen which will create the smallest hole which are not used frequently. In case of first-fit strategy, it uses least overheads in order to work because it is the simplest strategy to work upon. Possibly worst-fit also sometimes leaves large holes that could further be used to accommodate other processes. Thus all these policies have their own merits and demerits.

### Hardware Support for Paging

It is seen that all logical page in paging scheme is further divided as:

- Page number (p) in logical address space
- Displacement in page pat which item resides

Such arrangement is known as Address Translation scheme as it shows that in case of a 16-bit address, we can divide the address as:

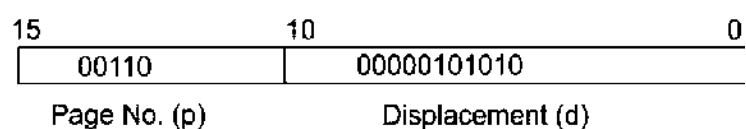


Fig 1.3 dividing the address

From the figure 1.3, it is seen that a page number will take 5 bits with its range starts from value 0 to 31 that can be  $2^5-1$ . Likewise, if we consider an offset value of having 11-bits, then the range will be from 0 to 2023 which is  $2^{11}-1$ . Totally, we see that the paging scheme uses 32 pages, each having 2024 locations. Also, the table that keeps virtual address to physical address translations is further classified as page table. It is found that as the displacement is fixed, the translation of virtual page number to physical page exists which can be seen in the figure 1.4.

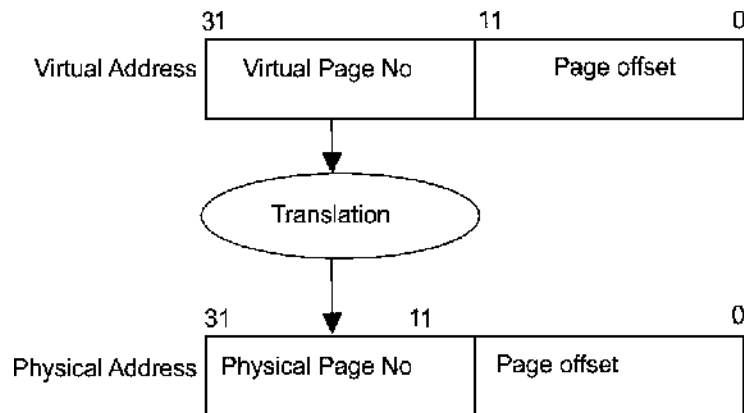


**Fig 1.4 Address Translation scheme**

It is seen that the page number is required in shape of an index which is into the page table containing base address for every corresponding physical memory page number. This arrangement will lowers the dynamic relocation efforts which are shown by the paging hardware support as in figure 1.5.

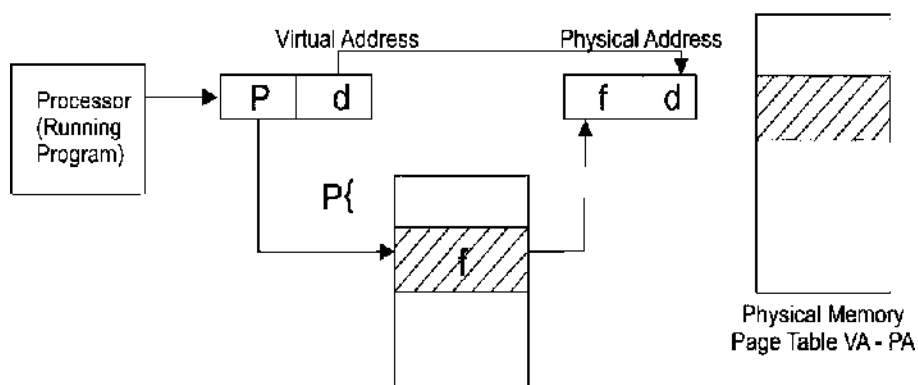


**Fig 1.5 Direct Mapping**



## Paging address Translation by direct mapping

Consider a case of direct mapping as shown in fig 1.5 where a page table sends directly to physical memory page. In this, the drawback is that the speed of translation decreases because the page table is put in primary storage place having a considerably larger in size that increases the instruction execution time and led to lowering of system speed. In order to conquer such situation, the use of extra hardware such as registers and buffers are used.

## Paging Address Translation with Associative Mapping

It is based on the utilization of fixed registers that has high speed and efficiency. Such small, fast-lookup Catch will help to put the whole page table in content-addresses associative storage place thereby making the speed to improve and further to care for the lookup problem of the Catch. These are known as associative registers or Translation Look-aside Buffers (TLB's). It is found that each register consists of two entries:

- 1) Key, which is matched with logical page.
- 2) Value which returns page frame number corresponding top.

Such arrangement is same as direct mapping scheme but only difference is that we have associative registers having few page table entries that made the search fast. It is quite expensive due to the presence of register support. Hence it is found that both direct and associative mapping schemes will combine to result in more benefits. In this, that page number is coordinated with associative registers at the same time. Also the percentage of number of times the page is found in TLB's is further termed as hit ratio. If it is not found, it is seek out in page table and added into TLB. In case if the TLB is full, then the page replacement policies will come into effect. It is found that the entry in TLB is limited only. Such type of combined scheme is shown in Figure 1.6.

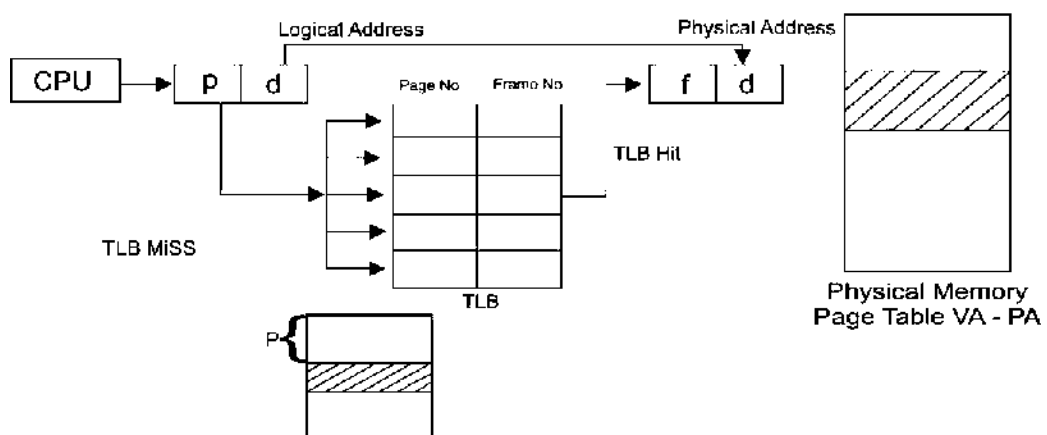


**Fig 1.6 Type of combined scheme**

It is seen that in paging hardware, there is a presence of some protection mechanism. Inside the page table there exists corresponding frame where a protection bit is linked. Such type of bit will show whether the page is read-only or read-write. In this, sharing code and data will take place only when two pages table entries in different process shows the similar physical page where every process shares the memory. It is seen that if one process writes the data, then the second process will locate for the changes. Such type of an arrangement is quite efficient while in communicating. Sharing is required to control in order to protect modification and admission of data in a single process with the help of second process. Such type of programs is kept independent as procedures and data where procedures and data that are pure/reentrant code get shared. Re-entrant code will not be able to change itself and should make sure that it contains a separate copy of per-process global variables. It is predicted that modifiable data and procedures will not share without the intervention of concurrency controls. Such type of non-modifiable procedures sometimes are called as pure procedures or reentrant codes. In case of an example, it is illustrated that in such system only single copy of editor or compiler code be kept in the memory, and all editor or compiler processes and executes sit with the help of single copy of code which will help in memory utilization.

### **Advantages**

There are certain advantages of paging scheme such as:

1. Virtual address space must be greater than main memory size. i.e., can execute program with large logical address space as compared with physical address space.
2. Avoid external fragmentation and hence storage compaction.
3. Full utilization of available main storage.

### **Disadvantages**

The disadvantages of paging scheme include:

1. Internal fragmentation problem led to wastage inside the allocated page
2. Extra resource consumption
3. Overheads for paging hardware
4. Virtual address to physical address translation takes place

## Segmentation

In generic, a consumer or a programmer likes to observe system memory as an assembly of variable-sized allocations rather than as a linear arrangement of words. Segmentation occurs as a memory management arrangement that accepts this glance of memory.

### Principles of Operation

Segmentation demonstrates an exchange arrangement for memory management. This arrangement bisects the logical address space into variable length allocations, named segments, with no appropriate sequencing among them. Each allotment has a name and a length. For clarity, segments are acknowledged by a segment number, rather than by a name. Hence, the logical addresses are acknowledged as a pair of segment number as well as offset within segment. It empowers a program to be broken down into feasible parts according to the user opinion of the memory, which is that time mapped into physical memory. Furthermore logical addresses are two-dimensional although actual physical addresses are still one-dimensional arrangement of bytes only.

### Address Translation

This mapping between two is done by segment table, which contains segment base and its limit. The segment base has starting physical address of segment, and segment limit provides the length of segment. This scheme is depicted in Figure 1.7.

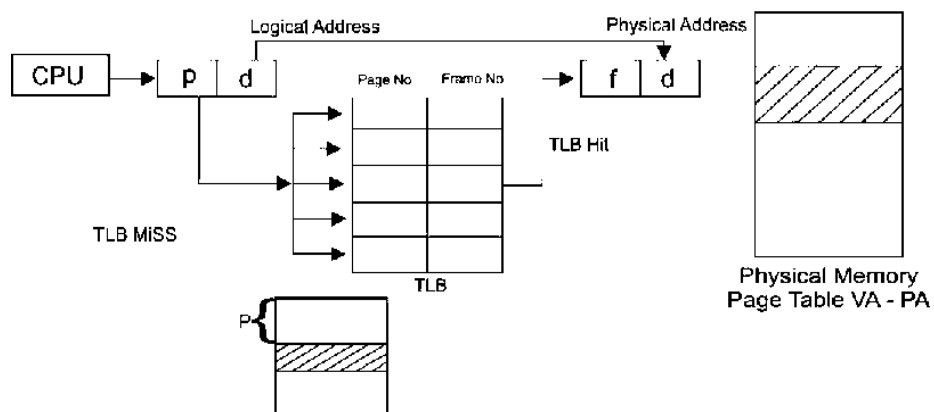


Fig 1.7 Address translation

The offset  $d$  must range between 0 and segment limit/length, otherwise it will generate address error. For example, consider situation shown in Figure 1.8.

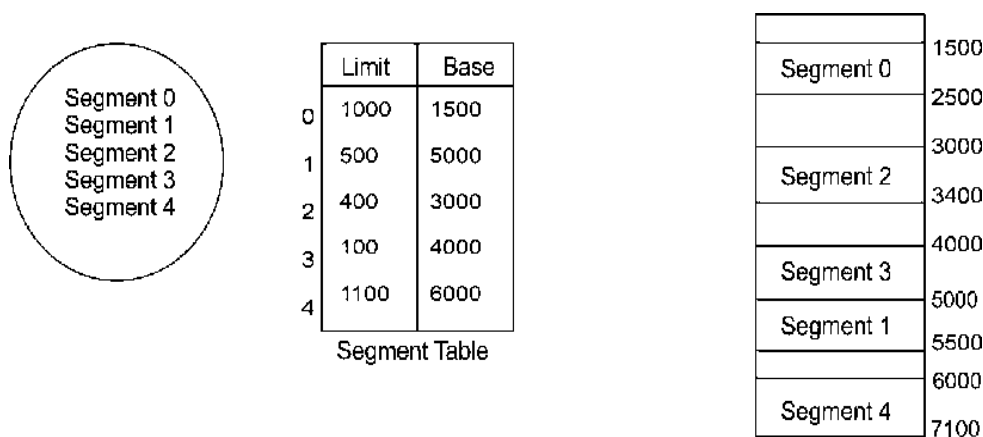


**Fig 1.8 Principle of operation of representation**

This approximation is comparable to adaptable partition allocation method with advancement that the process is bifurcated into parts. For quick retrieval we can utilize registers as in paged approach. This is comprehended as a segment-table length register (STLR). The segments in a segmentation mechanism dispatch to logical divisions of the process additionally are described by program names. Extract the segment number along with offset from logical address originally so that time the use of segment number as index into segment table gets capture segment base address along with its limit /length. Additionally, contemplate that the offset is not greater than allocated limit in segment table. Today, normally physical address is acquired by adding the offset to the base address.

### Protection and Sharing

This approach in addition enables segments that are read-only to be allotted, so that two approaches can utilize shared code for advance memory efficiency. The intervention is comparable that no program can read from or write to chunks belonging to another program, except the allocations that have been set up to be apportioned. With each segment-table entry safety bit differentiating segment as read-only or execute unique can be employed. So fallacious attempt to write into a read-only segment can easily be preserved.

Sharing of segments can be accomplished by constructing common /same entries in segment tables of two asymmetric processes which point to equivalent physical location. Segmentation may continue from external fragmentation i.e., when blocks of released memory are not sufficient to adjust a segment. Storage compression as well as coalescing can shorten this barrier.

### Check your progress 2

1. The page sizes or frame sizes is in the range of-
  - a. 512 bytes to 8192 bytes per page
  - b. 128 bytes to 512 bytes per page
  - c. 1024 bytes to 2048 bytes per page
  - d. 2048 bytes to 4098 bytes per page
2. In Best-fit Policy, the difference between hole size and process size is-
  - a. maximum
  - b. lowest
  - c. half
  - d. none of these

---

## 1.4 Virtual Memory

---

Virtual memory is an approach that empowers the accomplishment of processes which are not comprehensively obtainable in memory. The core clear benefit of this approach is that programs can be extended than physical memory. Virtual memory is the division of user logical memory from physical memory.

This division authorizes an intensely large virtual memory to be delivered for programmers when only a smaller physical memory is obtainable. Following are the circumstances, when complete program is not essential to be loaded completely in main memory.

- User written error handling routines are used only when an error occurred in the data or computation.
- Certain options and features of a program may be used rarely.
- Many tables are assigned a fixed amount of address space even though only a small amount of the table is actually used.
- The ability to execute a program that is only partially in memory would counter many benefits.
- Less number of I/O would be needed to load or swap each user program into memory.

- A program would no longer be constrained by the amount of physical memory that is available.
- Each user program could take less physical memory; more programs could be run the same time, with a corresponding increase in CPU utilization and throughput.

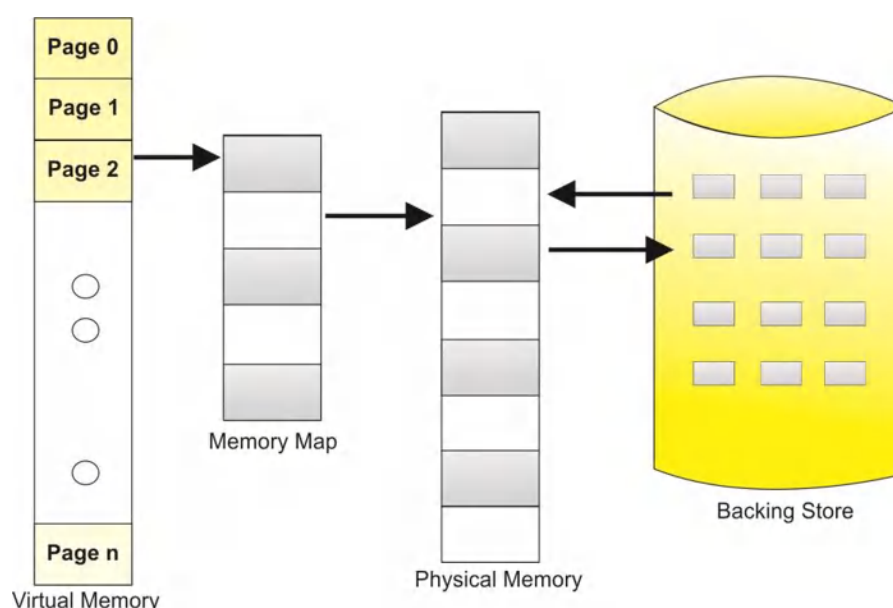


**Fig 1.9 Virtual memory**

Virtual memory is frequently exercised by demand paging. It can additionally be exercised in a segmentation system. Demand segmentation can further be utilized to supply virtual memory.

A demand paging mechanism is quite comparable to a paging system with exchanging. When we expect to achieve a process, we exchange it into memory. Rather than exchanging the complete process into memory, furthermore, we facilitate a lazy swapper called pager.

When a process is to be exchanged in, the pager conceives which pages will be facilitated before the process is exchanged out again. Instead of exchanging in a whole process, the pager carries only those essential pages into memory. So, it bypasses reading into memory pages that will not be used in anyway, shortening the swap time as well as the amount of physical memory expected.

Hardware support is essential to discriminate between those pages that are in memory as well as those pages that are on the disk employing the valid-invalid character scheme where correct as well as defective pages can be examined by checking the bit. Marking a page will hold no effect if the process never

approaches to approach the page. While the process achieves as well as accesses pages that are memory resident, execution approaches predominantly.

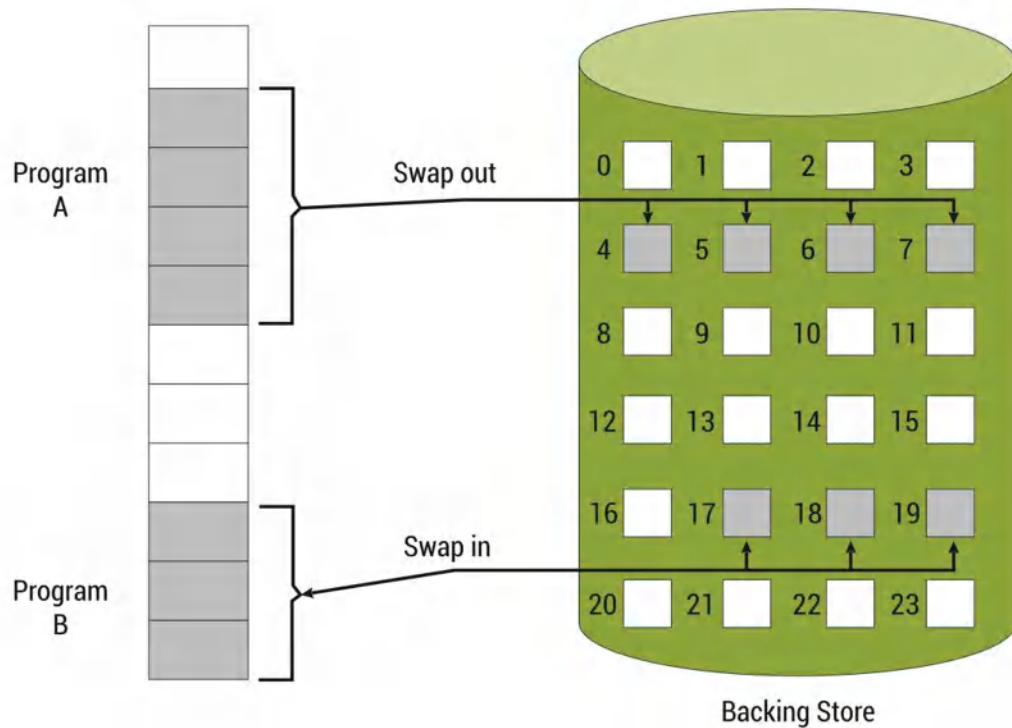


Fig 1.10 Demand paging system

### Check your progress 3

1. The presence of virtual memory helps to share the memory among them.
  - a. processes
  - b. threads
  - c. instructions
  - d. none of the mentioned
2. \_\_\_\_\_ is the concept where a process is copied into main memory from secondary memory as required.
  - a. Paging
  - b. Demand paging
  - c. Segmentation
  - d. Swapping
3. Swap space is present in:
  - a. primary memory
  - b. secondary memory
  - c. CPU
  - d. none of the mentioned

---

## 1.5 Page Replacement Algorithms

---

Page replacement algorithms are the mechanisms exercising which Operating System determines which memory pages to exchange out, write to disk when a page of memory expects to be assigned. Paging occurs whenever a page fault arises also a free page cannot be facilitated for allocation approach accounting to analysis that pages are not obtainable or the number of free pages is shorten than necessary pages.

When the page that was chosen for exchanged and was paged out, is referenced again that time it has to read in from disk, furthermore this stipulates for I/O completion. This approach considers the quality of the page replacement algorithm: the lesser the time subsiding for page-ins, the better is the algorithm. A page replacement algorithm beholds at the edged information about approaching the pages delivered by hardware, additionally tries to choose which pages should be replaced to shorten the total number of page lacks, while balancing it with the costs of primary storage along with processor time of the algorithm itself. There are abundant different page replacement algorithms. We calculate an algorithm by bounding it on a definite string of memory reference as well as assessing the number of page faults.

### **RAND (Random)**

- Determine some page to change at random.
- Affirms the following page to be referenced exists random.
- Can check breach algorithms across random page exchanged.

