

BUREAU'S
HIGHER SECONDARY (+2)
INFORMATION TECHNOLOGY

PART - I

Approved by the Council of Higher Secondary Education,
Odisha for +2 Examination

BOARD OF WRITERS

Ranjan Kumar Bal

Department of Physics
B.J.B. Junior College, Bhubaneswar

Ajay Kumar Mishra

Department of Computer Science
P.N. Autonomous College, Khordha

Dhruba Charan Pradhan

Department of Information Technology
S.B. Women's College, Cuttack

REVIEWER

Nrusingha Charan Samal

Retired Reader in Physics



Published by
**THE ODISHA STATE BUREAU OF TEXTBOOK
PREPARATION AND PRODUCTION,
PUSTAK BHAVAN, BHUBANESWAR**

Published by :

The Odisha State Bureau of Textbook Preparation and Production
Pustak Bhavan, Bhubaneswar, Odisha, India.

First Edition : 2016 / 5000

Publication No : 177

ISBN : 978-81-8005-351-1

© Reserved by **The Odisha State Bureau of Textbook Preparation and Production, Bhubaneswar**. No part of this publication may be reproduced in any form without the prior written permission of the publisher.

Typesetting and Designed by : Print-Tech Offset Pvt. Ltd., Bhubaneswar

Printed at : Jagannath Process Pvt. Ltd., Cuttack

Price : ₹ 150/- (Rupees One Hundred Fifty Only)

Foreword

In recent past Information and Communication Technology (ICT) has caught the attention of academia, business, government and communities to use it for innovative profitable propositions. It is becoming common to use the devices like, desktop, laptop or palmtop computers, smart phones, digital & wireless gadgets for streaming videos, etc in public life. This century is characterized with the emergence of knowledge based society wherein ICT plays a pivotal role. In its vision, the National Policy on ICT in School Education by Ministry of Human Resource & Development, Govt. of India, states “The ICT policy in School Education aims at preparing youth to participate creatively in the establishment, sustenance and growth of a knowledge society leading to all around socio-economic development of the nation and global competitiveness”. In order to bring in uniformity in course curriculum over national level, Council of Higher Secondary Education (Odisha) has decided to orient the +2 courses in CBSE pattern. This would enable the students of the state to get a level playing field, in so far as the national level competitive examinations are concerned. The current syllabus of Informatics Technology has been revisited accordingly with a focus on generic concepts with domain specific practical experiments and projects to ensure conceptual knowledge with practical skills. . The societal impact of ICT has been discussed. A new Unit on IT Applications has been added to enhance the knowledge of e_Governance and e_Learning platforms available over the internet. Open Standards and Open Source to promote Vendor Neutrality of tools has been introduced. Creativity and Collaborative Learning/ Programming is also an added feature. For IDE based programming, Java is introduced in place of VB. Relational Database Management System is dealt with using My SQL in place of MS-Access.

The Council had constituted a board of writers comprising of professionals and experienced teachers and entrusted the task of publishing the Textbook on Information Technology to the Bureau.

I record my sincere thanks to the Council for allowing the Bureau to publish the text book for +2 students of the state. I do hereby express my appreciation and gratitude to the writers, for their co operation and diligence in preparing the manuscript of this book. My special thanks goes to Prof N.C Samal for meticulously reviewing the contents of the book before Publication.

Finally, I could not but extend my deep sense of appreciation to the officers and staff of Bureau, and specially to Sri Biraja Bhusan Mohanty, Deputy Director of the Bureau for his untiring efforts to make this new version press ready. Due care has been taken to make the book free from errors. But it is the first edition of the book! Therefore, I shall happily acknowledge the suggestions regarding any correction or improvement from its readers.

I hope all success for the book.