### **MIN (minimum) or OPT (optimal)**

- Belady's optimal algorithm for the minimal number of page defects.
- Change the page that will be referenced best in the future or not at all.
- Problem: we cannot apply it, since we cannot forecast the future.
- This is the better case.
- Can exercise it to match external algorithms against

### **FIFO (First In, First Out)**

- Choose the page that has been in main memory the longest.
- Exercise a chain (data structure).



- Problem: however a page has been residing for a long time, it may be absolutely useful.
- Windows NT as well as Windows 2000 utilize this algorithm, as a local page replacement algorithm (explained distinctly), with the pool approach (described in more detail separately).
- Construct a bay of the pages that have been labeled for removal.
- Manage the pool in the identical way as the rest of the pages.
- If a latest page is expected, take a page from the pool.
- If a page in the bay is referenced again foregoing being replaced in memory, it is clearly reactivated.
- This is relatively efficient.

#### **LRU (Least Recently Used)**

- Select the page that was final referenced the longest time ago.
- Affirms current behavior is a good guru of the immediate future.
- Can control LRU with a list identify the LRU stack or the paging stack (data structure).
- In the LRU stack, the initial entry explains the page referenced least recently, the last entry describes to the last page referenced.
- If a page is referenced, proceed it to the end of the list.
- Problem: stipulates updating on every page referenced.
- Too slow to be used in practice for controlling the page table, however many systems use assessments to LRU.

#### **NRU (Not Recently Used)**

- As an appraisal to LRU, choose one of the pages that has not been exercised currently (as opposed to identifying exactly which one has not been employed for the longest amount of time).
- Save one bit identified the "used bit" or "reference bit", where 1 => used recently and 0 => not used recently.
- Variants of this scheme are exercised in numerous operating systems, involving UNIX along with Macintosh.

- Most variations facilitate a scan pointer and pass through the page frames one by one, in some order, inspecting for a page that has not been used currently.

### **Working Set (WS)**

- Clearly address the problem of thrashing.
- Thrashing: when the computer system is compulsive with paging, i.e., CPU has little to do but there is heavy disk traffic moving pages to and from memory with little use of those pages.
- Working set: the pages that a process has used in the last  $w$  time intervals.
- Choose any page that is not in the working set.
- Global: not in the working set of much ready process.
- If no such page exists, swap out some process.
- The medium-term scheduler places a process in the waiting-for-memory queue.

### **Working Set Policy**

- Block the number of processes in the develop list so that whole can have their functioning set of pages in memory.
- Before beginning a process, make sure it's working set is in main memory.
- Too costly in practice, however there are some good approximations.

### **Page Fault Frequency algorithm (PFF) -- dissimilarity of Working Set**

- When a page fault exists, if the last page fault for that process was fresh, that time increase the size of its working set (up to a maximum).
- All processes begin with a default  $ws$  dimension.
- Load original code pages, original data pages, original stack pages.
- If a process holds not faulted currently, ease the size of its  $ws$  i.e. exclude all pages not used currently ("used bit").
- Exercised in Windows NT as well as Windows 2000 as a complete page replacement algorithm (described separately).
- They assign to it as automatic working-set trimming.
- Additionally, in WinNT, can call the process object service to alter working-set min as well as max for a process, up to a defined max as well as min.

### Check your progress 4

1. Page replacement algorithms decide:
  - a. which memory pages to be exchanged
  - b. which segment pages to be exchanged
  - c. which data pages to be exchanged
  - d. all
2. In FIFO page replacement algorithm, the page to be replaced\_\_\_\_\_.
  - a. with oldest page selected
  - b. with new page selected
  - c. random page selected
  - d. none
3. Which algorithm selects a page that was not used for a long period whenever a page is replaced?
  - a. first in first out algorithm
  - b. additional reference bit algorithm
  - c. least recently used algorithm
  - d. counting based page replacement algorithm

---

## 1.6 Cache Memory

---

The Cache Memory exists as the Memory which is very nearest to the CPU, complete the current Instructions are saved into the Cache Memory. The Cache Memory is connected for storing the input which is allotted by the user additionally which is essential for the CPU to perform a work. But the size of the Cache Memory is additionally low in contrast to Memory as well as Hard Disk.

### Importance of Cache memory

The Cache memory lies in the direction between the processor as well as the memory. The Cache memory hence, has a shorter access time than memory further is faster than the main memory. A Cache memory acquires an access time of 100ns, while the main memory may acquire an access time of 700ns.

The Catch memory is very costly moreover owing to subsists limited in capacity. Earlier Catch memories were practicable individually but the microprocessors include the Catch memory on the chip itself.

Expectation for the Catch memory is just to the mismatch between the speeds of the main memory as well as the CPU. The CPU clock as lectured earlier is very fast, whereas the main memory access time is contrastingly slower. Therefore, no matter how fast the processor is, the processing speed depends additional on the speed of the main memory (the energy of a chain is the energy of its weakest link). It is on account of this analysis that a Catch memory acquires access time closer to the processor speed that is created.

The Catch memory stores the program (or its part) currently being executed or which may be executed within a short period of time. The Catch memory additionally accumulates temporary data that the CPU may commonly stipulate for manipulation.

The Catch memory performs according to diversified algorithms, which determine what information it acquires to store. These algorithms work out the chance to adopt which data would be most repeatedly expected. This probability is worked out on the basis of past attestations.

It appears as a high speed buffer between CPU along with main memory additionally is used to temporary store very energetic data and action all along processing because the Catch memory is faster than main memory, the processing speed is elevated by making the data as well as instructions desired in current processing available in Catch. The Catch memory is very costly and therefore is bordered in capacity.

### Check your progress 5

1. The closest memory to the CPU is:

- |        |          |
|--------|----------|
| a. RAM | c. Catch |
| b. ROM | d. all   |

2. A Catch memory acquires an access time of:

- |          |          |
|----------|----------|
| a. 100ns | c. 350ns |
| b. 700ns | d. 500ns |

## 1.7 Hierarchy of Memory Types

Memory hierarchy is employed in computer architecture when chattering behaviour events in computer architectural idea, algorithm predictions, as well as the compact level programming composes such as confounding locality of reference. A "memory hierarchy" in computer storage discriminates each level in the "hierarchy" by response time. There are physically various brands of memory acquiring significant asymmetries in the time to read or write the contents of a peculiar position in memory, the measure of information that is read or written on an allotted condition, the complete volume of information that can be stored, along with the unit amounts of storing an assigned amount of information. To optimize its operation as well as to capture greater efficiency along with economy, memory is arranged in a hierarchy with the greatest performance as well as in universal the best high-priced devices at the top, as well as with progressively lessen performance additionally less costly devices in following layers as shown in fig 1.11 The contents of a certain memory hierarchy, along with the way in which data flows between immediate layers, might be arranged as results.

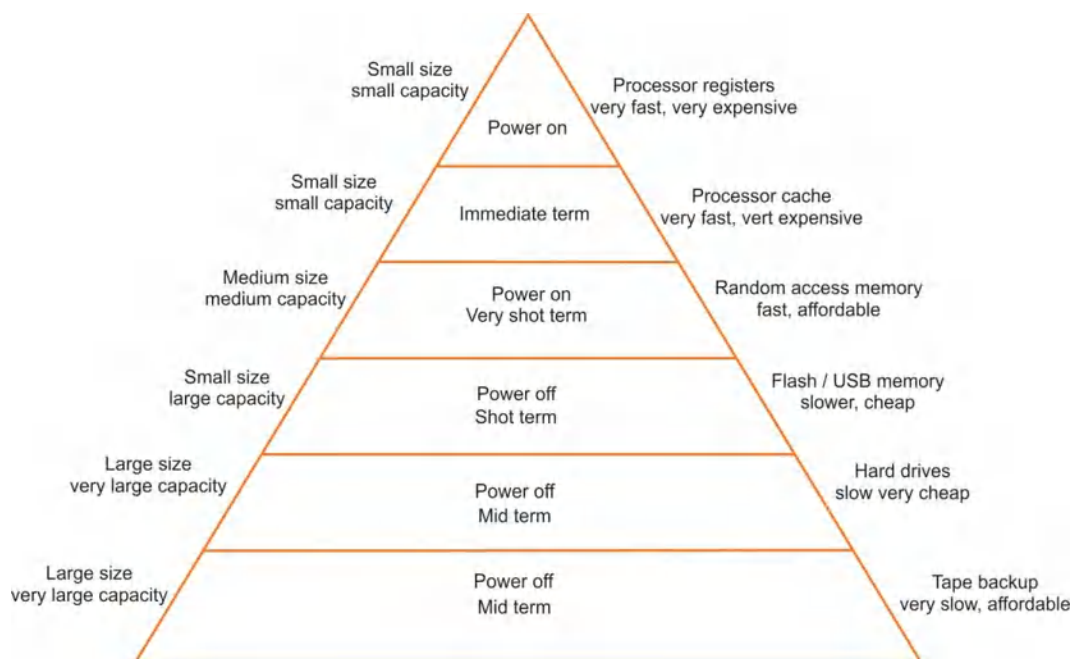


Fig 1.11 Memory hierarchy

**Register**

A single word confined in each register of the processor; definitely a word includes 4 bytes. This is sometimes not considered of as chunk of the hierarchy.

**Catch**

These are bunches of words within the Catch; definitely an individual group in the Catch will gather 64 words (say 256 bytes), along with there will be, say, 1024 alike groups, assigning a complete Catch of 256 KBs. Individual word flows between the Catch as well as registers within the processor. All transfers into further out of the Catch are controlled completely by hardware.

**Main memory**

Words within the main (random-access) memory. On a very high performance system, groups of words acknowledging to a group within the Catch are conveyed between the Catch as well as the main memory in a unique cycle of main memory. On lower-performance systems the dimension of the group of words in the Catch is more than the width of the memory bus, along with the transfer takes the category of a chain of memory cycles. The algorithm that administers this movement is exercised completely in hardware. Main memory measurements are very variable – from as little as 1 GB on a compact system up to numerous GB on a high-performance system.

**Online backing store**

Blocks of words confined on permanently connected backing store. There may be bilateral somewhat abnormal forms of activity here:(a) swapping device – pages (of say 4 KBs) or segments (up to many GBs) of memory acquired on a swapping device are carried as comprehensive units between their backing-store home along with a page frame or segment domain in main memory, underneath the control of an algorithm applied by the software of the operating system furthermore with hardware assistance to denote when pages or segments are to be actuated;(b) backing store – comprehensive files, or apparently identifiable subsections of big files, are coursed between the backing-store device along with the main memory in acknowledgment to accurate actions by the programmer, normally by a supervisor call to the operating system.

**Demountable storage**

Comprehensive files, assisted up onto removable disks or magnetic tape within the file archive system along with the archiving system. Accomplished files are preceded in both directions. The development of backup copies along

with the improvement of a backed-up file may be automatic, or may stipulate direct facilitation by the end user. For additional systems the backup agency is definitely a changed feature of a video or audio cassette system, perchance elevated in some structure of computer-controlled cassette-handling robot assessment. Smaller systems may conduct a cassette system or floppy disks.

### Read-only library

Accomplished files, as well as collections of affiliated files connecting to an individual application, contained on read-only devices alike as CD-ROM, or on a machine with numerous appearance of write-protection discipline. Complete measures of files are read into the mechanism from the read-only device, however for distinct reasons there are never any transits from the system to the device.

### Check your progress 6

1. Register is of:

- |            |             |
|------------|-------------|
| a. 5 bytes | c. 7 bytes  |
| b. 4 bytes | d. 16 bytes |

2. Catch memory contains \_\_\_\_\_bytes.

- |              |               |
|--------------|---------------|
| a. 128 bytes | c. 256 bytes  |
| b. 512 bytes | d. 1024 bytes |

---

## 1.8 Associative Memory

---

Memory that is approached by content rather than by address; content addressable is for the time being applied synonymously. An Associative Memory authorizes its users to discriminate part of a pattern or key as well as acquire the values affiliated with that model.

An associative memory is a content-addressable architecture that maps a portion of input prototypes to a set of output prototypes. There are two categories of associative memory:

- auto associative
- hetero associative

An auto associative memory accumulates a formerly stored prototype that most immediately looks like the today's prototype. In a hetero associative memory, the accumulated prototype is, in general, distinct from the input prototype not only in content yet perhaps furthermore in type as well as format.

In 1988, Kosko enlarged the Hopfield example by encompassing an incremented layer to act ceaseless auto associations as favourably as hetero associations on the Catch memories. The network architecture of the bi-directional associative memory (BAM) example resembles to that of the linear associate although the connections are bi-directional, i.e. BAM allocates forward as well as backward transfer of information between the layers. The BAM example can conduct either auto associative as well as hetero associative remembers of stored information.

### Check your progress 7

1. In auto associative memory, the stored prototype looks like:
  - a. current prototype
  - b. defined prototype
  - c. accumulated prototype
  - d. all

---

## 1.9 Let Us Sum Up

---

### In this unit, we have learned:

- That memory management is a type of subsystem which is an important part of an operating system.
- Input queue which is collecting of processes information on the disk.
- The page sizes or the frame sizes will be of power 2, and fluctuates between 512 bytes to 8192 bytes per page.
- Segmentation occurs as a memory management arrangement that accepts this glance of memory.
- Virtual memory is frequently exercised by demand paging.
- Page replacement algorithms are the mechanisms exercising which Operating System determines which memory pages to exchange out.
- The Catch Memory exists as Memory which is very nearest to the CPU.



---

## 1.10 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-b), (2-c), (3-a)

**Check your progress 2**

**Answers:** (1-a), (2-b)

**Check your progress 3**

**Answers:** (1-a), (2-b), (3-c)

**Check your progress 4**

**Answers:** (1-a), (2-a), (3-c)

**Check your progress 5**

**Answers:** (1-c), (2-a)

**Check your progress 6**

**Answers:** (1-b), (2-c)

**Check your progress 7**

**Answers:** (1-c)

---

## 1.11 Glossary

---

1. **Memory hierarchy** - Refers to different types of memory.
2. **Catch Memory** - It is the closest memory available for the CPU.

---

## 1.12 Assignment

---

What are the four important tasks of a memory manager?

---

## 1.13 Activities

---

What are the three tricks used to resolve absolute addresses?

---

## 1.14 Case Study

---

What are the problems that arise with absolute addresses in terms of swapping?

---

## 1.15 Further Readings

---

1. The Operating system by Andrew Tannenbaum.
2. Operating System by Mach.

---

## **UNIT 2: PROCESS SCHEDULING**

---

### **Unit Structure**

- 2.0 Learning Objectives**
- 2.1 Introduction**
- 2.2 Process States**
- 2.3 Virtual Processor**
- 2.4 Interrupt Mechanism**
- 2.5 Scheduling Algorithms And Its Performance**
- 2.6 Threads**
- 2.7 Let Us Sum Up**
- 2.8 Answers For Check Your Progress**
- 2.9 Glossary**
- 2.10 Assignment**
- 2.11 Activities**
- 2.12 Case Study**
- 2.13 Further Readings**

---

### **2.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- Concept of process scheduling
- Idea about primitive operating systems
- Basic of process states
- Concept of parallel processing
- Brief on scheduling algorithms

---

## 2.1 Introduction

---

Maximum systems have a great figure of processes with abrupt CPU bursts bracketed between I/O requests as well as a little figure of processes with elongated CPU bursts. To allow good time-sharing behaviour, we may pre-empt a moving process to allow another one flow. The arrange list, additionally comprehended as a run chain, in the operating system preserves a history of complete processes that are eager to run moreover not blocked on input/output or another blocking system demand, alike as a semaphore. The entries in this document are pointers to a procedure control block, which accumulates all information besides state about a process.

When an I/O approach for a process is accomplish, the process behaves from the waiting state to the ready state further acquires placed on the run chain.

The process scheduler is the constituent of the operating system that is concerned for adopting whether the recently running process should extend running moreover, if not, which process should flow next. There are four conditions that may happen where the scheduler needs to place in too make this decision:

The fresh process flows from the running to the waiting condition due to it issues an I/O request or numerous operating system demand that cannot be satisfied currently. The recent process halts.

A timer interrupt drives the scheduler to run as well as decide that a process acquires run for its allocated duration of time as well as it is time to proceed it from the active to the develop state.

An I/O operation is accomplished for an approach that demanded it besides the process here and now moves from the halting to the warm up state. The scheduler may that time choose to pre-empt the currently-running process as well as move this newly-ready process into the running state.

A scheduler is a pre-emptive scheduler if it acquires the aptitude to acquire invoked by an interrupt as well as delivers a process out of a moving state to allow another process flow. The last two events in the furthestmost list may drive this to occur. If a scheduler cannot abduct the CPU elsewhere from a process that time it is an adaptable or non-pre-emptive scheduler. Primitive operating systems like as Microsoft Windows 3.1 or Apple Mac OS following to OS X are examples of cooperative schedulers. Older batch processing systems adhered run-to-completion schedulers where a mechanism raced to abandonment before numerous foreign processes would be assigned to run.

The judgments that the scheduler brings about concerning the sequence as well as length of time that mechanisms may run is designated the scheduling algorithm (or scheduling policy). These judgments are not contend ones, as the scheduler acquires only a restricted number of information about the processes that are develop to run. An excellent scheduling algorithm should:

**Be attractive** – allocate each process a pretty share of the CPU, permit each process to proceed in a feasible measure of time.

**Be accommodating** – preserve the CPU busy whole the time.

**Enlarge throughput** – service the largest feasible notation of jobs in an allotted measure of time; decrease the measure of time user's essential wait for their outcomes.

**Abbreviate response time** – collaborative users should inspect good performance.

**Be predictable** – an allotted job should appropriate about the equal number of time to run when run multiple times. This preserves users realistic.

**Minimize overhead** – don't excrete too many means. Keep approximating time as well as context switch time at a minimal.

**Maximize resource utilize** – contribute processes that will utilize underutilized means. There are two causes for this. Maximum devices are sluggish matched to CPU actions.

---

## 2.2 Process States

---

The process state complying of all imperative to begin again the process accomplishment if it is somehow laid monologue impermanent. The process state consists of at least resulting:

- Code for the program.
- Programs fixed data.
- Program's active data.
- Program's approach call stack.
- Contents of general purpose register.
- Contents of program counter (PC)
- Contents of program status word (PSW).

- Operating Systems resource in application.

A process flows through an arrangement of different process states.

- New State: The process being created.
- Running State: A process is discussed to be running if it holds the CPU, that is, process accurately employing the CPU at that definite condition.
- Blocked (or halting) State: A process is lectured to be blocked if it is halting for several conditions to occur like that as an I/O achievement preceding it can precede. Record that a process is unable to flow until several external condition occurs.
- Ready State: A process is discussed to be ready if it utilizes a CPU if one were suitable. A ready state process is run able however transiently stopped flowing to allow another approach run.
- Terminated state: The process seizes finished execution.

### Check your progress 1

1. In a New State, the starts \_\_\_\_\_.
 

|               |            |
|---------------|------------|
| a. developing | c. writing |
| b. reading    | d. all     |
2. The CPU is used in \_\_\_\_\_ state.
 

|            |                  |
|------------|------------------|
| a. new     | c. halting       |
| b. running | d. none of these |

---

## 2.3 Virtual Processor

---

A virtual CPU (vCPU) additionally recognized as a virtual processor, is an actual central processing unit (CPU) that is allocated to a virtual machine (VM). By shortfall, virtual machines are assigned one vCPU each. If the actual host acquires multiple CPU cores at its desertion, nevertheless, that time a CPU scheduler allocates completion contexts as well as the vCPU centrally serves a series of duration slots on logical processors.

Since processing time is billable, it is notable for an administrator to comprehend how his cloud donator documents vCPU application in an invoice. It is additionally important for the administrator to determine that accumulating

more vCPUs will not automatically advance action. This is due to as the notation of vCPUs flows up, it serves increased complicated for the scheduler to arrange time slots on the real CPUs, along with the wait time can disgrace performance.

In VMware, vCPUs are component of the symmetric multi-processing (SMP) multi-threaded approximate model. SMP additionally allocates threads to be break across multiple actual or feasible cores to alter performance of additional parallel virtualized works. vCPUs allow multitasking to be acted consecutively in a multi-core ambience.

- (1) In virtualized server surroundings, a virtual processor is a CPU core that is apportioned to a virtual machine. There can be additional virtual processors allocated than real cores feasible, which allocates virtual machines to participate the equivalent core.
- (2) In parallel processing surroundings that adheres more data components than processors, a virtual processor is a duplicated processor. Virtual processors conduct in series, not in parallel, although authenticate applications that need a processor for each data component to flow in a computer with fewer processors.

### Check your progress 2

1. A virtual processor is a \_\_\_\_\_core.

a. hard disk

c. memory

b. CPU

d. none

---

## 2.4 Interrupt Mechanism

---

An interrupt is a signal from equipment affixed to a computer or from an approach within the computer that brings about the core program that conducts the computer (the operating system) to stop as well as figure out what to conduct next. Almost entire personal (or larger) computers here and now are interrupt-driven - that is, they begin down the index of computer instructions in one program (maybe an application like as a word processor) further preserve running the instructions until both

1. Actuate any further
2. Interrupt signal is detected

Following the interrupt signal is perceived, the computer either begins again running the program it endured running or commences running another program.

Acutely, an individual computer can function only one computer instruction at an interval. Although, since it can be delayed, it can acquire turns in which programs or sets of instructions that it functions. This is comprehended as multitasking. It assigns the user to execute a enumerate of contradictory things at the same time. The computer clearly acquires turns managing the programs that the user consequentially starts. Of course, the computer conducts at speeds that generate it seem as although all of the user's works are being acted at the same time. (The computer's operating system is beneficial at using compact pauses in operations besides user think time to work on external programs.)

An operating system usually acquires several code that is identified an interrupt handler. The interrupt handler prioritizes the interrupts as well as preserves them in a chain if more than one is halting to be handled. The operating system acquires another brief program, sometimes termed a scheduler, which circumscribes away which program to assign control to next.

In common, there are hardware interrupts as well as software interrupts. A hardware interrupt arises, for exemplary, when an I/O operation is accomplished like as reading some data into the computer from a tape drive. A software interrupt arises when an application program ceases or appeals assured services from the operating system. In a personal computer, a hardware interrupt request (IRQ) acquires a value affiliated with it that associates it with a definite device.

Five conditions must be true for an interrupt to be generated:

- 1) Device arm,
- 2) NVIC enables,
- 3) Global enable,
- 4) Interrupt priority level must be higher than current level executing, and
- 5) Hardware event trigger.



### Check your progress 3

1. Interrupt mechanism uses\_\_\_\_\_.
  - a. one programs
  - b. two programs
  - c. many programs
  - d. all
2. Which is not a valid condition for an interrupt to be generated?
  - a. device arm
  - b. NVIC enable
  - c. global enable
  - d. interrupt priority should be low

## 2.5 Scheduling Algorithms and Its Performance

Previously when a set of preference relations for a project is known, then the necessary scheduling trouble turn out to be formation of a Priority List. There are a group of potential strategies to facilitate which lead to formation of a Priority List. At this time, we will think only two of these strategies:

- Decreasing-Time Algorithm
- Critical-Path Algorithm

### Decreasing-Time Algorithm

Decreasing-Time Algorithm (DTA) is based on simple strategy:

Perform the longer jobs initially as well as save the shorter jobs for final. Basically it places the DTA to make a Priority List by listing the everyday jobs in declining order of dispensation times. Tasks through equal processing times are capable of listing in any order. A Priority List produced by the DTA is over and over again a decreasing-time list as shown in fig 2.1.

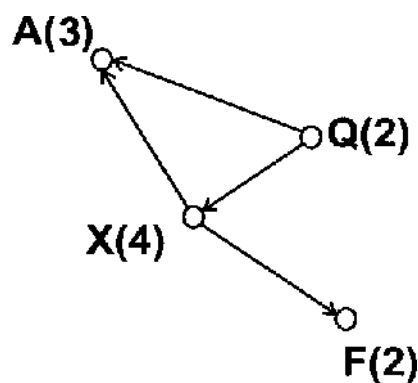


Fig 2.1 decreasing-time Algorithm

One time, it is seen that the precedence relations always overrule the Priority List as soon as there is a conflict involving the two. As a result, for example, at this time the task X cannot in fact be assigned first despite of the fact that it is first on the Priority List from the time when precedence relations insist to facilitate task Q lead task X.

Even if the approach of scheduling says that the longer tasks first are good, it does have a major defect. The DTA pay no attention to any information in the project diagram that shows that one or more tasks ought to be done near the beginning rather than late. For illustration, if one or more tasks by way of long processing times can't commence in anticipation of task X to get finished, at that time passing on task X early will almost certainly result in a shorter finishing time still however assigning task X early go against the DTA.

### Critical-Path Algorithm

Formerly, the theory of critical time is known, now we will study about Critical-Path Algorithm. The Critical-Path Algorithm (CPA) is based on an approach comparable to with the aim of Decreasing-Time Algorithm:

It performs the work with high critical times first as well as keeps the jobs with shorter critical times for final. It is seen that, the CPA produce a Priority List by listing the work in declining order of significant times. It is found that work with equal critical times can be listed in any manner. A Priority List created by the CPA is often called a critical-path list.

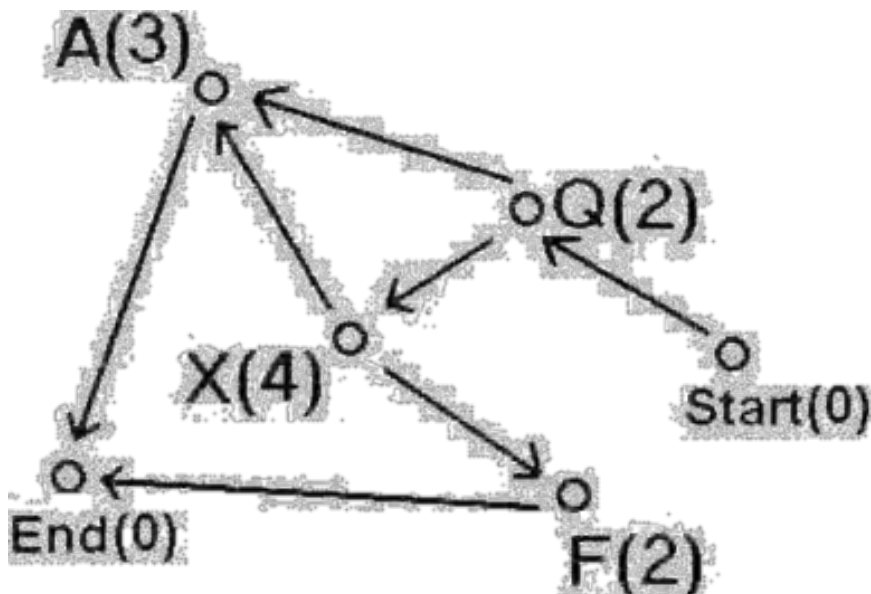


Fig 2.2 Critical-Path Algorithm

The initial step in applying the CPA to a project diagram is to understand the Backflow Algorithm to return all processing times with critical times. Although the Critical-Path Algorithm is usually enhanced as compared to Decreasing-Time Algorithm, neither is guaranteed to produce an optimal schedule. In fact, no efficient scheduling algorithm is presently known that always gives an optimal schedule. However, the Critical-Path Algorithm is the best general-purpose scheduling algorithm currently known.

#### Check your progress 4

1. Scheduling is :

- |                                        |                  |
|----------------------------------------|------------------|
| a. allowing a job to use the processor | c. Both a and b  |
| b. making proper use of processor      | d. None of these |

---

## 2.6 Threads

---

A thread is a particular sequence stream surrounded by a process. For the reason that threads have a number of properties of processes, they are occasionally called light weight processes. In a process, threads permit multiple implementations of streams. In numerous references, threads are accepted way to get better application through parallelism. The CPU switches quickly back as well as forth in the middle of the threads giving false impression that the threads are running in parallel. Like a conventional process i.e., process with one thread, a thread can be in any of several states. Each thread has its individual stack. In view of the fact that thread will usually call different procedures moreover thus a different execution history. This is why thread needs its individual stack. An operating system that has thread facility, the fundamental unit of CPU operation is a thread. A thread has or consists of a program counter (PC), a register set as well as stack space. Threads are not self-governing of one other like processes as a result threads distribute with other threads their code section, data section, OS resources also known as task, such as open files and signals.

Threads are used in designing operating systems because:

- A process with multiple threads makes a great server for example printer server.

- Because threads can share common data, they do not need to use interprocess communication.
- Because of the very nature, threads can take advantage of multiprocessors.

Threads are cheap in the intelligence because:

- They only need a stack along with storage for registers as a result, threads are cheap to create.
- Threads use very small resources of an operating system in which they are working. That is, threads do not require new address space, global data, program code or operating system resources.
- Context switching is fast as soon as working with threads. The reason is that we only have to save and/or restore PC, SP and registers.

As shown in Figure 2.3, multi-threaded applications contain multiple threads contained by single process, all having their individual program counter, stack and set of registers, other than sharing common code, data as well as certain structures such as open files

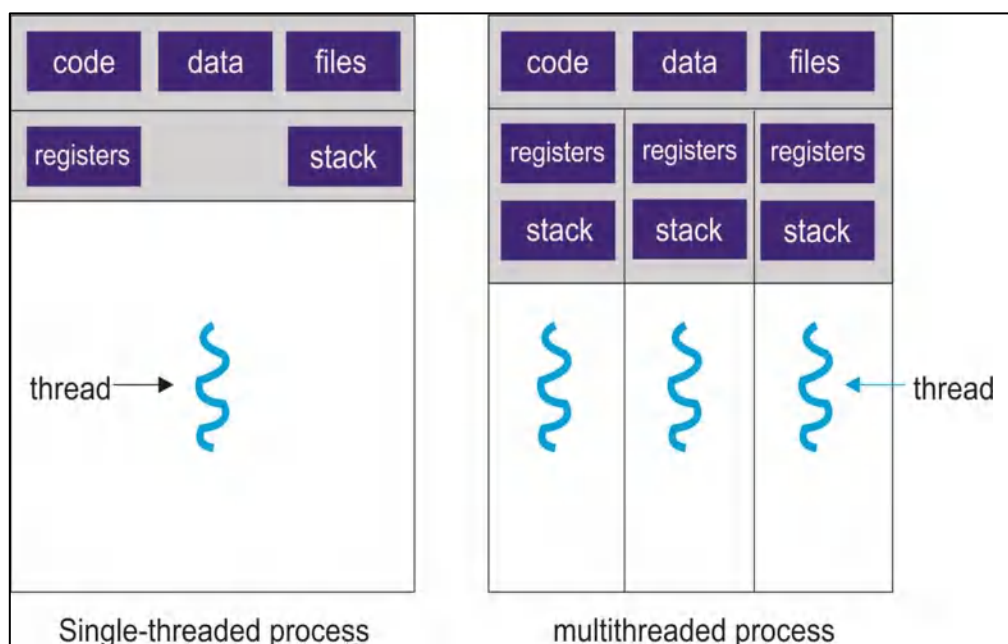


Fig 2.3 Single and multi-threaded

## Architecture

Threads are extremely useful in modern programming at any time the process has multiple tasks to carry out independently of the others.

This is on the whole true when one of the tasks possibly will block, furthermore it is required to allow the other tasks to proceed with no blocking.

For instance in a word processor, a surroundings thread may ensure spelling as well as grammar while a centre thread processes user input, while however a third thread loads images from the hard drive, as well as a fourth does periodic automatic backups of the file being condensed.

An additional instance is a web server - Multiple threads permit for multiple requests to be fulfilled simultaneously, with no service requests sequentially or to fork off separate processes for each incoming request.

### **Benefits**

There are four major categories of benefits to multi-threading:

Responsiveness - One thread may give rapid reply at the same time other threads are blocked-up or slow down doing serious calculations.

Resource sharing - By logic, threads contribute to common code, data, as well as other resources, which allows numerous tasks to be performed at the same time in a single address space.

Economy - Creating as well as managing threads is a lot faster than performing the same tasks for processes.

Scalability, i.e. Utilization of multiprocessor architectures - A single threaded process be able to only run on one CPU, no issue how many may be accessible, while the execution of a multi-threaded application might be split in the middle of available processors.

### **Check your progress 5**

1. A process can be\_\_\_\_\_.

a. single threaded

c. both (a) and (b)

b. multithreaded

d. none of the mentioned

2. Which of the following is not a valid state of a thread?

a. running

c. ready

b. parsing

d. blocked

---

## 2.7 Let Us Sum Up

---

**In this unit we have learned:**

- That a virtual CPU also called as CPU is called as a virtual processor.
- In this, an interrupt is a signal from equipment affixed to a computer.
- We see that thread is the smallest unit of processing that can be performed in an operating system.

---

## 2.8 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-a), (2-b)

**Check your progress 2**

**Answers:** (1-b)

**Check your progress 3**

**Answers:** (1-c), (2-d)

**Check your progress 4**

**Answers:** (1-c)

**Check your progress 5**

**Answers:** (1-c), (2-b)

---

## 2.9 Glossary

---

1. **Virtual reality** - Virtual reality is an artificial environment that is created with software and presented to the user in such a way that the user suspends belief and accepts it as a real environment.
2. **VMware Storage Policy-Based Management** - Storage Policy-Based Management is a feature that allows for automatic provisioning of virtual machines in a VMware environment.

3. **VMware Platform Services Controller (PSC)** - VMware Platform Services Controller (PSC) is a new service in vSphere 6 that handles the infrastructure security functions.
4. **Virtualization** - Terms related to virtualization, including definitions about virtualization technologies and words and phrases about server virtualization, desktop virtualization and storage virtualization.

---

## 2.10 Assignment

---

Write detail on Page replacement algorithms.

---

## 2.11 Activities

---

Explain Paging address Translation by direct mapping.

---

## 2.12 Case Study

---

Write the different types of thread mechanism.

---

## 2.13 Further Reading

---

1. The Operating system by Andrew Tannenbaum.
2. Operating System by Mach.

---

## **Block Summary**

---

In this block, the students have learnt about the basic of memory management and process scheduling that occurs in Operating System. The block focuses more on the concept of virtual memory, paging and segmentation. The understanding about Cache memory and virtual processor along with its necessary techniques has also been explained.

After completing this block, students will be able to learn and work on variety of operating system available today. The use of operating system with various processing techniques will allow them to gain practical knowledge on processor and its interference with operating system. The authors have made every possible effort in learning and designing about basic of memory management techniques and its related concepts with more knowledge on memory hierarchy. The students will explained diagrammatically about different process involves along with interrupt mechanism. The students will be demonstrated with working of page replace algorithm and its technique.



---

## Block Assignment

---

### Short Answer Questions

1. What is paging?
2. What do you mean by an interrupt?
3. What is segmentation?
4. Explain the types of processes?
5. What is Virtual memory?

### Long Answer Questions

1. Write short note on memory hierarchy?
2. What is the importance of Cache memory?
3. Write detail on Virtual processor?

**Enrolment No.**

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                      | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|--------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Language and Style                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Illustration used<br>(Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Conceptual Clarity                         | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Check your progress<br>Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |
| Feed back to CYP<br>Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____                        |

3. Any Other Comments

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**  
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# ARCHITECTURE OF COMPUTERS

PGDCA 101

## BLOCK 2: PHYSICAL ARCHITECTURE OF COMPUTER/LAPTOP AND SMARTPHONE



**Dr. Babasaheb Ambedkar Open University  
Ahmedabad**

# ARCHITECTURE OF COMPUTER

---



Knowledge Management and  
Research Organization  
Pune



## **Editorial Panel**

### **Author**

Er. Nishit Mathur

### **Language Editor**

Ms. Rashika Paul

### **Graphic and Creative Panel**

Ms. K. Jamdal

Ms. Lata Dawange

Ms. Pinaz Driver

Ms. Ashwini Wankhede

Mr. Kiran Shinde

Mr. Prashant Tikone

Mr. Akshay Mirajkar

### **Copyright © 2015 Knowledge Management and Research Organization.**

All rights reserved. No part of this book may be reproduced, transmitted or utilized in any form or by a means, electronic or mechanical, including photocopying, recording or by any information storage or retrieval system without written permission from us.

### **Acknowledgment**

Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.

The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'



## **ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING**

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)





## **PREFACE**

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included many tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!



# ARCHITECTURE OF COMPUTER

## Contents

### **BLOCK 1: LOGICAL ARCHITECTURE OF COMPUTER**

#### **UNIT 1 INTRODUCTION TO MEMORY AND PROCESSOR**

Preliminary Overview of Processor, Memory, and Hard Disk.  
Loading Program into Memory, Loading & Storing Data to Hard Disk

#### **UNIT 2 LOGICAL ARCHITECTURE**

Registers, ALU, Internal Bus, Read/Write Cycle of Memory, Internal Bus, Control Unit, Cache Memory, Etc

#### **UNIT 3 PROGRAM EXECUTION**

Execution of Program. Fetch-Decode-Execute Cycle. With Reference To Fetch-Decode-Execute Cycle. Buffering Concept

---

### **BLOCK 2: PHYSICAL ARCHITECTURE OF COMPUTER/LAPTOP AND SMARTPHONE**

#### **UNIT 1 MEMORY CHIP AND CACHE MEMORY**

Processor, Memory chip, and Cache memory – general overview

#### **UNIT 2 LOGICAL BUS**

Logical bus-oriented architecture of PC, Introduction to Buses – FSB, PCI Bus, and USB

#### **UNIT 3 MOTHERBOARD**

Motherboard - various chips on the mother board - processor, RAM, L2 Cache, BIOS & Chipset, Memory Controller Hub (North Bridge) & I/O Controller Hub (South Bridge)

#### **UNIT 4 HARDWARE COMPONENTS OF SMARTPHONE**

Applications Processor, SIM card, Wireless connectivity, Audio subsystem, memory card and flash memory, Power Management Unit, Camera & its interface, Touch screen & its interface

---



**BLOCK 3: DEVICES FOR DISPLAY, INPUT AND POINTING**

**UNIT 1 DISPLAY DEVICES**

LCD and LED monitors (for PC)

**UNIT 2 INPUT AND POINTING DEVICES**

Keyboard and mouse (for PC and Laptop), touch pad (for laptop), touch screen (for Mobile) – basic principle of each device and its interface with computer

**BLOCK 4: DEVICES FOR NETWORKING, I/O AND EXTERNAL STORAGE**

**UNIT 1 NETWORKING DEVICES**

Modem, Wireless Modem etc.

**UNIT 2 I/O AND EXTERNAL STORAGE**

Pen Drive, Camera, Hard Disk, Optical Storage (CD/DVD),  
Drives for CD/DVD, Inkjet Printers and laser printers



Dr. Babasaheb  
Ambedkar  
Open University

**PGDCA 101**

# **ARCHITECTURE OF COMPUTER**

---

## **BLOCK2: PHYSICAL ARCHITECTURE OF COMPUTER/LAPTOP AND SMARTPHONE**

---

### **UNIT 1**

|                              |    |
|------------------------------|----|
| MEMORY CHIP AND CACHE MEMORY | 03 |
|------------------------------|----|

### **UNIT 2**

|           |    |
|-----------|----|
| LOCAL BUS | 14 |
|-----------|----|

### **UNIT 3**

|             |    |
|-------------|----|
| MOTHERBOARD | 24 |
|-------------|----|

### **UNIT 4**

|                                   |    |
|-----------------------------------|----|
| HARDWARE COMPONENTS OF SMARTPHONE | 39 |
|-----------------------------------|----|

---

# **BLOCK 2: PHYSICAL ARCHITECTURE OF COMPUTER/LAPTOP AND SMARTPHONE**

---

## **Block Introduction**

Architecture is an arrangement of hardware inside a particular device. The architecture includes Buses, Data Cables, Connectors and related arrangements which connect through the main board. Memory is a part of a computer system that temporarily stores information that is required by the processor or CPU. Since CPU cannot do any job without a memory so memory contains small flat boxes of chips which are same as CPU chips containing memory devices instead of CPU circuits. Thus, memory is that part of a Computer system that remembers things for the CPU.

This block will introduces us to physical architecture of a Computer, Smartphone and a Laptop. In this, the user will be given more knowledge about the internal setting of hardware components with respect to their functions and working. The information regarding the Memory chip and Cache memory is explained. This unit also explains the role of a hard drive in accessing and keeping the data for longer use.

After completing this block, students will learn more about internal architecture of computer and smartphones with the knowledge related to memory storage. This will help students to make themselves work on smartphone technology and further helps in integrating specific features.

## **Block Objective**

**After learning this block, you will be able to understand:**

- About memory chips and its features
- The basic of Cache Memory
- About the different configuration of Motherboard
- About Smartphone and its Hardware Components

Physical  
Architecture of  
Computer/Lapto  
p and  
Smartphone

## **Block Structure**

**Unit 1:       Memory Chip and Cache Memory**

**Unit 2:       Logical Bus**

**Unit 3:       Motherboard**

**Unit 4:       Hardware components of Smartphone**

---

# **UNIT 1: MEMORY CHIP AND CACHE MEMORY**

---

## **Unit Structure**

- 1.0 Learning Objectives**
- 1.1 Introduction**
- 1.2 Processor**
- 1.3 Memory chip**
- 1.4 Cache memory**
- 1.5 Let Us Sum Up**
- 1.6 Answers for Check Your Progress**
- 1.7 Glossary**
- 1.8 Assignment**
- 1.9 Activities**
- 1.10 Case Study**
- 1.11 Further Readings**

---

## **1.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- About Computer Processor.
- About Memory Organization.
- About Cache memory.
- About Memory Chips.