Dr. Geetika Patnaik
Director

Preface

Information and Communication Technology has been rapidly changing and its performance growth has been exponential, thanks to development in hardware as well as software sector all around. Initially the focus was on computer architecture and operating system techniques to support run-time programming with platform independence and to increase application programmers' productivity. Client could separately buy servers, Operating Systems, middleware, storage, etc. Next came the web browsers, leading to the emergence of connected computer systems and computing platforms. Key factors included TCP/IP and Ethernet as standard communication protocols and affordable PCs. IT service industry grew that led to outsourcing IT operations. This delivery model is now being challenged by another model of **cloud computing**. Information Technology that refers to the convergence of computer, communication and content technologies plays catalytic role in emergence of new socio-economic applications. Need of introducing IT as an elective subject to all the students of Arts, Commerce and Science stream was felt earlier to build problem-solving skills among the students just before they enter the professional courses. Accordingly Textbook on IT was published as per the syllabus of prepared by the CHSE (Odisha). The curriculum was revisited so as to make students better prepared for career in various industries requiring IT tools and applications. Therefore, it was felt to upgrade the existing Text book as per the revised syllabus. This is the beginning of preparing a text book for +2 1st year classes on Information Technology.

In this new version, we have introduced Open Standards and Open Source aiming at Vendor Neutrality of tools, Creative computing and Collaborative Learning environment, development of Front End Interface, Back End Database and connectivity of both, so that a student would get a complete feel of Application Development. A Unicode Chart of Odia Characters is being provided at the end of Chapter-3 for the benefit of students & teachers

The authors would like to express their appreciation of the expertise, understanding and kindness displayed by the Director, Dy Director Sri Biraj Bhusan Mohanty and staff of the Bureau in the preparation of these books. Authors are grateful to Dr Brundaban Pradhan, Lecturer in Physics, B J B Junior College, Bhubaneswar for providing valuable logical and material support during the preparation of this book.

Akshayatrutiya
The 09th May, 2016

Authors

CHSE SYLLABUS
+2 1st Year Class (IT Syllabus) 2016

Unit 1: Introduction to Computer Systems

(A) Hardware Concepts: (10 periods / 10 mark)

- (i) **Computer organization** : CPU, Memory (RAM and ROM), I/O devices, communication bus, ports (serial, parallel), device specific ports;
- (ii) **Input devices:** Keyboard, Mouse, Light pen, Touch screen, Graphics Tablet, Joystick, Microphone, OCR, OMR, Scanner, Smart Card reader, Barcode reader, MICR scanner, Biometric sensor, Web camera;
- (iii) **Output Devices:** Monitor/Visual Display Unit (VDU), LCD/LED screens, Television, Printer (Dot Matrix Printer, DeskJet/ Inkjet/ Bubble jet Printer, Laser Jet Printer), Plotter, Speaker;
- (iv) **Memory Units:** Computer memory, Types of memory- RAM (static & dynamic), ROM (PROM, EPROM, EEPROM), Registers.

(B) Types of Software: (10 periods / 10 mark)

- (i) **System Software :**
 - (a) Operating Systems (OS), Need for operating system, major functions of Operating System; OS for mainframe, PC/Server, mobile devices.
 - b) Language Processors: Assembler, Interpreter, and Compiler
- (ii) **Utility Software :** Compression tools, disk defragmenter, anti-virus.
- (iii) **Application Software :**
 - (a) General Purpose Application Software: Word Processor, Spreadsheet Packages, Presentation software, Database Management System, and Integrated Development Environment (IDE)

- (b) Specific Purpose Application Software: Inventory Management System, Human Resource Management System, Payroll System, Financial Management System, Reservation System.

(C) Open Source Concepts: (10 periods / 05 mark)

- (a) Open Source Software (OSS), common FOSS/FLOSS (e.g., GNU/Linux, Firefox, OpenOffice, Java, Netbeans, MySQL), common open standards (WWW, HTML, XML, DHTML, TCP, IP).
- (b) Indian Language Computing: Character encoding, UNICODE, different types of fonts (open type versus true type, static versus dynamic), Entering Indian Language Text - phonetic & key map based.

Unit 2: Introduction to Programming

- (i) **Getting started with Programming using IDE:** (20 periods / 11 mark)

Introduction, Rapid Application Development using IDE (Integrated Development Environment); Basic Interface components- Label, Text Field, Text Area, Button, Checkbox, Radio Button.

Developing General Application: Getting Familiar with Java Swing User Interface components-Frame, Dialog, Option Pane, Panel, Scroll Pane, Label, Text Field, Password Field, Text area, Button, Check Box, Radio Button, Combo Box, and List.

Basic component handling methods and properties: setText(), getText(), isSelected(), setSelected().