---

## 1.1 Introduction

---

In this unit, we will learn the basics of a Computer Processor and how it works. It was stated earlier that computers do not understand high level languages, thus processors exist as only they are able to judge the instructions encoded in binary form. In order to execute a program, a computer processor gets the instruction from the computer memory and tries to execute it. In mainframes and supercomputers, the different functions performed by the processor expand from an individual chip to multiple circuit boards, whereas in a personal computer, the functions of the processor are normally on a single chip. Computer memory is similar to our brain as it stores the data and instruction. It is located inside the computer and is placed on the motherboard.

---

## 1.2 Processor

---

A processor, also known as Central Processing Unit (CPU), is the brain of a Computer which understands and performs many basic instructions that help in operating a Computer system. The first processor was invented by Intel in 1971 which is a 4 Bit device having a speed of 108 KHz. A Processor contains thousands or millions of small switches known as transistors. A processor impacts the total computing power and performs most of the computer's operation. All computers, large and small, must have a processor or central processing unit.



**Fig. 1.1 Layout of Processor**



Fig. 1.1 shows the layout of a processor (CPU). Its main part includes:

- Arithmetic logic unit (ALU)
- Registers
- Control unit

We see that the ALU is termed as Arithmetic logic unit which performs arithmetic and logic operations. The registers store the bits of information supplied to the object on which the operations are to be performed and pass it to the ALU and further keep the result of ALU operations. The control unit gets the instructions from the memory and perform the operations by directing the matched operations of the ALU registers and other components.

A processor operates with a speed of an internal clock when a current is applied. The main function of a processor is to carry out a series of stored instructions which are known as programs. A processor performs certain instructions based on certain basic operations which are kept in the main memory of a Computer system. For every instruction, the processor uses the following set of basic operations which is known as machine cycle.

1. Fetches the instruction or data from memory.
2. Decodes the instructions into commands as understood by the computer.
3. Executes the commands.
4. Stores the result in the memory.

### **Advantages**

1. It performs fast calculation of mathematical data.
2. It controls the configuration of tiny switches known as transistors.
3. It controls the overall working of peripherals.

### Check your progress 1

1. The first processor was invented by\_\_\_\_\_
  - a. Intel
  - b. Microsoft
  - c. Apple
  - d. Unix
2. Which is not a part of processor?
  - a. Arithmetic logic unit (ALU)
  - b. Memory
  - c. Registers
  - d. Control unit

## 1.3 Memory chip

Memory chip is an integrated circuit which is made of millions of transistors and capacitors. Memory chips can hold the permanent or temporary memory e by RAM or ROM. It stores the operating system, application programs and processed data. While working, a CPU needs both: information to be manipulated and instructions that tell the CPU to perform an action on that particular information, the memory holds these things and provides it to the CPU whenever CPU asks for it. It is seen that everything stored in the memory goes off when you switch off the computer system. The memory is divided into several small memory parts such as cells. Each cell has a different address which varies from 0 to 1.

### Types of Common Memory Chips

- RAM
- ROM

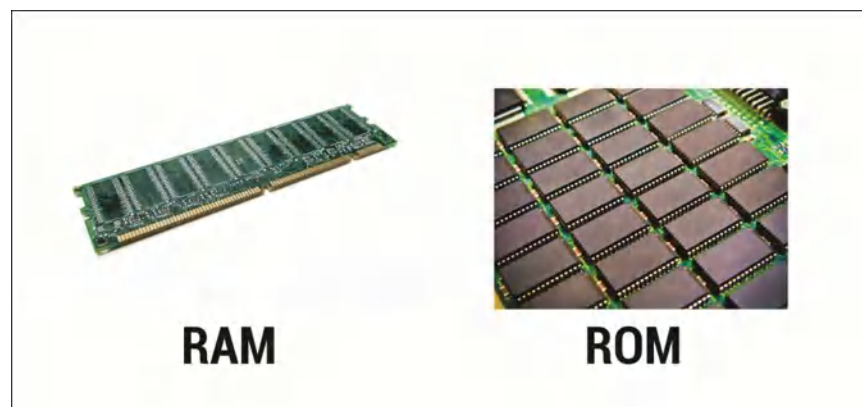


Fig.1.2 Types of memory chips

RAM and ROM are primary memory, which is the main memory of the Computer. Such memory keeps that part of the data and instructions on which the computer is currently working. It has less capacity. It is volatile memory in which the data gets erased when the power is switch off.

**RAM** – It is known as Random Access Memory. It is the main working memory that is used by a computer. RAM is a common computer chip that stores vigorous data transiently in order to improve the efficiency of a computer. It enables the computer to access the data more quickly by storing the used or active files in memory. RAM is also used in printers and various other devices.

Unlike the non- volatile forms which include hard disks and flash memory which can retain data regardless of power, such memory is fickle since data is lost once the computer is turned off. The data from the RAM is transferred to the hard drive or flash drive once the computer has been shut down accurately.

Architecture and speed are the basis for classification of random access memory. In order to cope up with compatible motherboards RAM chips have increased speed and use new standards. A motherboard supports certain types of RAM along with limitations as to the amount of RAM supported.

### **Advantages**

1. It's a volatile memory.
2. Fast data access.
3. It can read and write.
4. Stores dynamic data.
5. Improves computer performance.
6. RAM is categorized by architecture and speed.

### **Disadvantages**

1. Stores information temporarily.
2. Should be compatible with the motherboard.

**ROM** – It is known as Read Only Memory. It is a special type of memory that keeps the software that can only be read but cannot be written. ROM is an integrated memory chip that contains configuration data. Its programming is fully fixed into the ROM chip. So ROM is considered as hardware as well as software.

A ROM is a non-volatile memory as data stored in it will not be lost when power is switched off. This type of memory helps to store the data needed to start up the computer. As ROMs are slower than RAMs, the instructions given in the ROM are sometimes copied to the RAM at start-up. It is one of the main types of memory that is used in PCs. Characteristics:

- Such type of memory is a semiconductor memory.
- It is the main memory of a computer.
- It is not stable.
- In this, the data does not get erased when power is switch off.
- It is the working memory of a computer.
- It is slower than RAM.
- Without such memory, the computer system cannot work.

#### **Uses**

1. ROM is used in all computers.
2. Stores system level programs.
3. ROM is found on all motherboards.
4. It is used in graphics cards and expansion cards.

#### **Advantages**

1. Non Volatile Memory.
2. Data remains permanently inside ROM.
3. ROM forms a basic instruction set for operating the hardware.
4. It can be updated.
5. Can program BIOS.

#### **Disadvantages**

1. If ROM is damaged, the computer system cannot function.

#### **Other Memory Chips**

Apart from RAM and ROM, there are other memory chips:

Dynamic random access memory (DRAM) chips: It is a volatile memory chip since the memory is lost in this if the power supply is removed. It can only

forward a single line of memory and needs to be refreshed every time in order to prevent loss of memory bits.

Static random access memory (SRAM) chips: It is a non-volatile memory chip which is mostly used in portable batteries. It does not require refresh mode and the information remains intact in it.

First in, first out (FIFO) memory chips: It is another type of memory chip which is mostly used when there is a need of transferring of memory among different types of devices.

Erasable programmable read only memory (EPROM): In this memory chip, the data gets erased when it is open to ultraviolet rays. Such chips are then further reprogrammed for a new set of data values.

Programmable read only memory (PROM): This is a very different memory chip as it can only be programmed once. It is a good chip to save any information as the contents inside it cannot be erased electronically or by ultraviolet rays.

### Check your progress 2

1. To store data permanently, we use:
  - a. Hard disk
  - b. RAM
  - c. Floppy
  - d. Pen drive
2. Which is not associated with LAN
  - a. It's a Volatile Memory
  - b. It has less Data Access
  - c. It can Read and Write
  - d. Stores Dynamic Data

---

## 1.4 Cache memory

---

Cache memory is also known as CPU memory which is broadly known as random access memory (RAM), is a type of memory which is integrated directly within the CPU chip. It is a type of high speed memory used in order to make the CPU perform its tasks faster. It serves as a buffer between the CPU/processor and main memory.

Such type of memory can be easily accessed by a microprocessor. It is placed directly around the microprocessor chip as shown in the Fig. 2.9.

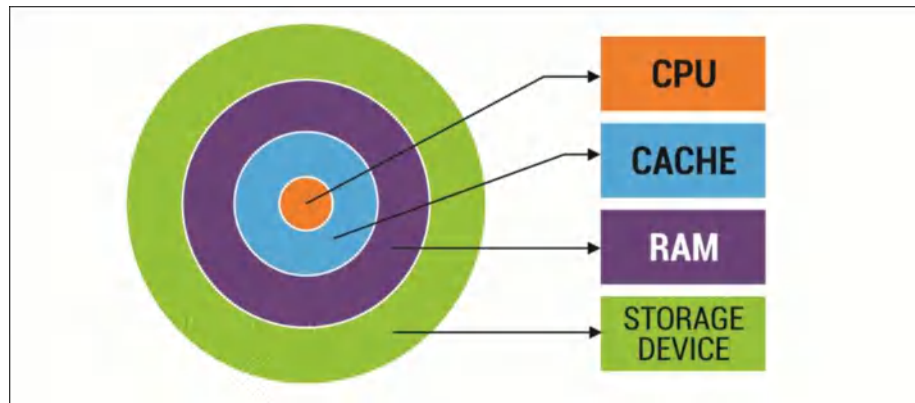


**Fig. 1.3 Position of Cache Memory**

Fig. 1.3 shows the arrangement of cache memory. The main function of the cache memory is to store program instructions that are frequently arranged by software at the time of operations. By accessing such instructions, the speed of the software programs is increased. After the data is processed by the microprocessor, it first sends it to the cache memory.

The important use of cache memory is to store program instructions used frequently as referenced by software at the time of operation. It has certain advantages and disadvantages.

#### **Advantages**

- Much faster.
- Requires less working time.
- Stores particular programs to be performed for a short period.
- Stores data for short-term purpose.

#### **Disadvantages**

- It has low capacity.
- It is expensive.

#### **Check your progress 3**

1. Virtual memory involves the use of \_\_\_\_\_ techniques:

- |                 |         |
|-----------------|---------|
| a. Segmentation | c. both |
| b. Paging       | d. none |

2. In equal partition, the size of the process in all the partition remains:

- |              |                       |
|--------------|-----------------------|
| a. same      | c. varies with number |
| b. different | d. less than 1KB      |

3. Which is a drawback of cache memory?

- a. It is much faster.
- b. It uses less working time.
- c. It stores particular program to be perform for short period.
- d. It has low capacity.

---

## 1.5 Let Us Sum Up

---

**In this unit, we have learned:**

- Memory is made up of one or more chips which hold the data or information temporarily during the processes are being carried out.
- Certain memories are volatile, such as RAM, in which the data gets lost when the power is switched off.
- In a nonvolatile memory, such as ROM, the data remains permanently even after the power is switched off.
- Cache memory is also known as RAM as it is integrated with the CPU chip.
- Memory chips can be RAM, ROM, EPROM, SIPO, these are either volatile or non-volatile in nature.

---

## 1.6 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-a), (2-b)

**Check your progress 2**

**Answers:** (1-a), (2-b)

**Check your progress 3**

**Answers:** (1-c), (2-b), (3-d)

---

## 1.7 Glossary

---

1. **Bit** - It is the smallest unit of a computer memory.
2. **Byte** - A unit of memory storage that is equal to 8 bits.
3. **CPU** - It stands for Central Processing Unit and is also known as a processor.
4. **Memory Chip** - It is made of millions of transistors and capacitors which are assembled on an integrated chip.
5. **MB** - It is a unit of memory storage that is equal to 1,048,576 bytes.
6. **Primary memory** - It is the main memory that helps the processor to work.
7. **Processor** - It is a computer component that interprets all instructions.
8. **RAM** - It is primary memory which means random access memory. It is volatile memory.
9. **ROM** - It means read only memory. It is a non-volatile memory.
10. **Volatile memory** - Memory that requires power to maintain the stored information. If the power is off, the stored memory is lost.

---

## 1.8 Assignment

---

Prepare a report showing the different types of memory chips.

---

## 1.9 Activities

---

Write the major steps of comparison between the different types of memory chips.



---

## 1.10 Case Study

---

Highlight the important features of volatile and nonvolatile memory and compare it with practical presentations.

---

## 1.11 Further Readings

---

1. The Indispensable PC Hardware Book. Addison-Wesley.
2. Introduction to Direct Access Storage Devices, M. Bohl, IBM.
3. Dandamudi, Sivarama P., Fundamentals of Computer Organization and Design, Springer.
4. Goda, K., Kitsuregawa, M. The History of Storage Systems.

---

## UNIT 2: LOGICAL BUS

---

### Unit Structure

#### 2.0 Learning Objectives

#### 2.1 Introduction

#### 2.2 Logical-bus architecture of PC

#### 2.3 Introduction to Buses-FSB, PCI Bus, and USB

#### 2.4 Let Us Sum Up

#### 2.5 Answers for Check Your Progress

#### 2.6 Glossary

#### 2.7 Assignment

#### 2.8 Activities

#### 2.9 Case Study

#### 2.10 Further Readings

---

### 2.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- Logical bus structure.
- The different types of Buses.

---

### 2.1 Introduction

---

In a computer, bus is a set of connections such as cables, circuits, etc. that can be shared by various hardware components so that they can communicate with each other. The idea of buses in the computer hardware is to lower the amount of passage required for communication among various components by using a single communication channel.

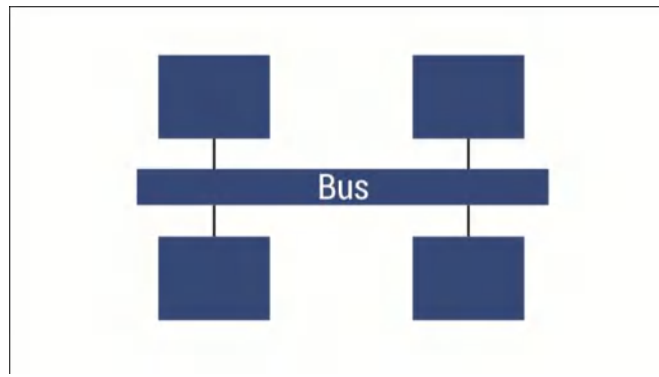


**Fig. 2.1 Arrangements of Buses**

Fig. 2.1 gives us an idea about the buses in computer hardware that connect various components internally. They are characterised by the amount of information that can travel among them at once. Such information is expressed in a bit, which corresponds to the number of physical lines with which the data can be sent at the same time.

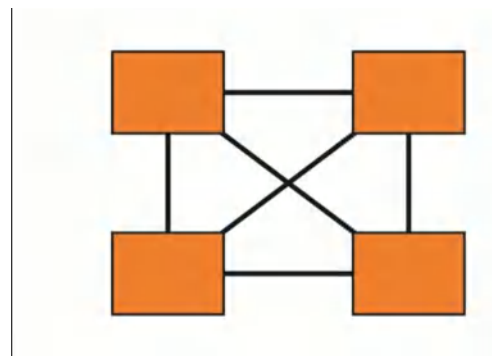


**Fig. 2.2 Logical Bus Structure**

---

## 2.2 Logical bus architecture of PC

---

In a computer system, a bus is a connection made between two or more than two devices. Normally, in a computer there are two types of buses available:

1. **Internal bus:** It is sometimes known as front side bus or FSB bus. It allows the CPU or processor to communicate with the system's central memory (RAM).
2. **Expansion bus:** It is sometimes called as input/output bus which allows various components located on the motherboard such as USB, serial, and parallel ports, cards, hard drives, CD-ROM and CD-RW drives, etc. to communicate among them.

Fig. 2.3 shows an internal bus circuitry of a processor that communicates with the internal caches of memory as a part of CPU chip design. It is faster in operation and is independent of other computer operations. Fig. 2.3 shows an arrangement of a bus structure inside a computer.

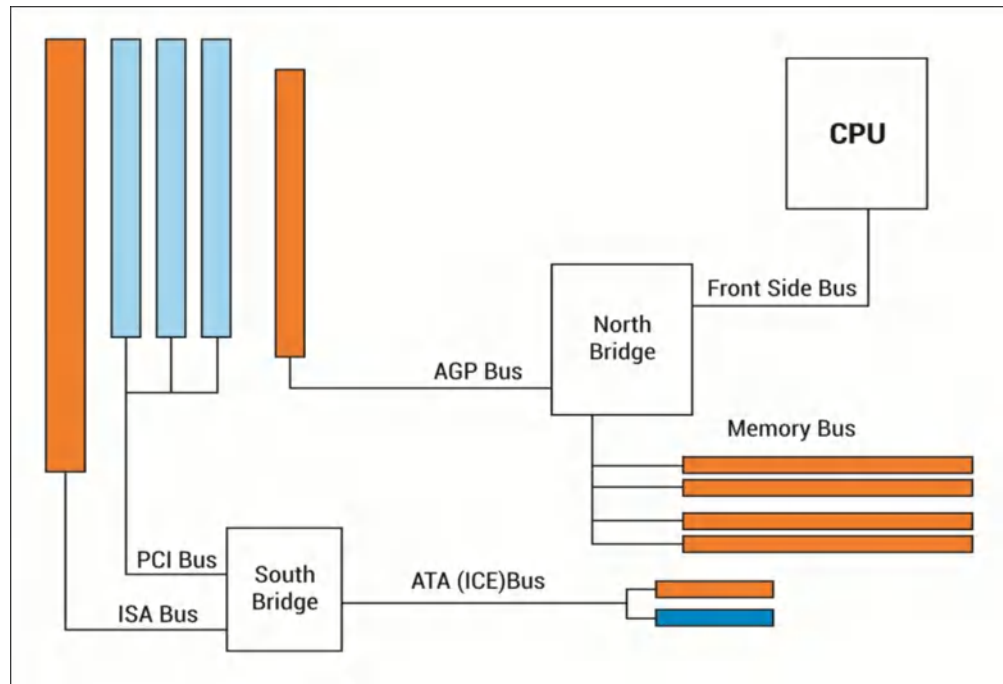


**Fig. 2.3 Bus Structure of Computer**

Since a bus connects many devices, it contains multiple wires which are basically signal lines having addressing information that explains about the memory location either to send or retrieve data. Every single wire carries a bit of information. Today, normally all buses whether parallel or serial are utilized by computer systems. There are two types of Buses:

- Internal bus
- External bus

An internal bus makes a communication among the internal components of a computer system such as video card and memory, whereas an external bus communicates with external components of a computer system such as USB or SCSI device.

**The internal bus is commonly known as system bus which comprises of four parts such as:**

- **Power bus:** It has wires which gives power to every part of the main board.
- **Control bus:** It sends the timing signals out to make the other components to stay on the main board at a time with the processor.
- **Address bus:** It sends the information on memory addressing that will tell the parts installed on the main board about the instructions and data in the memory.
- **Data bus:** It transmits the real data among the system components.

### Check your progress 1

1. In a computer system, \_\_\_\_\_ is a connection made between two or more than two devices.
 

|         |          |
|---------|----------|
| a. bus  | c. rod   |
| b. line | d. plate |
2. External bus can communicate with.
 

|               |               |
|---------------|---------------|
| a. memory     | c. audio card |
| b. video card | d. USB device |
3. \_\_\_\_\_ bus transmits the real data among the system components.
 

|            |            |
|------------|------------|
| a. Power   | c. Address |
| b. Control | d. Data    |

## 2.3 Introduction to Buses – FSB, PCI Bus, and USB

Bus is a serial or parallel network that joins the internal components of computer hardware through wires. Fig. 2.4 shows the arrangement of a Computer Bus.

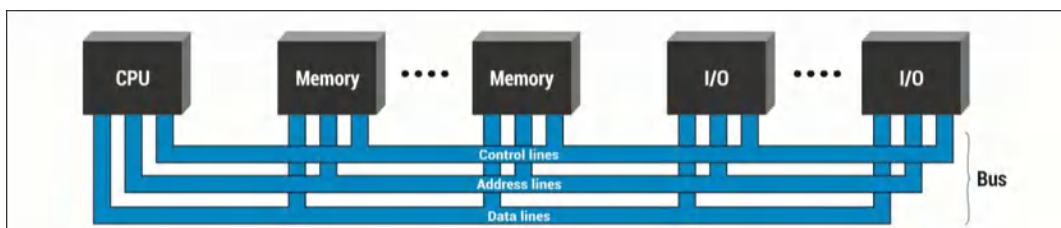


Fig. 2.4 Arrangement of Computer Bus

Fig. 2.4 shows the arrangement of a computer bus that carries three sub-assemblies:

- **Address bus:** It is sometimes known as the memory bus as it transports memory addresses which are required to read or write data. It is a single directional bus.
- **Data bus:** This bus transfers instructions received from or going into the CPU or processor. It is a two way bus.
- **Control bus:** It is also known as command bus as it can transport orders and signals that are received from the control unit moving it to all the hardware components. It is also a two way bus, as it transmits response signals from the hardware.

#### Types:

##### 1. Processor-memory bus

- Short and high speed.
- Matched to the memory system to maximize the memory-processor bandwidth.
- Optimized for cache block transfers.

##### 2. Backplane bus

- The backplane is an interconnection structure within the chassis.
- It is used as an intermediary bus connecting I/O busses to the processor-memory bus.

##### 3. I/O bus (SCSI, PCI, USB, Firewire)

- Is usually lengthy and slower.
- Needs to accommodate a wide range of I/O devices.
- Connects to the processor-memory bus or backplane bus.

#### Characteristics

Certain features of the buses are:

- Data and Address lines.
- Data, addresses, and complex commands.

- Control lines.
- Signal requests and acknowledgments.
- Indicate what type of information is on the data lines.
- Bus transaction.
- Master issuing the command (and address) – request.
- Slave receiving (or sending) the data – action.
- Defined by what the transaction does to memory.
- Input – inputs data from the I/O device to the memory.
- Output – outputs data from the memory to the I/O device.

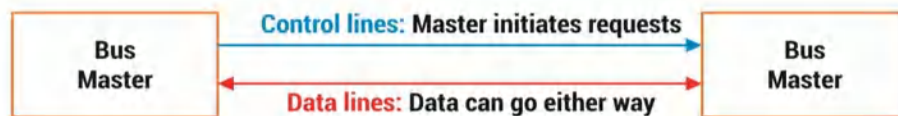


Fig.2.5 Master and Slave bus characteristics

### FSB Buses

It is known as the front side bus which is a computers communication bus that is used in computers that supports Intel chips. It serves as an external interface from the CPU to the other parts of the computer system, as opposed backward from where the back side bus connects the cache. Fig. 2.6 shows architecture of FSB bus system.

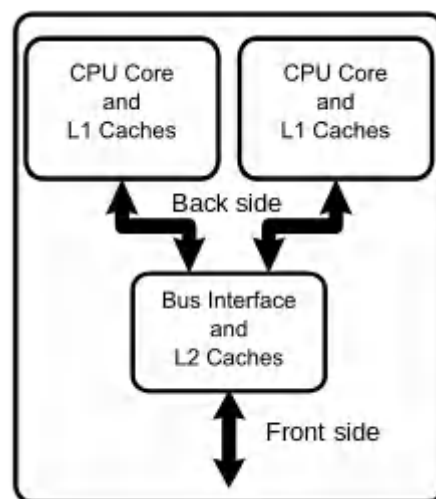


Fig. 2.6 Architecture of FSB bus

It is frequently used in Computer motherboards, only just with data and address buses as used in fixed systems and in small computers. Such design shows a performance improvement over the single system bus designs used earlier which are still called as system bus.

### **PCI Buses**

PCI is also known as Peripheral Component Interconnect which is a computer bus which was created by Intel in 1993. This bus can transfer 32 or 64 bits of information or data at a single time. Such a type of bus can run at a speed of 33 Mhz.

It is available in 32-bit and 64-bit versions and was commonly used to attach computer hardware. Due to advancement in Computer technology, there are many revisions which first came in the year 1993 with version 2.0 and with version 2.1 in 1995 as an expansion slot to the ISA bus. Fig. 2.7 shows an example of PCI slots on a motherboard.

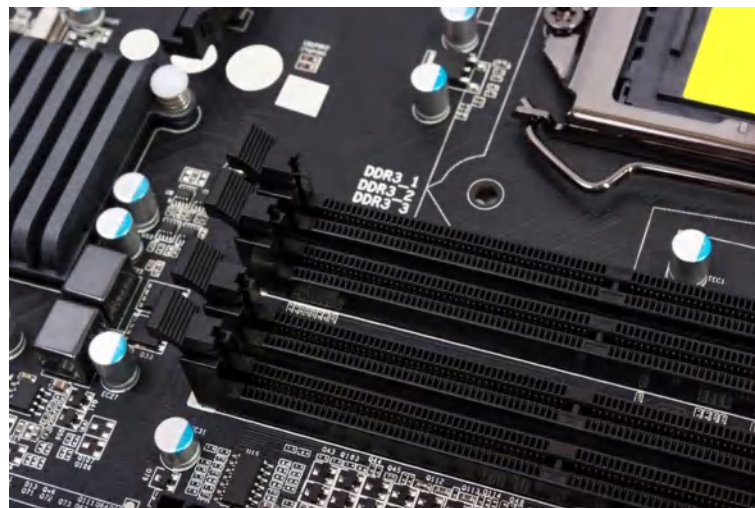


**Fig. 2.7 PCI slots on a motherboard**

### **USB Buses**

USB is referred to as Universal Serial Bus which is a computer bus introduced in 1995 by Intel, Compaq and Microsoft. It is used for connecting a keyboard and mouse, and other USB devices with the computer. Normally, a USB bus has a connector with four wires out of which 2 wires are used for supplying power to USB devices. Fig. 2.8 shows an arrangement of USB bus.





**Fig. 2.8:** It is an external bus standard that supports data transfer rates of 12 Mbps and is good for supporting up to 127 devices.

### Check your progress 2

1. Bus is a \_\_\_\_\_ joining internal components of computer hardware.
  - a. serial network
  - b. parallel network
  - c. serial or parallel network
  - d. complex network
2. Which bus is known as memory bus?
  - a. Address bus
  - b. Data bus
  - c. Control bus
  - d. Network bus

---

## 2.4 Let Us Sum Up

---

### In this unit, we have learned:

- ALU stands for Arithmetic and Logic Unit. It performs Addition, Subtraction, Multiplication and Division operations.

- Data paths are the internal registers or can be arithmetic and logic parts which carry the desired bus structures.
- Memory in computer architecture refers to the main or primary memory which is a collection of cells or locations.
- Control Unit is the main part of the computer architecture which controls the progress of other parts.

---

## 2.5 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-a), (2-d), (3-d)

**Check your progress 2**

**Answers:** (1-c), (2-a)

---

## 2.6 Glossary

---

1. **Buses** - It is the physical interconnection of components with the help of wires.
2. **Data path** - These are internal registers, such as ALU and its connecting buses.
3. **Memory** - It is a collection of cells or locations.
4. **Bus bar** - It is a copper, aluminum rod like shape that conduct electricity within the switchboards.

---

## 2.7 Assignment

---

Prepare a report and write the various features about the logical architecture of Computer Buses.

---

## 2.8 Activities

---

Design an internal architecture model of a computer bus and discuss.

---

## 2.9 Case Study

---

Visit a Computer Hardware Store and ask about the arrangement of computer buses in the PC.

---

## 2.10 Further Readings

---

1. Linda Null; Julia Lobur. Essentials of Computer Organization and Architecture
2. Walter A. Elmore. Protective Relaying Theory and Applications.

---

## UNIT 3: MOTHERBOARD

---

### Unit Structure

#### 3.0 Learning Objectives

#### 3.1 Introduction

#### 3.2 Various chips on mother board

#### 3.3 RAM

#### 3.4 L2 Cache

#### 3.5 BIOS & Chipset

#### 3.6 Memory Controller Hub (North Bridge)

#### 3.7 I/O Controller Hub (South Bridge)

#### 3.8 Let Us Sum Up

#### 3.9 Answers for Check Your Progress

#### 3.10 Glossary

#### 3.11 Assignment

#### 3.12 Activities

#### 3.13 Case Study

#### 3.14 Further Readings

---

### 3.0 Learning Objectives

---

**After learning this unit, you will be able to understand:**

- About execution of programs.
- The Fetch Decode Execution Cycle.
- About buffering.

---

## 3.1 Introduction

---

Motherboard is the main board of the computer. It is an assembly where all components get connected. It has lots of chips, connectors and other devices mounted on it. Inside the computer, data is regularly exchanged among the various devices as shown in Fig. 3.1.

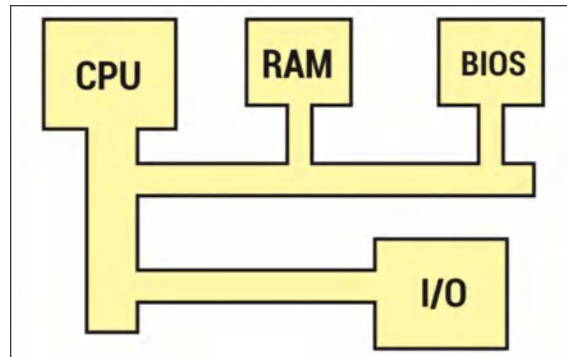


Fig. 3.1: Various devices

In the Fig. it is seen that the data exchange takes place on the motherboard itself, where all the components are connected to each other. The motherboard is made of a PCB sheet where all sockets, ICs, transistors put in. On the motherboard, there are electrical connectors which get connected to the respective components.

---

## 3.2 Various chips on mother board

---

The motherboard is made of a PCB plastic sheet where all sockets, plastic brackets and screws are present. Every component that is available in the computer gets connected through the motherboard with the help of connectors. Fig. 3.2 shows the motherboard.



Fig. 3.2: Motherboard Chipset

## Chips

It is a place where all active devices get assembled on the motherboard. Chips are basically small electronic circuits which are packed with transistors. There are different chips which have varied functions such as:

- **ROM chips:** It stores the BIOS and other programs.
- **CMOS:** It is a storage chip, which carries users that are used by setup program.

Chipset normally comprises of controllers, which includes a number of essential functions.

## Sockets

There are sockets available on the motherboard. Sockets are holders that are fixed on the motherboard with the help of soldering. Fig. 3.3 shows the socket on the motherboard.

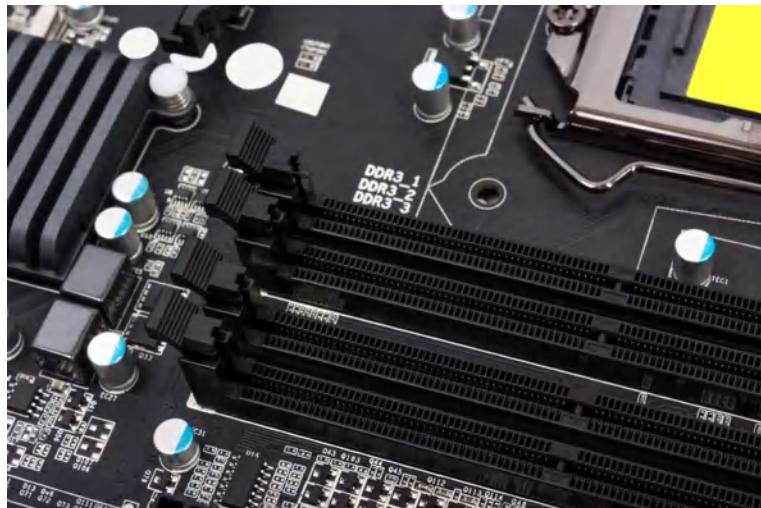


**Fig. 3.3 Motherboard socket**

These sockets serve as a place where certain components are fixed or mounted directly.

Sockets are the place to mount:

- Processor or CPU
- RAM
- Expansion cards
- Adapters (PCI, AGP and AMR)

The basic idea of a socket is to install the component directly on the motherboard without the use of fixing tools. The component is pushed carefully and firmly into the socket to get it fixed.

### **Plugs, connectors and ports**

Apart from chips and sockets, motherboard carries a number of inputs and outputs, so that various components can be connected. There are certain ports which have a connector at the back to get the components connected. These ports are for:

- Keyboard and mouse.
- Serial ports, the parallel port, and USB ports.
- Speakers/microphone.

These ports or sockets are soldered on the motherboard, so that it is easy for external components such as keyboard, mouse, printer, speakers, etc., to get connected easily as shown in Fig. 3.4.



**Fig. 3.4 Socket/ports on Motherboard**

Apart from these sockets, there are connectors and ports on the motherboard where other components are connected.

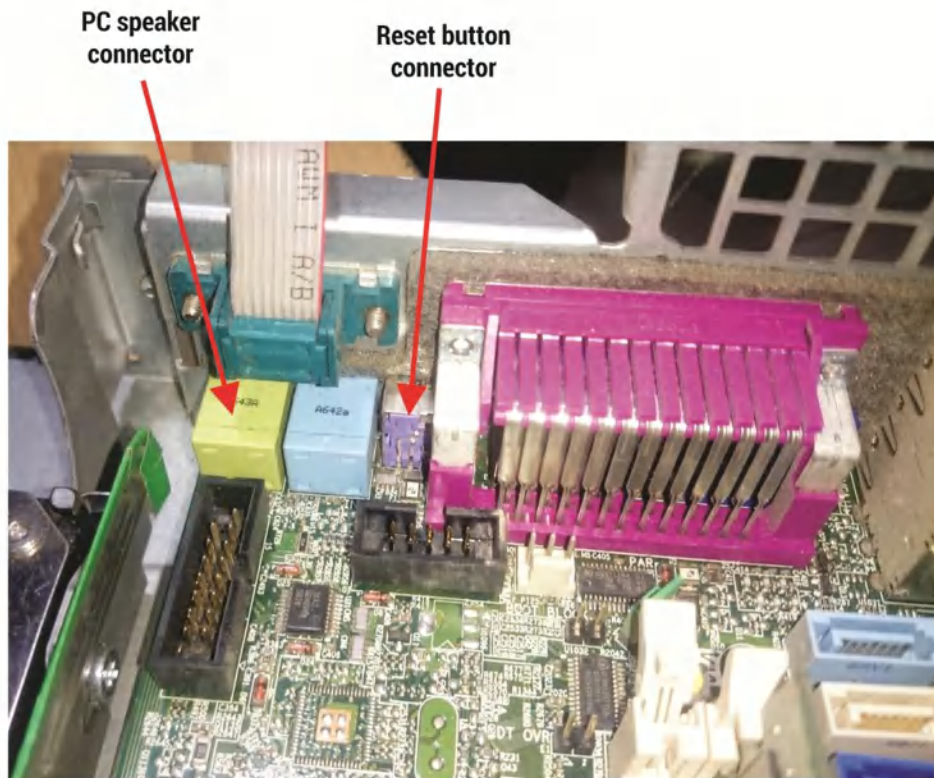


**Fig. 3.5 Motherboard connectors**

It includes:

- Power Supply Connector: Supply of power to motherboard and other components.
- Other connectors: These are the connectors which are used to connect a hard drive, floppy drive, CD ROM drive etc.

Apart from plugs and connectors, there are jumpers located on the motherboard, which are used to adjust the voltage and operating speed. A number of pins used to connect the reset button, LED for hard disk activity, built-in speaker, etc.



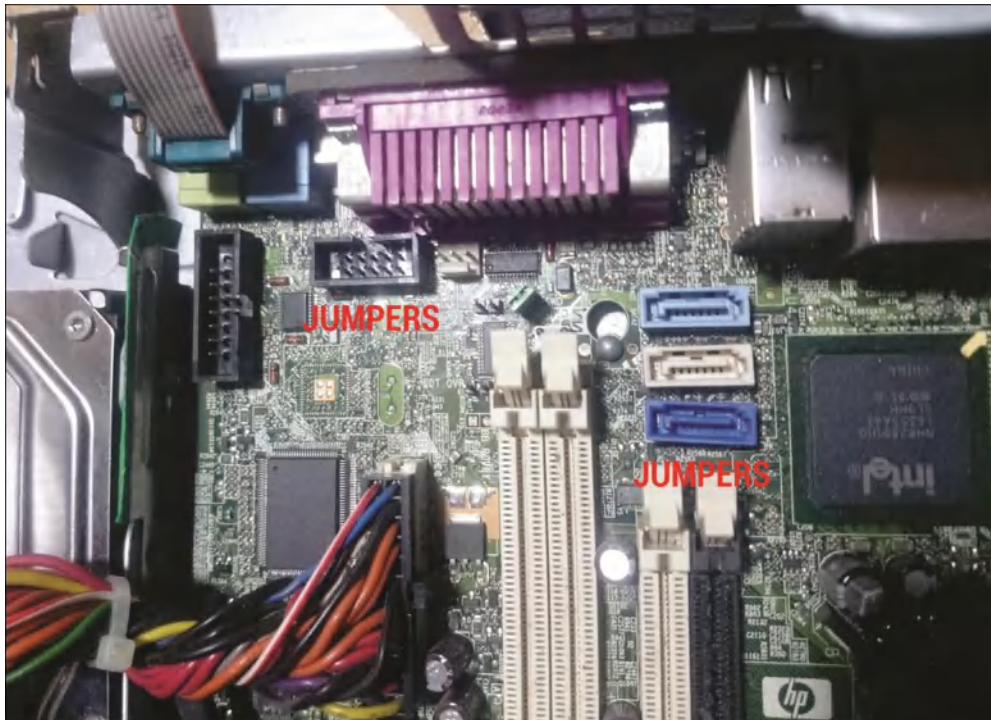


**Fig. 3.6 Jumpers on motherboard**

### Check your progress 1

1. \_\_\_\_\_ is not directly connected on Motherboard.
 

|         |              |
|---------|--------------|
| a. RAM  | c. CPU       |
| b. BIOS | d. Hard Disk |
2. Motherboard is made of:
 

|                    |                  |
|--------------------|------------------|
| a. Aluminium sheet | c. PCB sheet     |
| b. Metal sheet     | d. Plastic sheet |
3. Chips are small electronic circuits made of:
 

|                |                   |
|----------------|-------------------|
| a. insulators  | c. conductors     |
| b. transistors | d. semiconductors |
4. Which of the following component is not mounted directly on the Socket?
 

|              |              |
|--------------|--------------|
| a. Processor | c. Adapters  |
| b. RAM       | d. Hard Disk |

---

### 3.3 RAM

---

RAM also called as Random Access Memory as shown in Fig. 3.7 is a type of memory that temporarily stores the instructions that the computer is running, and the data it is processing.



**Fig. 3.7 RAM**

RAM is faster in speed and can easily read and write. The information stored in the memory gets lost when you switch off the computer or the power goes off. RAM increases the speed of your computer. When a computer is in use, its RAM will contain:

- Operating system software
- Software currently running
- Data

#### **Check your progress 2**

1. What does RAM stand for?
  - a. Remote Authorization Mechanism
  - b. Readily Accessed Mailer
  - c. Random Access Memory
  - d. Random Authorization Mechanism

---

### 3.4 L2 Cache

---

It is the CPU cache memory which is a Level 2 cache known as L2 caches. The L2 cache feeds the Level 1 cache and hence L2 cache is known as cache of

the L1 cache. This type of CPU cache is built in the same way as L1 caches, into the CPU but often it is located in another chip or in a Multichip Package Module which is a fully different chip.

With its exceptions, L2 caches are sometimes considered as SRAM or static RAM but the memory of the computer is considered as DRAM or Dynamic Ram. A L2 cache is bigger than a L1 cache, hence the data can be easily located in L2 cache which makes the data accessing faster. Now if the data is not present in a L1 cache, then it is looked for into a L2 cache, which is bigger than L1 cache.

L2 cache memory is located on a separate chip which can be seen more quickly than the main memory of the computer. It is visualised that a famous L2 cache memory is of the size 1,024 kilobytes which is equal to 1 megabyte.

### Check your progress 3

1. L2 cache is a \_\_\_\_\_ cache memory.
  - a. RAM
  - b. BIOS
  - c. CPU
  - d. Hard Disk
2. L2 cache memory is \_\_\_\_\_ than L1 cache memory.
  - a. smaller
  - b. bigger
  - c. equal
  - d. different

---

## 3.5 BIOS and Chipset

---

### Bios

It is seen that you require a special kind of program to enable the CPU to communicate with other devices. Such programs are commonly called as Basic Input/Output Service (BIOS). These programs are stored in the ROM. Fig. 3.7 shows the circuitry of BIOS on motherboard.



**Fig. 3.8 BIOS on motherboard**

In this:

- Each program is called a service.
- Programs are stored on ROM chips.
- Programs stored on erasable media are called software.