- (ii) **Programming Fundamentals:** (15 periods / 10 mark)

Data Types: Concept of data types; Built-in data types - Byte, Short, Int, Long, Float, Double, Char, String, Boolean

Variables: Need to use variable, declaring variables, variable naming convention, assigning value to variables;

Integer object method: parseInt

Double object method: parseDouble, parseFloat

Control structures:

- (a) Decision structure - if, if-else, switch;
- (b) Looping structure- while, do—while, for;

(iii) **Programming Guidelines:** (10 periods / 04 mark)

General concepts, modular approach, Running and debugging programs, syntax errors, run-time errors, logical errors;

Problem solving procedure: Understanding of the problem, Identifying minimum number of inputs required for output, breaking down problem into simple logical steps.

Unit 3: Relational Database Management System

(i) **Database Management System :** (10 periods / 10 mark)

Introduction to database concepts: Database, relational Database, Relation/Table, Attribute/Field, Tuple / Row;

Data types: Text (CHAR, VARCHAR), Number (DECIMAL, INT/ INTEGER), Date and Time

Keys: Candidate key, Primary key, Alternate key, Foreign key;

Examples of common database management tools for mobile devices

(ii) **Introduction to MySQL :** (13 periods / 05 mark)

(ANSI SQL 99 standard commands)

Classification of SQL Commands:

DML: - SELECT, INSERT, UPDATE, DELETE

DDL:- CREATE, DROP, ALTER

Creating and using a database: SQL CREATE command to create a database, USE command to select a database.

Creating a table: CREATE command to create a table, DESC command to display a table structure, INSERT command for inserting new rows

Displaying table data: SELECT command for selecting all the columns, selecting specific column(s) using arithmetic operators, operator precedence

Defining and using column alias, Eliminating duplicate values from display using DISTINCT keyword, Limiting rows during selection (using WHERE clause), Using Comparison operators (=, <, >, <=, >=, <>, BETWEEN, IN, LIKE(%,_)), Logical Operators(AND, OR, NOT) and corresponding operator precedence.

Unit 4: IT Applications

- (i) **e-Governance** : (04 periods / 03 mark)

Definition, benefits to citizens, e-Governance websites- their salient features, and societal impacts; e-Governance challenges.

- (ii) **e-Learning** : (03 periods / 02 mark)

Definition, benefits to students (learners), teachers (trainers) and Institution management; e-Learning websites and their salient features and societal impacts.



CONTENTS

Unit 1 :	Page
INTRODUCTION TO COMPUTER SYSTEMS	
Chapter - 1 : Hardware Concepts:	1.1 - 1.42
Computer	
Input devices	
Output Devices	
Memory Units	
Chapter - 2 : Software Concepts	2.1 - 2.30
System Software:	
Operating Systems	
Language Processors	
Utility Software :	
Compression tools	
Disk defragmenter	
Anti-virus	
Application Software:	
General Purpose Application Software:	
Word Processor, Spreadsheet Packages	
Presentation software	
Database Management System	
Integrated Development Environment (IDE)	
Specific Purpose Application Software	
Inventory Management System	
Human Resource Management System	
Payroll System, Financial Management System	
Reservation System	
Chapter - 3 : Open Source Concepts	3.1 - 3.16
Open Source Software (OSS) : FOSS/FLOSS	
Common Open Standards	
WWW, HTML, XML, DHTML, TCP, IP	

Indian Language Computing

Character encoding

UNICODE

Types of Fonts (open type versus true type, static versus dynamic),

Entering Indian Language Text:- phonetic & key map based.

UNICODE chart for Odia scripts.