Fig. 3.8 shows the ROM chip

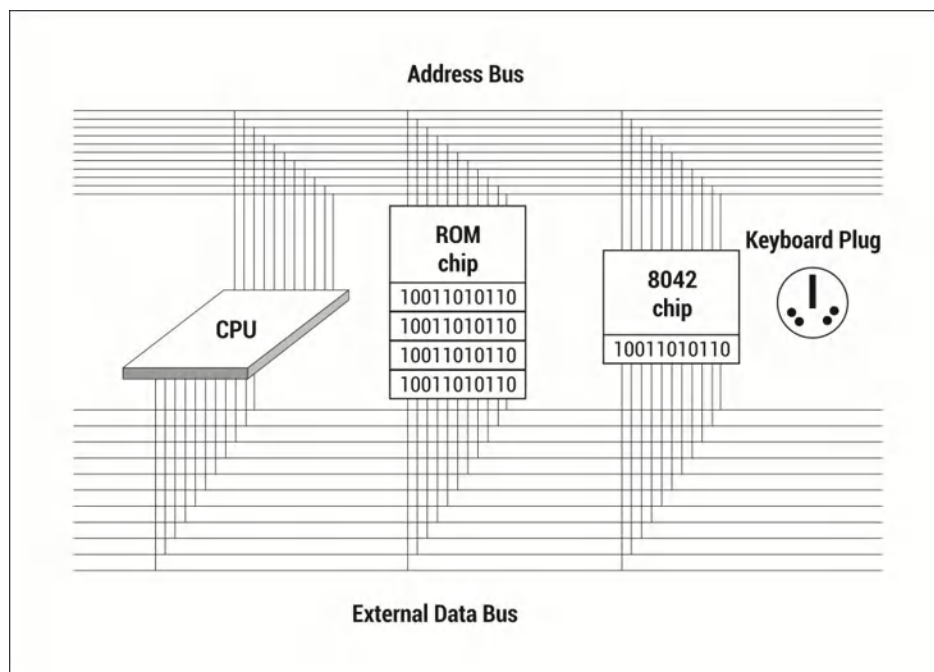


**Fig. 3.9 ROM chip**

The chipset is the mixture which normally pastes the CPU or microprocessor with the motherboard and with rest of the computer. Motherboard chipsets are manufactured by Intel, VIA, Nvidia, SiS and by many other companies. There are different chipset designed for different processors. It is the duty of the manufacturer to match the respective chipset with the required processor so that they can coordinate in speed, accuracy, performance etc. Basically a chipset is an integrated part of the motherboard; hence it cannot be removed or upgraded. There are two different microchips that are present in a chipset:

- Northbridge
- Southbridge

The Fig. 3.9 shows the arrangement of the Northbridge and Southbridge microchip.

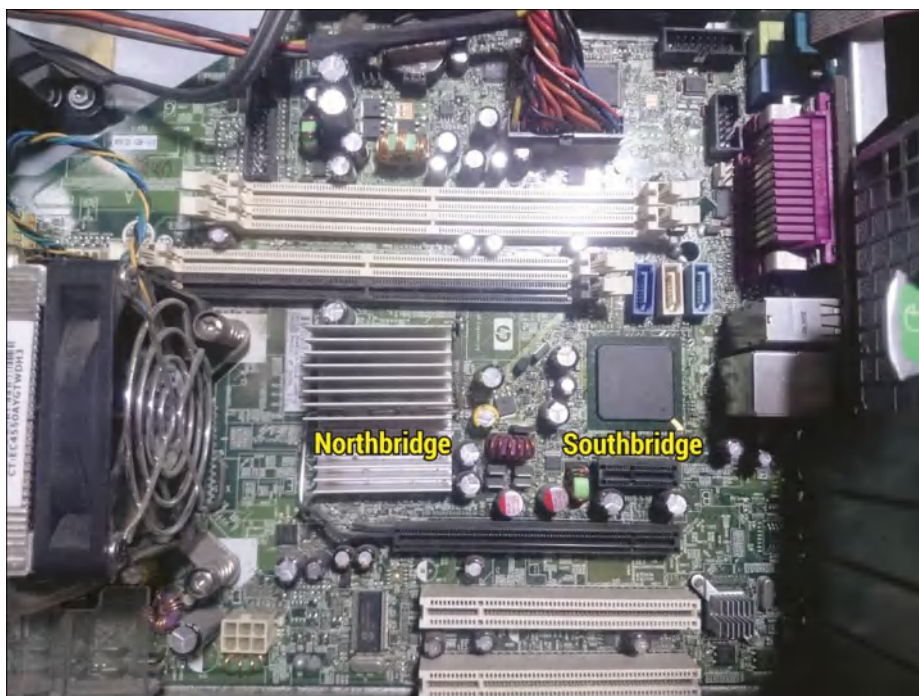


**Fig. 3.10 Northbridge and Southbridge microchip**

All of the various components of the computer communicate with the CPU through the chipset.

### Check your progress 4

1. BIOS is
  - a. Basic Input/output Service
  - b. Base Input/output Service
  - c. Before Input/output Service
  - d. Basic Input/output System

## 3.6 Memory Controller Hub (North Bridge)

A Northbridge is also known as a Host Bridge, it is a microchip which is present on the computer motherboards which is connected directly to the CPU. A Northbridge is required in need of highest performance. The Southbridge which is known for I/O controller hub is usually grouped with the Northbridge. All communications that exist between a CPU and the respective parts of a motherboard are handled by the Northbridge and Southbridge which provide as the core logic chipset of the motherboard.

Northbridge was known to be the external memory controller hub on former motherboards when they were fitted alongside integrated VGA memory controller hubs. The qualities of this bridge have now been integrated into the CPU chip starting with memory and graphics controllers.

On the motherboard, a Northbridge connects directly to the CPU or processor from the front side bus (FSB). The CPU is made to access the memory quickly when a memory controller is established on the Northbridge. The Northbridge also connects to AGP or PCI Express bus and also to the memory itself. It is analysed that a North Bridge handles data for a graphics port as to whether the AGP or PCI expresses along with the main memory including FSB.

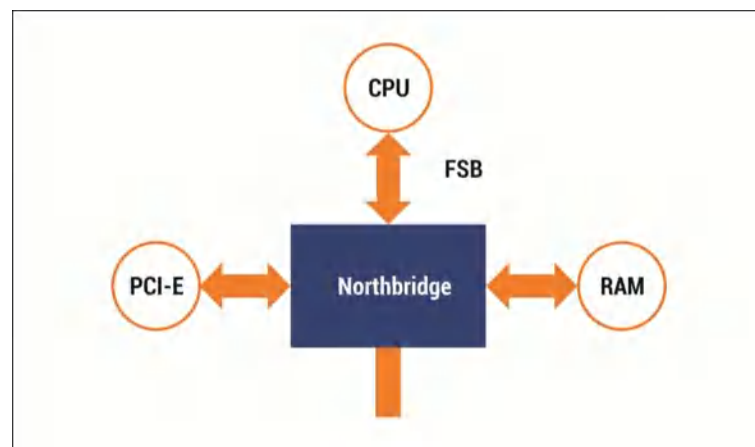


Fig. 3.11 Northbridge



**Check your progress 5**

1. Northbridge also known as \_\_\_\_\_ Bridge
  - a. RAM
  - b. BIOS
  - c. Ghost
  - d. Host

---

### 3.7 I/O Controller Hubs (South Bridge)

---

Another microchip that is available in the core logic chipset on a computer motherboard is the Southbridge which is responsible for enacting slower functions on the motherboard chipset. . It is named as I/O Controller Hub for Intel (ICH), while Fusion Controller Hub (FCH) for AMD. Southbridge can be distinguished from Northbridge as it is not directly linked to the CPU. It is seen that a Southbridge chipset can handle all computer I/O functions like USB, audio, serial, system BIOS, ISA bus, interrupt controller and IDE channels.

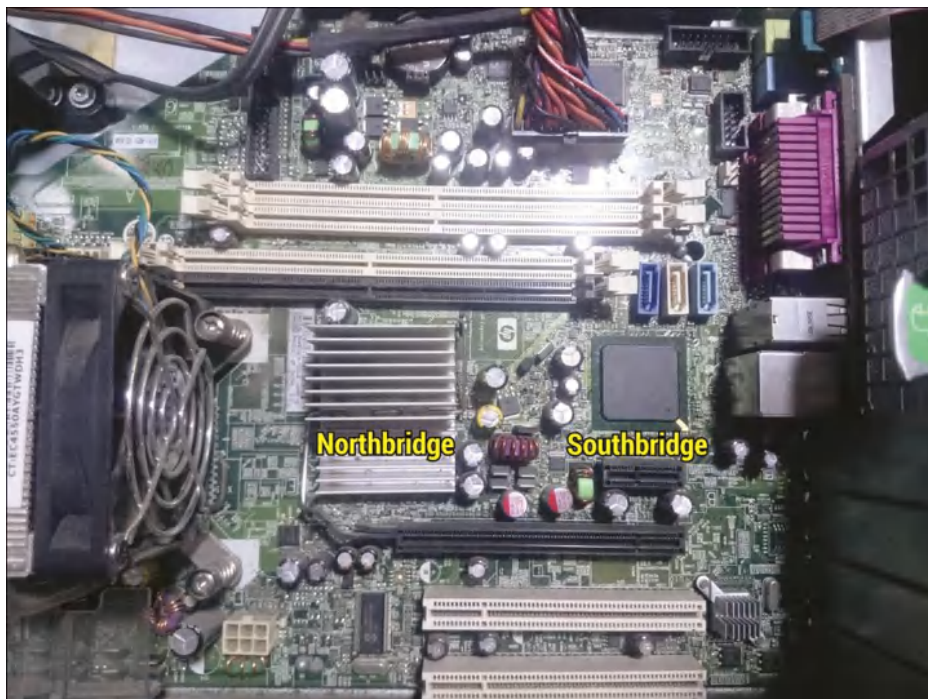


**Fig. 3.12 Arrangement of Southbridge**

Fig. 3.11 shows an arrangement of a Southbridge. This bridge will handle data from the PCI x1 slot having integrated components as Audio and/or on board graphics. The Southbridge is rather sluggish since the information from the CPU

first progresses toward the Northbridge and then reaches the Southbridge. Here the Southbridge is connected to the PCI bus, the USB ports and the IDE or SATA hard disk connections through busses.

### Check your progress 6

1. Southbridge is known for \_\_\_\_\_ controller hub.
  - a. input
  - b. output
  - c. input/output
  - d. memory
2. Southbridge chipset cannot handle functions like.
  - a. USB
  - b. audio
  - c. system BIOS
  - d. speed of CPU

---

## 3.8 Let Us Sum Up

---

**In this unit, we have learned:**

- Motherboard is the main board of any computer system that carries and supports all hardware attached to it. All hardware components are connected on motherboard with the help of wires.
- It is seen that motherboard is a chipset where different chips are soldered on it. Every chip has its own function. There are certain sockets, plugs, ports available on motherboard which connects several other components.
- Northbridge and Southbridge are two microchips of a chipset available on the motherboard.

---

## 3.9 Answers for Check Your Progress

---

### Check your progress 1

**Answers:** (1-d), (2-c), (3-b), (4-d)



**Check your progress 2**

**Answers:** (1-c)

**Check your progress 3**

**Answers:** (1-c), (2-b)

**Check your progress 4**

**Answers:** (1-d)

**Check your progress 5**

**Answers:** (1-d)

**Check your progress 6**

**Answers:** (1-c), (2-d)

---

## 3.10 Glossary

---

1. **Motherboard** - Principal printed circuit board assembly in a computer system.
2. **Chipset** - It is a combination of two or more integrated circuits that controls interface among system processor, RAM, I/O devices and adapter cards.
3. **Port** - port is an interface connector which is used to connect numerous types of devices.
4. **USB** - Universal Serial Bus - a medium speed interface typically used for mouse, keyboards, scanners, display panels (control features, not data), speakers (control features, not sound), scanners, and some digital cameras.
5. **Connector** - a series of two or more metal pins assembled on the motherboard or PCB to attach different cables on the motherboard.
6. **Jumper** - it is a series of two pin groups used to set CPU performance.

---

### 3.11 Assignment

---

Give reasons why L1 cache memory is faster than L2 cache memory.

---

### 3.12 Activities

---

Write the series of steps, how the motherboard will help other components to work after all is connected to it.

---

### 3.13 Case Study

---

Locate what components are present on the motherboard chipset.

---

### 3.14 Further Readings

---

1. Computer Organisation by Goldberg
2. Computer Architecture by Sarah Harris
3. Computer Mainframes by Almasi and Gottlieb

---

## **UNIT 4: HARDWARE COMPONENTS OF SMARTPHONE**

---

### **Unit Structure**

- 4.0 Learning Objectives**
- 4.1 Introduction**
- 4.2 Applications Processor**
- 4.3 SIM card**
- 4.4 Wireless connectivity**
- 4.5 Audio subsystem**
- 4.6 Memory card and flash memory**
- 4.7 Power Management Unit**
- 4.8 Camera and its interface**
- 4.9 Touch screen & its interface**
- 4.10 Let Us Sum Up**
- 4.11 Answers for Check Your Progress**
- 4.12 Glossary**
- 4.13 Assignment**
- 4.14 Activities**
- 4.15 Case Study**
- 4.16 Further Readings**

---

### **4.0 Learning Objectives**

---

**After learning this unit, you will be able to understand:**

- The hardware of Smart phone.
- The basic functions of various hardware of a Smartphone.
- The various features of a Smart phone.

---

## 4.1 Introduction

---

A Smartphone is a type of mobile phone that can work similarly as a computer. It has a touch pad, touch screen, internet connectivity, internet browsing, music, watching movies, downloading applications and several other features. Since it carries such different advance features, hence it is known as a Smartphone. Smartphones are manufactured by many companies and are commonly available with different models, sizes, features and characteristics.

It is seen that a Smartphone carries a number of hardware components such as:

- applications processor
- baseband chip
- communications chips (Bluetooth, GPS functions, Wi-Fi)
- graphics processor
- memory for operating system
- data user applications

### **Features of Smartphone:**

- Contains recognized mobile operating system, as Nokia's Symbian, Google's Android, Apple's iOS or the BlackBerry OS.
- Internet connectivity.
- Mobile browser.
- Good synchronization.
- Embedded memory.
- Hardware and/or software-based QWERTY keyboard.
- Wireless synchronization with compatible devices as laptop or desktops.
- Ability to download and run applications.
- Support for third-party applications.
- Ability to run multiple applications simultaneously.
- Touchscreen.
- Wi-Fi.

## 4.2 Applications Processor

A smartphone processor is similar to that of a computers processor. Since it performs similar functions as computer and has the same memory and capacity, so in order to handle all this, companies now are stressing on to develop good mobile processors that can support all necessary applications in no time. Fig. 4.1 shows a diagrammatic view of application processor.



**Fig. 4.1 Diagrammatic view of application processor**

An application processor is a self-contained operating environment which delivers the required system capabilities which is needed to give support to a devices application such as memory management, graphics processing and multimedia decoding. It is studied that an application processor is an advanced and powerful device operating processor which can operate at frequencies of several hundred MHz, to few GHz. Some of the high performance application processors are packaged with Package on Package (PoP) technology, which saves the motherboard space.

### **Application processors have certain features:**

- With high processing and packaging technology, these processors are much advanced and speedy.
- They are built in small sizes and require less board space.
- Because of their small size, they are used in digital still/video cameras and mobile handsets.

- These are high speed computing devices having 1 computing core per device.
- These are multi core application processors that use extensive on-chip interconnection among the cores.
- They have high-density I/O quality.
- They have excellent thermal stability.
- They are operationally reliable.

### Check your progress 1

1. Smartphone is smart because it has
  - a. Touch screen
  - b. Internet Connectivity
  - c. Downloading applications
  - d. all of these
2. Smartphone differs ordinary phone in
  - a. features
  - b. size
  - c. shape
  - d. model
3. Smartphone processor has
  - a. big size chip
  - b. cannot fit in size
  - c. faster in working
  - d. gets heated fast

---

## 4.3 SIM card

---

Every mobile phone works with the help of a SIM card. A SIM card is a subscriber identity module that stores data for GSM/CDMA mobile users. The SIM card stores the user information such as:

- user identity
- network authorization data
- personal security keys
- contact lists
- stored text messages

Apart from this, there are security features also which are stored in the SIM card such as authentication and encryption to protect data and prevent eavesdropping. Also, a smartcard with user identity module application is normally called a SIMCARD. If you have a SIM card, then you can switch it easily from one phone set to another. In this the stored data on the SIM card can be viewed by connecting it to other Smartphone's.

### **Types of SIM card**

There are three types of SIM cards:

- Standard
- Micro
- Nano

Every Smartphone can work with any SIM card provided the SIM card should fit in the slot designed for it.

#### **Standard SIM Card**



**Fig. 4.2 Standard SIM Card**

Fig. 4.2 shows Standard SIM Card which is used in earlier mobile phones and devices. Some of the devices that use such SIMs are:

- Apple- iPhone 3GS
- Samsung - Galaxy SII, Note
- Nokia - 300, 301
- HTC - Desire HD
- EE Mobile broadband/4G WiFi - Buzzard, Osprey

### Micro SIM Card



**Fig. 4.3 Micro SIM Card**

Fig. 4.3 shows a Micro SIM Card which is normally smaller in size than the Standard SIM Card. It is mainly used in Smartphone's. Some of the devices that use such SIMs are

- Apple - iPhone 4/4S
- Apple iPad Retina Display
- Sony - Xperia S, T, X, Z
- HTC - One, One X, One SV, 8X
- Nokia - N9, Lumia 720/820
- Samsung - Galaxy Express, SIII, S4, Note II

### Nano SIM Card



**Fig. 4.4 Nano SIM Card**

Fig. 4.4 shows a Nano SIM Card which is used in the latest Mobile phones and Smartphone's which is smaller than the Micro SIM card. The examples of Nano SIM devices are:



- Apple - iPhone 5, 5c, 5s and iPad Mini
- HTC - One (M8) HTC Desire 610
- Multi and Combi SIMs

### Function of SIM card

- **Subscriber Identification:** The SIM card has IMSI programmed which is used to find a subscriber. Every IMSI is matched with a specific mobile number and temporary on HLR which allows subscriber to be identified.
- **Subscriber Authentication:** The SIM card authenticates the user with the help of authentication algorithm that is there on the SIM card. Each subscriber provides a unique response based on IMSI, Ki and RAND. After matching these with values calculated on the network, it logs a legal subscriber on the network to use the specific services.
- **Storage:** SIM card is used to store phone numbers and SMS.
- **Applications:** The SIM card tools allow creating applications on the SIM card to provide the basic information as required and for other applications such as m-commerce, chatting, cell broadcast, phonebook backup, location based services etc.

### Check your progress 2

1. SIM card is known as:

|                                 |                               |
|---------------------------------|-------------------------------|
| a. subscriber identified module | c. subscriber identity module |
| b. subscriber identity modulus  | d. subscriber item module     |
2. Which is the small sized SIM card?

|             |                  |
|-------------|------------------|
| a. Standard | c. Nano          |
| b. Micro    | d. None of these |
3. In Apple-iPhone 4/4S, which SIM card is used?

|             |                  |
|-------------|------------------|
| a. Standard | c. Nano          |
| b. Micro    | d. None of these |

---

## 4.4 Wireless Connectivity

---

Nowadays, many mobile phones including Smartphones are enabled to exchange ActiveSync that can connect to a wireless LAN. It is studied that a wireless LAN connection is faster among all networks and carries excellent coverage in areas where other networks are not proper. Wireless access is now available in almost every Smartphone. The main advantage of such connectivity is:

- Wider coverage
- Economical
- Faster Speed
- Fast browsing
- Excellent download

### Check your progress 3

1. Smartphones nowadays uses:
  - a. WAN connectivity
  - b. LAN connectivity
  - c. MAN connectivity
  - d. all

## 4.5 Audio subsystem

Nowadays there are mobile phones which have audio input with speakers.

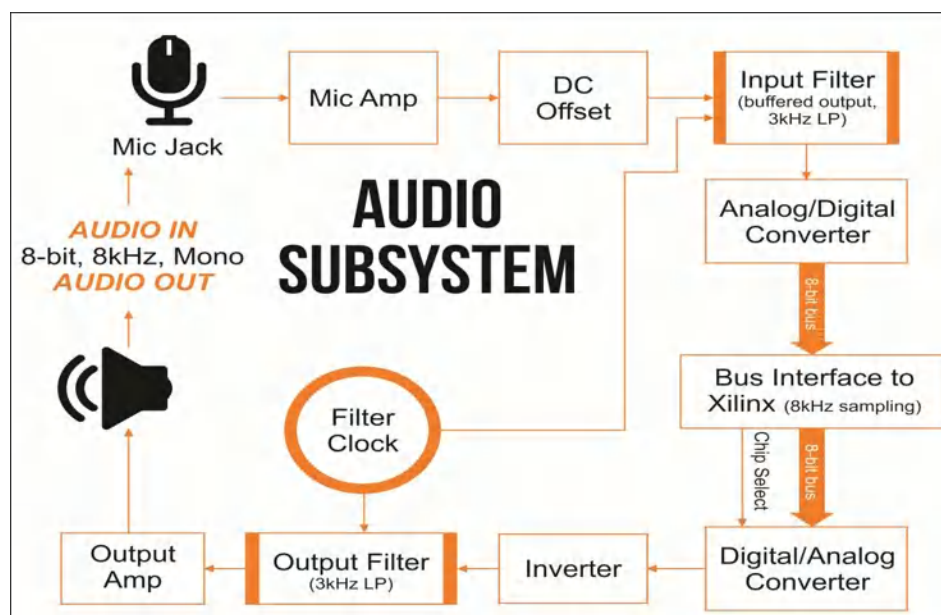


Fig. 4.5 Audio subsystem

Fig. 4.5 shows the basic structure of an audio subsystem in mobile phones. The audio subsystem delivers telephone quality audio in digital form and further playback that audio through the speaker. It is studied that the audio quality of telephone can be of 8-bit which is somewhat around 3 kHz mono speeches. As explained by the Nyquist's theorem, the sampling rate should be twice the signal frequency in order to correctly reproduce a waveform. Based on the Nyquist's theorem, the audio was best sampled with play back at 8 kHz. Further, 8 bit allows sound to be of 256 possible values having an input voltage range of 0 to 5V having step function of 0.02V / bit. It is seen that an audio subsystem comprises of two distinct systems:

- audio input (analog-digital conversion).
- audio output (digital-analog conversion).

We see that at the time of recording, audio is input by the means of a microphone which is amplified, filtered, digitized and is sent back. At the playback time, audio is converted back to analog which is filtered, amplified and gives the output by speaker.

### Check your progress 4

1. Sampling of audio is explained by\_\_\_\_\_theorem.
  - a. Nyquist's
  - b. Pythagorean
  - c. Rutherford
  - d. none of the above

---

## 4.6 Memory card and flash memory

---

### Memory card

Memory card is also known as a flash card which is an electronic flash memory data storage device which is typically used for keeping digital information. These are mostly used in several electronic devices such as:

- digital cameras
- mobile phones
- laptop computers
- tablets
- MP3 players
- video game consoles

Out of the above listed memory card, most of these can be very small, re-recordable and can obtain data without using power.

### Flash Memory

Flash memory is an electronic non-volatile memory which can be erased and reprogrammed in units of memory known as blocks. It was invented by Toshiba in 1984 which was developed from EEPROM. There are two types of flash memory. Flash memory is so called because the microchip is arranged in such manner that a section of memory cells gets erased in single action or flash.

Similarly like RAM, flash memory will also continue to store information without the need of a power source. With this quality, the flash memory is ideal for use in:

- Smart Phones
- Digital Cameras

- Tablets
- Laptops
- MP3 Players

Similarly like ROM, a flash memory can be written off, which updates the operating system \and can be used as application software. Apart from these, the other advantages of flash memory are:

- fast access time
- compact size
- fixed

The applications of both types of flash memory include:

- personal computers
- PDAs
- digital audio players
- digital cameras
- mobile phones
- synthesizers
- video games
- scientific instrumentation
- industrial robotics
- medical electronics

### **Check your progress 5**

1. Flash memory is not used in:
  - a. Smart Phones
  - b. Digital Cameras
  - c. Tablets
  - d. Desktops

---

## 4.7 Power Management Unit

---

In a smartphone the power management integrated circuits are used to handle power requirements and support voltage scaling and correct power transfer sequencing to other smartphone units or devices. Power management in a smartphone is the main component of any device as it contains a power supply, battery, or power cord, and optimizes power usage. Such integrated power system is used in electronic systems like:

- cellphones
- TVs
- computers
- smart meters
- grids

The power management circuit used in Smartphone's are herein known as Smartphone power management ICs. These integrated circuits manage power requirements of the host system. It carries a PMIC which covers battery operated devices as mobile phones and portable media players. In a Smartphone, these power management circuits perform the following functions:

- Battery charging
- Power source selection
- Voltage scaling
- Power sequencing
- supply power to related Smartphone components

It is observed that 4G Smartphone's use large battery for searching signals, which are currently limited in case of 3G signals as they consume more battery power by decoding high levels of data that is transmitted in the spectrum. Also, mobile users are now using their devices regularly for talk, text, email, web surfing, viewing high definition videos, GPS maps, video calls, play games and listen music.

Apart from this, the user demand for high display resolution and better touch functionality. Such features majorly consume high battery power, so there is a need for effective power management.

### Check your progress 6

1. In Smartphone, the role of power management integrated circuits:
  - a. handles power requirements
  - b. support voltage scaling
  - c. correct power transfer sequencing
  - d. all of these

## 4.8 Camera & its Interface

Nowadays every Smartphone is coming up with a camera. Camera phones are easy and simple as compared to digital cameras. Since digital standalone cameras have fixed-focus lenses and smaller sensors which limits their performance under bad lighting. A smartphone has a high definition camera which has resulted in lower sales of digital cameras. It is seen that many Smartphone's only have a single start for camera applications and a simple on-screen button to start. The main advantage of a camera in a cell phone is basically its cost and compactness. Smartphones with cameras can run mobile applications such as geo-tagging and image capturing. These high end phones use a touch screen to operate the camera to focus on the required object. However, touch screen, as general purpose control, has no quickness of different camera's particularly with buttons and dial(s).

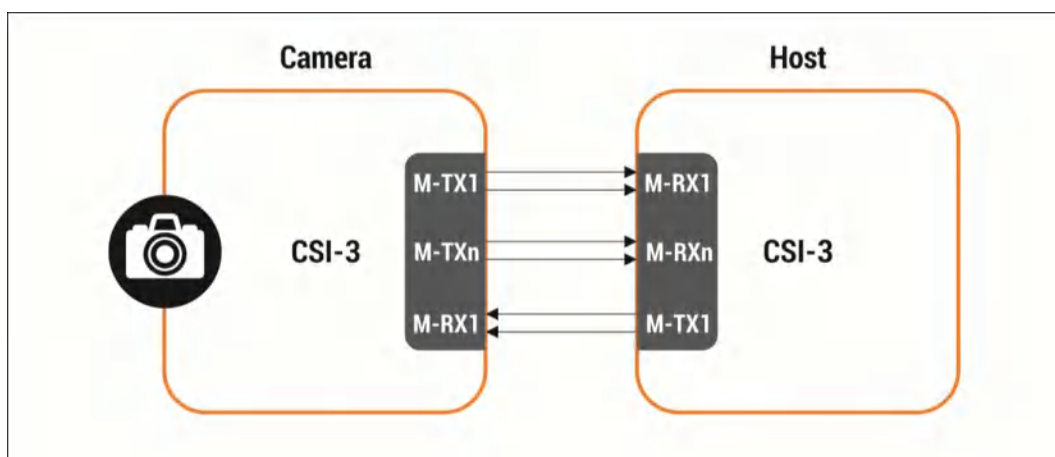


Fig. 4.6 Camera interface

Since long the mobile manufacturers have needed a standard interface so as to attach a camera subsystem to an application processor. With this a CSI-2 was developed as shown in Fig. 4.6 that will support a range of devices in mobile products. As a user continues to change, MIPI meets new generation camera serial interface which is required for many camera applications. The camera systems in mobile phones support both still and video cameras. In this, high resolution sensors give digital zoom, and high pixels that are required for still or moving images with much high frame rates.

### Check your progress 7

1. Old cameras will perform:
  - a. catching of still images
  - b. catching of videos
  - c. capturing moving images
  - d. capturing live auditions
2. The main feature of Digital Camera is:
  - a. catching of videos
  - b. capturing moving images
  - c. capturing live auditions
  - d. all of these

---

## 4.9 Touch screen & its interface

---

Practically speaking, a touch screen is an input device which is put on the top of any electronic visual display of an information processing system which enables the user to control the information processing system with the help of simple and easy multi touch gestures. Users operate the device by simply touching the screen with their fingers. Many touch screens contain specially coated gloves to work with while some can be operated by a special pen. While using a touch screen, the user can see the displayed information and can control the device by simply following instructions displayed on touch screen. The touch screen helps the user to interact directly with the displayed information, rather than using a mouse or any other intermediate device.

### Touch screen comprises of four layers:

1. Top layer is of polyester coated with a transparent metallic conductive coating on the bottom.



2. Second layer is of adhesive spacer.
3. Third layer has glass which is coated with a transparent metallic conductive coating on top.
4. Fourth layer is an adhesive layer which is on the backside of the glass used for mounting.

Touch screens are commonly available in devices such as:

- game consoles
- personal computers
- tablet computers
- smartphones

### Check your progress 8

1. Touch screens are not available in:
  - a. game consoles
  - b. personal computers
  - c. smartphone
  - d. zip drives

---

## 4.10 Let Us Sum Up

---

### In this unit we have learned:

- That there are certain hardware components that are present inside the smartphone's. Because of this variation in hardware, smartphone carries specific features.
- There are specific characteristics configuration and components present that makes a mobile phone to work smart. There are exceptional applications that can run with various components present in a smartphone which makes us to distinguish among 3G and 4G Smartphone's mechanism.
- There are variety of touch screens that carries special coated gloves to work as many carries special pen. While using a touch screen, the user can see the displayed information and can control the device by simply following instructions displayed on touch screen.

---

## 4.11 Answers for Check Your Progress

---

**Check your progress 1**

**Answers:** (1-d), (2-a), (3-c)

**Check your progress 2**

**Answers:** (1-c), (2-a), (3-b)

**Check your progress 3**

**Answers:** (1-b)

**Check your progress 4**

**Answers:** (1-a)

**Check your progress 5**

**Answers:** (1-c)

**Check your progress 6**

**Answers:** (1-d)

**Check your progress 7**

**Answers:** (1-a), (2-d)

**Check your progress 8**

**Answers:** (1-d)

---

## 4.12 Glossary

---

1. **Memory card** - It is the storage memory of a smartphone which stores data upto 32GB.
2. **3G** - It is a type of wireless mobile technology which gives access to internet connection in smartphones.

3. **4G** - It is a high speed mobile network compatible with latest smartphones.
4. **GPS** - It is a navigation application available in smartphones.
5. **Smartphone** - It is a mobile communications device that uses an identifiable open OS.

---

### 4.13 Assignment

---

Study the different types of Smartphone available in the market.

---

### 4.14 Activities

---

Collect information on a 4G Smartphone from the market and discuss.

---

### 4.15 Case Study

---

Does the application processor of a smartphone bear the same speed as a processor found in a computer system?

---

### 4.16 Further Readings

---

1. Johnson, E.A. (1965). "Touch Display - A novel input/output device for computers".
2. Hooper, Steven (2013-11-11). "Design for Fingers and Thumbs Instead of Touch.
3. Building a Better Battery, NY Times, 2 February 2014, BRIAN X. CHEN.
4. Gye, Lisa (2007) Picture This: The Impact of Mobile Camera Phones on Personal Photographic Practices.

---

## Block Summary

---

In this block we have given knowledge related to Logical Architecture of Computer and Smartphones. Here the students will be trained with basic Computer motherboard working and detailed with various hardware related to it. The knowledge related to various bus architecture will make the students to create interest about learning computer hardware.

The block helped the students with the introduction of memory and cache memory. Features related to smartphone were explained with detailed description. The knowledge about motherboards and its related bus structure will made the student to enter into chip level repairing. So this unit stresses on the execution, decoding and carrying out of necessary applications that will help to work with computer and smartphone hardware.

---

## **Block Assignment**

---

### **Short Answer Questions**

1. What is a Computer Memory?
2. What is Buffering?
3. What are registers?
4. What do you mean by Program Counter?
5. What is the function of ALU?

### **Long Answer Questions**

1. What is the role of CPU in Computer architecture?
2. How can a Data be written on a Hard Drive?
3. How registers are different from Program Counters

Enrolment No.

1. How many hours did you need for studying the units?

| Unit No    | 1 | 2 | 3 | 4 |
|------------|---|---|---|---|
| Nos of Hrs |   |   |   |   |

2. Please give your reactions to the following items based on your reading of the block:

| Items                                   | Excellent                | Very Good                | Good                     | Poor                     | Give specific example if any |
|-----------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|------------------------------|
| Presentation Quality                    | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Language and Style                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Illustration used (Diagram, tables etc) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Conceptual Clarity                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Check your progress Quest               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |
| Feed back to CYP Question               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | _____<br>_____               |

3. Any Other Comments.

.....

.....

.....

.....

.....

.....

.....

.....



“

*Education is something  
which ought to be  
brought within  
the reach of every one.*

”

- Dr. B. R. Ambedkar



**Dr. Babasaheb Ambedkar Open University**

Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,  
Ahmedabad-382 481.

# **Post Graduate Diploma in Computer Application**

**Lab Manual**

**For**

## **Web Application Development**

**Course Code: PGDCA – 202**

**Dr. Babasaheb Ambedkar Open University  
Gujarat**



## Contents

| Sr. No. | Objectives                                                                   |
|---------|------------------------------------------------------------------------------|
| 1       | Introduction of Web Application Development                                  |
| 2       | Importance                                                                   |
| 3       | Objective of Lab Manual                                                      |
| 4       | Introduction of LAB                                                          |
| 5       | Guidelines related to Lab                                                    |
| 6       | Lab 1: To Install PHP web Application.                                       |
| 7       | Lab 2: To Create a php webpage and print “hello world”.                      |
| 8       | Lab 3: To create a php program to find odd or even number from given number. |
| 9       | Lab 4: To write a PHP program to swap two numbers.                           |
| 10      | Lab 5: Give PHP Example to calculate the area of the circle                  |
| 11      | Lab 6: To declare multiple variables in for loop.                            |
| 12      | Lab 7: To declare a user defined function.                                   |
| 13      | Lab 8: Example of Numeric Array.                                             |
| 14      | Lab 9: Give the example of multiple dimensional array.                       |
| 15      | Lab 10: Give the example of string function: substr():                       |
| 16      | Lab 11: Give the example of string function: strcmp()                        |
| 17      | Lab 12: Write a PHP program to create a database using MySQL.                |
| 18      | Lab 13: Write a program to use namespace                                     |
| 19      | Lab 14: Write a PHP program to create a table in MySQL.                      |
| 20      | Lab 15: Write a PHP program to insert record into a table using MySQL.       |
| 21      | Lab 16: Write a PHP program to drop table using MySQL.                       |

|    |                                                                               |
|----|-------------------------------------------------------------------------------|
| 22 | <b>Lab 17: Write a PHP program to select data and show into table format.</b> |
| 23 | <b>Lab 18: Update the data present in MYSQL database using web form</b>       |
| 24 | <b>Lab 19 : To create PHP session</b>                                         |
| 25 | <b>Lab 20 :Write a program to change session variable</b>                     |
| 26 | <b>Lab 21: Write a program to create ,modify and deleting cookies.</b>        |

## Introduction of Web Application Development

---

Web application or services are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. PHP is Hypertext Pre-processor a programming language which is applied by web developers in order to frame dynamic content which is easy to interact with databases. Many web services make use of Extensible Markup Language (XML) in order to define format of request and response messages. This feature is tagged structure which shows required flexibility for changing of information that exists among disparate components. In PHP, there exists strong set of efficient ways that will help in dealing with such arrays. Arrays are basically applied in order to store and organize data efficiently and quickly within no time. It serves as an important data type which is present in any programming language as it is easily described as ordered list of elements.

PHP carries various state management features which will overcome certain problems which are taken from stateless Web which can be cookie. It is noted that cookie serves as piece of information which is send to server which further sends it to Web browser. In case, a web browser enters to subsequent page within similar domain, it will show server access with similar cookie which is quiet easy to implement. It is found that cookies are restricted to how information gets stored with lots of security problems. Apart from cookies, another state management technique in PHP is session.

## Importance

---

- ❖ Web applications are built on client/server architecture where business logic is present in the application that works on web server and uses HTTP in order to communicate with clients over Internet.
- ❖ PHP is an important coding technique which can be easily embedded directly in HTML.
- ❖ You will make to learn and understand about the basic of PHP in built functions with their features.
- ❖ PHP carries various state management features which will overcome certain problems which are taken from stateless Web which can be cookie.
- ❖ You will make to learn and understand about the basic of session cookies and their related techniques.

## Objective of Lab Manual

---

After completion of this lab, learners will:

- Learn and understand about basic of outputting data to web browser and its techniques.
- Knowledge related to PHP Supported Data types
- Learn basic of Identifiers, Variables, Constants and Expressions
- Learn the concept of Control Structures.
- Learn basic of Arrays with their features associated with storing and organizing of data with required efficiency.
- Learn basic of database connectivity with MySQL.
- Learn features of selecting data.
- Learn to create, modify and delete cookies.

## Introduction of LAB

---

There are 50 systems installed in computer Lab. Their configurations are as follows:

|                        |   |                                |
|------------------------|---|--------------------------------|
| Processor              | : | Pentium IV 2.4 GHz             |
| RAM                    | : | 512 MB                         |
| Hard Disk              | : | 40 GB                          |
| Mouse                  | : | Optical Mouse                  |
| Operating System       | : | Windows XP (or latest version) |
| Software               | : | PHP, MySql                     |
| Network Interface card | : | Present                        |

## Guidelines Related To Lab

---

Guidelines to learners:

- Equipment in the lab for the use of learner's community. Learners need to maintain a proper decorum in the computer lab.
- Learners must use the equipment with care. Any damage is caused is punishable.
- Learners are required to carry their observation / programs book with completed exercises while entering the lab.
- Learners are supposed to occupy the machines allotted to them and are not supposed to talk or make noise in the lab. The allocation is put up on the lab notice board.
- Lab can be used in lab time decided by lab-in charge.
- Lab records need to be submitted on or before date of submission.
- Learners are not supposed to use any USB or other devices.
- Use of computer network is encouraged.

**Practicals  
&  
Their solutions**

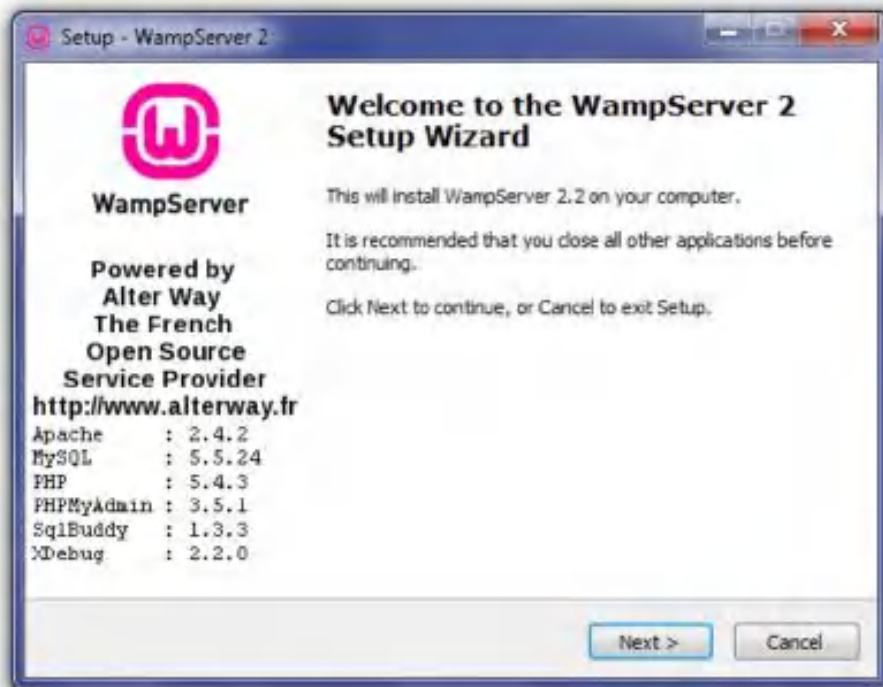


## Lab 1: To Install PHP web Application.

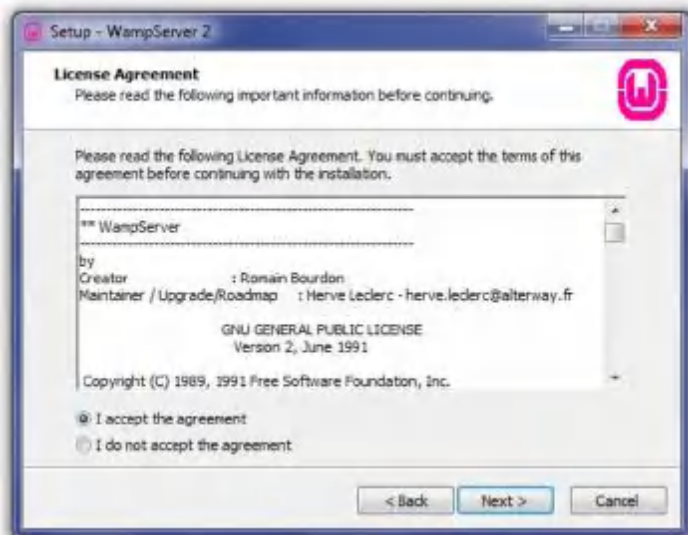
### Solution:

WAMP allows you to move between different versions of PHP if it is installed properly. There are certain steps to be followed while installing Wamp Server.

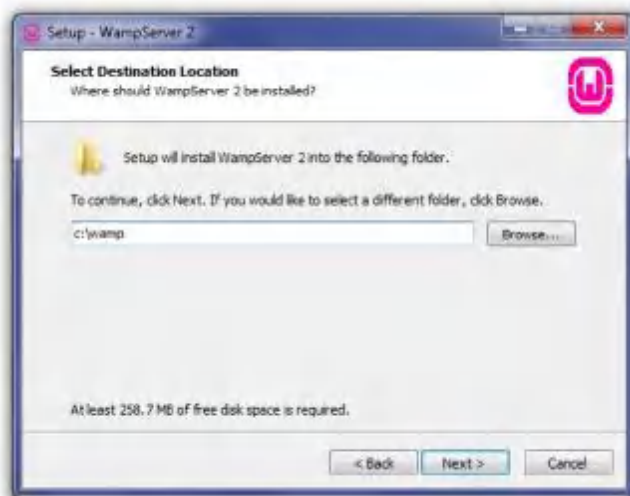
To start installing, you have to open the folder where you have saved your file and double click on installer file. After double clicking, a security warning window will open which will ask you whether to run this file or not. If you select on Run you will find that the installation process started and you will see Welcome To WampServer Setup Wizard screen on which you have choose next to forward with the installation.



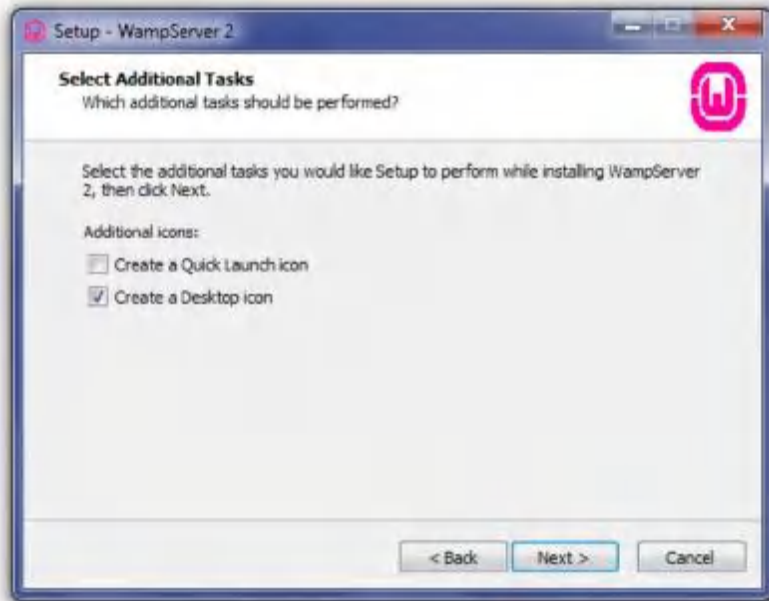
After this, the License Agreement screen will appear which will ask you to accept all agreements by selecting the radio button. Selecting the radio button and clicking on next will allow moving to next screen.



This screen will ask for the location where you have to select Destination Location. Unless you would like to install WampServer on another drive, you should not need to change anything. Click Next to continue.



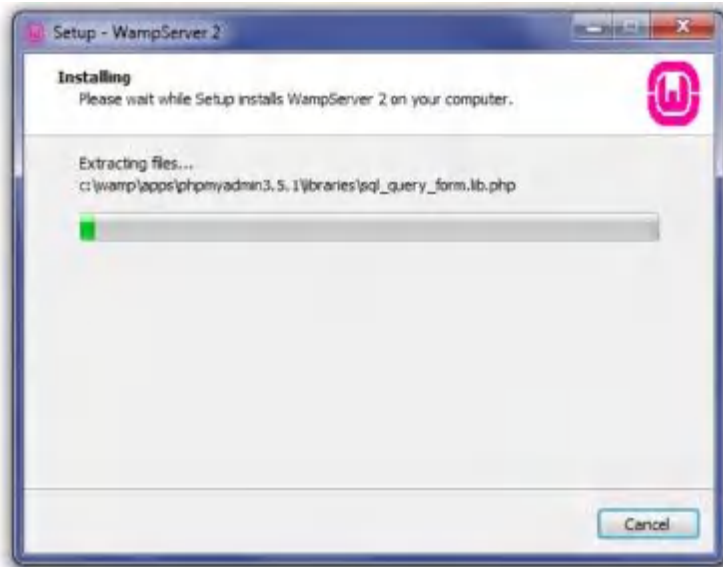
After this, the next screen shows Select Additional Tasks screen where you need to select whether you would like Quick Launch icon to be added to taskbar or Desktop icon after the installation. After making the selections, click Next to continue.



Clicking on Next will take you to Ready To Install screen where you can review your setup choices and change any by clicking Back to required screen. Once you have reviewed your choices, click Install to continue.



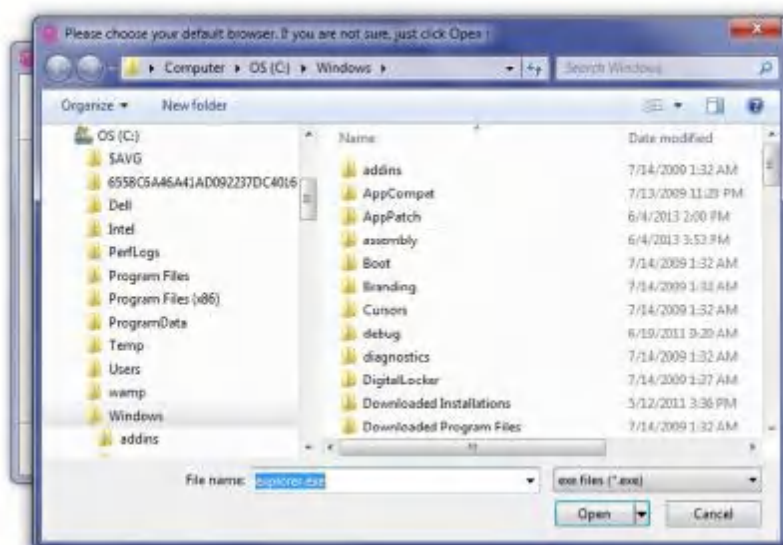
After this, the WampServer will start extracting files to required selected location.



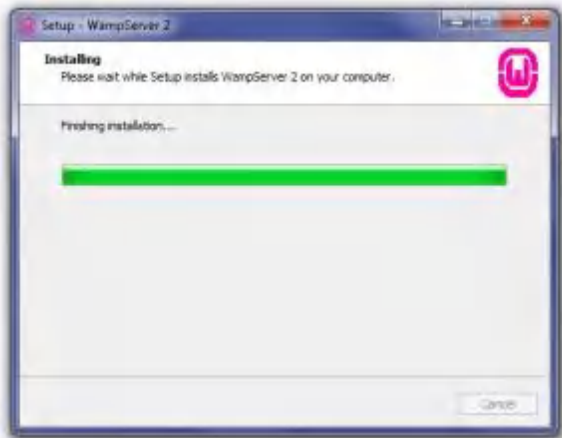
After extraction of files, you will be asked to select default browser. Wamp Server defaults to Internet Explorer upon opening the local file browser window. If your default browser is not an Internet Explorer, then find required .exe file:

- Opera: C:\Program Files (x86)\Opera\opera.exe
- Firefox: C:\Program Files (x86)\Mozilla Firefox\firefox.exe
- Safari: C:\Program Files (x86)\Safari\safari.exe
- Chrome:  
C:\Users\xxxxx\AppData\Local\Google\Chrome\Application\chrome.exe

Now select default browser's .exe file and click Open to continue.



On clicking you will find that a Windows Security Alert window will appear prompting that Windows Firewall has blocked some features of program. Here you need to check whether you want to allow Apache HTTP Server to communicate on private or public network and if this and then click Allow Access. You will see that the setup screen will appear showing status of installation process.



You will that the progress bar will work and once it is completely green, then PHP Mail Parameters screen will appear. Here you have to leave SMTP server local host and change email address to one of your selection. Click Next to continue.



After this you will find that the Installation complete screen will appear where you have to select Launch WampServer Now and click on Finish to complete installation.



Finally you will find WampServer icon in systray on right side of taskbar.

## Lab 2: To Create a php webpage and print “hello world”.

```
<Html>
<Head>
<Title> My Simple Program </Title>
</head>
<Body>
    <?php
        echo "Hello World"
    ?>
</Body>
</Html>
```

### Output:

Hello World

**Lab 3: To create a php program to find odd or even number from given number.**

```
<Html>
<Head>
<Title> Find out odd or even number. . ! </title>
</Head >
<body>
    <?php
    $a = 10;
    if ($a % 2==0)
    {
        echo "Given number is" . "<br>" . "<b>EVEN<b>" ;
    }
    else
    {
        echo "Given number is" . "<br>" . "<b>ODD<b>";
    }
    ?>
</body> </html>
```

**Output:**

Given number is  
EVEN

**Lab 4: To write a PHP program to swap two numbers.**

```
<html>
<head>
<title>
    Swapping of two numbers. . !
</title>
</head>
<Body>
    <?php
    $a = 10;
    $b = 20;
    echo "a = $a"."<br>". "b = $b"."<br>";
    $a = $a + $b;
    $b = $a - $b;
    $a = $a - $b;
    echo "<b>After Swapping"."<br>". "a = $a"."<br>". "b = $b<b>";
?>
</body>
</html>
```

**Output:**

```
a =10
b =20
After Swapping
a =20
b =10
```



### **Lab 5: Give PHP Example to calculate the area of the circle**

#### **Solution:**

```
<html>
  <body>
    <?php
      // defining constant value PI = 3.14
      define("PI","3.14");
      $radius=15;
      $area=PI*$radius*$radius;
      echo "Area=".$area;
    ?>
  </body>
</html>
```

#### **Output**

Area=706.5

### **Lab 6: To declare multiple variables in for loop.**

#### **Solution:**

```
<html>
  <body>
    <?php

      for ($x=0,$y=1,$z=2;$x<=3;$x++) {
        echo "x = $x, y = $y, z = $z <br>";
      }
    ?>
  </body>
</html>
```

#### **Output:**

x = 0, y = 1, z = 2   x = 1, y = 1, z = 2   x = 2, y = 1, z = 2   x = 3, y = 1, z = 2

**Lab 7: To declare a user defined function.****Solution:**

```
<html>
  <body>
    <?php

        // Function definition
        function myFunction()
        {
            echo "Hello world";
        }

        // Function call
        myFunction();
    ?>
  </body>
</html>
```

**Output:**

Hello world

### **Lab 8: Example of Numeric Array.**

#### **Solution:**

```
<html>
<body>
<?php
/* An array $flower_shop is created with three
Values - rose, daisy, orchid */
$flower_shop = array (
    "rose",
    "daisy",
    "orchid"
);
echo "Flowers: ".$flower_shop[0].",
    ".$flower_shop[1].", ".$flower_shop[2]."";
?>
</body>
</html>
```

#### **Output:**

**Flowers: rose, daisy, orchid**

**Lab 9: Give the example of multiple dimensional array.****Solution:**

```
<html>
<body>
<?php
$flower_shop = array(
    "rose" => array( "5.00", "7 items", "red" ),
    "daisy" => array( "4.00", "3 items", "blue" ),
    "orchid" => array( "2.00", "1 item", "white" ),
);
/* in the array $flower_shop['rose'][0], 'rose' indicates row and '0' indicates column */
echo "rose costs ".$flower_shop['rose'][0].
    ", and you get ".$flower_shop['rose'][1].".<br>";

echo "daisy costs ".$flower_shop['daisy'][0].
    ", and you get ".$flower_shop['daisy'][1].".<br>";

echo "orchid costs ".$flower_shop['orchid'][0].
    ", and you get ".$flower_shop['orchid'][1].".<br>";
?>
</body>
</html>
```

**Output:**

rose costs 5.00, and you get 7 items. daisy costs 4.00, and you get 3 items. orchid costs 2.00, and you get 1 item.

### Lab 10: Give the example of string function: substr():

#### Solution:

```
<html>
<head>
<title> String Function: Substr():1 </title>
</head>

<body>
  <b>If the start is non-negative, the returned string will start at the start'th position in string,
start from 0.</b><br/><br/>
  <?php
echo "Substring with positive start: “substr("abcdef",2)."<br/>";?><br/>
  <b>If the start is negative, the returned string will start at the start'th character in string,
from the end of the string.</b><br/><br/>
  <?php
echo "Substring with negative start: “substr("abcdef",-2)."<br/>";?><br/>
  <b>If the start is less than or equal to start characters long, false will return</b><br/><br/>
  <?php
echo "start is <= string”. Substr("abcdef",7)."<br/><br/>";
echo "Finish";
  ?>
</body>
</html>
```

O/P:

**If the start is non-negative, the returned string will start at the start'th position in string, start from 0.** Substring with positive start:cdef **If the start is negative, the returned string will start at the start'th character in string, from the end of the string.** Substring with negative start:ef **If the start is less than or equal to start characters long, false will return** start is <= string Finish

**Lab 11: Give the example of string function: strcmp()**

```
<html>
<head> <title> String Function:strcmp() </title> </head>
<body>
    <?php
    $str1 = 'a';
    $str2 = 'b';
    echo strcmp($str1, $str2)."<br/>";
    $str3 = 'b';
    $str4 = 'a';
    echo strcmp($str3, $str4)."<br/>";
    $str5 = "Anil";
    $str6 = "anil";
    echo strcmp($str5, $str6)."<br/>";
    ?>
</body></html>
```

**O/P:**

-1 1 -1

**Lab 12: Write a PHP program to create a database using MySQL.****Solution:**

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
    <?php
    $con = mysql_connect("localhost","root","");
    if(!$con)
    {
        die("not opened");
    }
    echo "Connection open."</br>";

    $query = "create database std";
    $crdb = mysql_query($query,$con);
    if(!$crdb)
    {
        die("not created. !".mysql_error());
    }
    echo "database created.. !";

    ?>
</body>
</html>
```

**O/P:**

Connection open database created.. !

### Lab 13: Write a program to use namespace

#### Solution:

```
<html>
<body>
<?php
namespace app\a{
class one{
public static function _1(){
echo 'a one _1<br>';
}
}
}
namespace app\b{
class one{
public static function _2(){
echo 'b one _2<br>';
}
}
}
namespace app{
echo a\one::_1();
echo b\one::_2();
echo a\two::_1();
}
namespace app\a{
class two{
public static function _1(){
echo 'a two _1<br>';
}
}
}
</body>
</html>
```



**Output:**

a one \_1

b one \_2

a two \_1

**Lab 14: Write a PHP program to create a table in MySQL.****Solution:**

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
    <?php
    $con = mysql_connect("localhost","root","");
    if(!$con)
    {
        die("not opened");
    }
    echo "Connection open."</br>";

    $db = mysql_select_db("studinfo",$con);
    if(!$db)
    {
        die("Database not found".mysql_error());
    }
    echo "Database is selected."</br>";

    $query = "create table computer(id INT not null,name varchar(50),branch varchar(50))";
    $crtb = mysql_query($query,$con);
    if(!$crtb)
    {
        die(" table not created. .!".mysql_error());
```

```

    }
    echo "table created.. !". "</br>";

    ?>
</body>
</html>

```

### Output:

Connection open Database is selected  
table created.. !

### Lab 15: Write a PHP program to insert record into a table using MySQL.

Solution

```

<html>
<head>
<title>Create Database. </title>
</head>
<body>
    <?php
    $con = mysql_connect("localhost","root","");
    if(!$con)
    {
        die("not opened");
    }
    echo "Connection open". "</br>";

    $db = mysql_select_db("studinfo",$con);
    if(!$db)
    {
        die("Database not found".mysql_error());
    }
    echo "Database is selected". "</br>";

```

```
$query = "insert into computer values(7009,'Anil J Basantani','Sadhana colony Jamnagar')";
```

```
$insrtb = mysql_query($query,$con);
```

```
if(!$insrtb)
```

```
{
```

```
    die("Record not inserted.".mysql_error());
```

```
}
```

```
echo "Record inserted successfully. . .!". "<br>";
```

```
?>
```

```
</body>
```

```
</html>
```

**Output:**

Connection open Database is selected Record inserted successfully. . .!

### Lab 16: Write a PHP program to drop table using MySQL.

#### Solution:

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
    <?php
    $con = mysql_connect("localhost","root","");
    if(!$con)
    {
        die("not opened");
    }
    echo "Connection open."</br>";

    $db = mysql_select_db("studinfo",$con);
    if(!$db)
    {
        die("Database not found".mysql_error());
    }
    echo "Database is selected."</br>";

    $query = "drop table ce";
    $crtb = mysql_query($query,$con);
    if(!$crtb)
    {
        die(" table not dropped. !".mysql_error());
    }
    echo "table dropped.. !".</br>";

    ?>
</body>
</html>
```

**Output:**

Connection open Database is selected  
table dropped.. !

**Lab 17: Write a PHP program to select data and show into table format.****Solution:**

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
    <?php
    $con = mysql_connect("localhost","root","");
    if(!$con)
    {
        die("not opened");
    }
    echo "Connection open."</br>";

    $db = mysql_select_db("studinfo",$con);
    if(!$db)
    {
        die("Database not found".mysql_error());
    }
    echo "Database is selected."</br>";

    $query = "select * from computer";
    $sldt = mysql_query($query,$con);
    if(!$sldt)
    {
        die("data not selected".mysql_error());
    }
    echo "<table border='1'>
```

```

<tr>
<th>ID</th>
<th>Name</th>
<th>Branch</th>
</tr>";
while($row = mysql_fetch_array($sldt))
{
echo "<tr>";
echo "<td>".$row['id']."</td>";
echo "<td>".$row['name']."</td>";
echo "<td>".$row['branch']."</td>";
echo "</tr>";
}
echo "</table>";

?>
</body>
</html>

```

### Output:

Connection open Database is selected

| ID | Name             | Branch |
|----|------------------|--------|
| 9  | Anil J Basantani | CE     |
| 9  | Anil J Basantani | CE     |
| 9  | Anil J Basantani | CE     |

## Lab 18: Update the data present in MYSQL database using web form

### Solution:

```
<html>
<body>
<?php
    // Open MYSQL server connection
    $db = mysql_connect("localhost", "root", "q1w2e3r4/");
    // Select the database using MYSQL server connection
    mysql_select_db("mydb", $db);
    if ($id) {
        if ($submit) {
            // Write UPDATE query and assign to $sql Variable
            $sql = "UPDATE employees SET
                first='$first', last='$last',
                address='$address',
                position='$position'
            WHERE id=$id";
            // Execute the query
            $result = mysql_query($sql);
            echo "Thank you! Information updated.";
        }
        else
        {

            // Write query to SELECT data from table
            $sql = "SELECT * FROM employees WHERE id=$id";
            // Execute the query
            $result = mysql_query($sql);
            // Fetch the values
            $myrow = mysql_fetch_array($result);
        }
    }
?>
<form method="post" action="<?php echo $PHP_SELF?>">
    <input type="hidden" name="id" value="<?php echo
        $myrow["id"] ?>">
```

```

First name:<input type="Text" name="first"
        value="<?php echo $myrow["first"] ?>"><br>
Last name:<input type="Text" name="last"
        value="<?php echo $myrow["last"] ?>"><br>
Address:<input type="Text" name="address"
        value="<?php echo $myrow["address"]?>"><br>
Position:<input type="Text" name="position"
        value="<?php echo $myrow["position"]?>"><br>
<input type="Submit" name="submit" value="Enter
        information">
</form>
<?php
    }
}
else
{
    // display list of employees
    $result = mysql_query("SELECT * FROM
        employees",$db);
    while ($myrow = mysql_fetch_array($result)) {
        printf("<a href=\"%s?id=%s\">%s %s</a><br>",
            $PHP_SELF, $myrow["id"],$myrow["first"],
            $myrow["last"]);
    }
}
?>
</body>
</html>

```



### **Lab 19 : To create PHP session**

#### **Solution:**

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// Echo session variables that were set on previous page
echo "Favorite picture is " . $_SESSION["favpicture"] . "<br>";
echo "Favorite plant is " . $_SESSION["favplant"] . ".";
?>
</body>
</html>
```

#### **Output:**

Favorite picture is junun.  
Favorite plant is rose.

### **Lab 20 :Write a program to change session variable**

#### **Solution:**

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// to change session variable, overwrite it
$_SESSION["favpicture"] = "junun";
print_r($_SESSION);
?>
```

```
</body>
</html>
```

### **Output:**

On running the above code, we will get the output as:

```
Array ( [favpicture] => junun [favplant] => rose )
```

## **Lab 21: Write a program to create ,modify and deleting cookies.**

### **Solution:**

#### **Create cookies**

```
<?php
$cookie_name = "user";
$cookie_value = "Sanjay Mathur";
setcookie($cookie_name, $cookie_value, time() + (30), "/"); //
?>

<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name])) {
echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
echo "Cookie '" . $cookie_name . "' is set!<br>";
echo "Value is: " . $_COOKIE[$cookie_name];
}
?>

</body>
</html>
```

Output:

```
Cookie named 'user' is not set!
```

#### **Modify cookies**

```
<?php
$cookie_name = "user";
```

```
$cookie_value = "Sanjay Mathur";
setcookie($cookie_name, $cookie_value, time() + (30), "/");
?>
<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name])) {
echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
echo "Cookie '" . $cookie_name . "' present<br>";
echo "Value is: " . $_COOKIE[$cookie_name];
}
?>
</body>
</html>
```

Output:

Cookie 'user' present

Value is: Sanjay Mathur

### **Delete a cookie:**

```
<?php
// set the expiration date
setcookie("user", "", time() - 36);
?>
<html>
<body>
<?php
echo "Cookie 'user' is deleted.";
?>
</body>
</html>
```

**Output:**

Cookie 'user' is deleted.