Unit 2 :

INTRODUCTION TO PROGRAMMING

Chapter - 4 :	Programming using IDE	4.1 - 4.44
	Rapid Application Development using IDE	
	Basic Interface components	
	Label, Text Field, Text Area, Button, Checkbox, Radio Button.	
	Developing General Application	
	Java Swing User Interface components	
	Frame, Dialog, Option Pane, Panel, Scroll Pane,	
	Label, Text Field, Password Field, Text area,	
	Button, Check Box, Radio Button, Combo Box, and List.	
	Basic component handling methods and properties	
	setText(), getText(), isSelected(), setSelected()	
Chapter - 5 :	Programming Fundamentals	5.1 - 5.46
	Data Types	
	Built-in data types - Byte, Short, Int, Long, Float, Double,	
	Char, String, Boolean	
	Variables : Declaring variables, variable naming convention,	
	assigning value to variables	
	Integer object method: parseInt	
	Double object method: parseDouble, parseFloat	
	Control structures:	
	Decision structure - if, if-else, switch	
	Looping structure- while, do—while, for	
Chapter - 6 :	Programming Guidelines	6.1 - 6.14
	General concepts, modular approach	
	Running and debugging programs	
	Syntax errors, run-time errors, logical errors	
	Problem solving procedure	

Unit 3 :

RELATIONAL DATABASE MANAGEMENT SYSTEM

Chapter - 7 :	Database Management System	7.1 - 7.16
	Introduction to database concepts Database, relational Database, Relation/Table, Attribute/Field, Tuple / Row Data types Text (CHAR, VARCHAR), Number (DECIMAL, INT/INTEGER) Date and Time Keys: Candidate key, Primary key, Alternate key, Foreign key Database management tools for mobile devices	
Chapter - 8 :	Introduction to MySQL	8.1 - 8.38
	Classification of SQL Commands: DML:- SELECT, INSERT, UPDATE, DELETE DDL: - CREATE, DROP, ALTER Creating and using a database SQL CREATE command to create a database USE command to select a database DESC command to display a table structure INSERT command for inserting new rows SELECT command for selecting all the columns	

Unit 4 :

IT APPLICATIONS

Chapter - 9 :	e-Governance	9.1 - 9.14
Chapter - 10 :	e-Learning	10.1 - 10.8



Unit

1

INTRODUCTION TO COMPUTER SYSTEMS

CHAPTER - 1

Hardware Concepts

CHAPTER - 2

Software Concepts

CHAPTER - 3

Open Source Concepts

UNIT - 1

INTRODUCTION TO COMPUTER SYSTEMS

CHAPTER - 1

HARDWARE CONCEPTS

1. A HARDWARE CONCEPTS :

Introduction

The term 'information' refers to the facts and opinions provided and received during the course of our daily life. We gather information, while interacting with living beings surrounding us, mass media, electronic media and all sorts of observable phenomena. Interest and necessity of information system has long been a part of the mankind since pre-historic era. The ancient man used to communicate with each other through symbolic and verbal actions. Time has passed and the system of communication has developed from action to alphabets. Modern day information system has grown to such a proportion that it has become an object of study in a number of disciplines, including philosophy, physics, biology, linguistics, information and computer science, electronic and communications engineering, management science, and the social sciences.

On the commercial side, the information service industry has become one of the newer industries worldwide. Manufacturing sectors, Service sectors, & others are considering the Information handling as an economic resource, somewhat on par with other resources like labour, material, and capital. This view stems from evidence that the possession, manipulation, and use of information can increase the cost-effectiveness of many physical and cognitive processes. The rise in information-processing activities in industrial manufacturing as well as in human problem solving has been remarkable.

As a societal resource, information has some interesting characteristics that separate it from the traditional notions of economic resources.

- (i) It is expansive, with limits apparently imposed only by time and human cognitive capabilities.
- (ii) It reproduces rather than being consumed through use
- (iii) It can be shared only, not exchanged during transactions.
- (iv) It is transportable at very high speed

Application of scientific knowledge to change and manipulate the human environment is often called as technology.

Thus information technology (IT) embodies scientific acquisition, recording, organization, retrieval, display, and dissemination of information for the betterment of the human society. It involves four basic operations- **gathering, storing, processing** and providing the **output** information to the outside world. This process has been there since ancient times. Before the advent of modern digital computer, the information was stored in analog form through printing, photography and telephony. Modern information technology greatly facilitates the storing and manipulation of information as a result of its representation in digital form.

IT literacy includes knowledge of technical tools, skill and comfort in the use of technical hardware and software packages, understanding of the ethical issues related to technology, and an understanding of technology's impact on society.

In our day to day life, we are gradually being driven towards automation in various mundane activities like; washing clothes by washing machines, cooking by microwave ovens, conventional banking transactions through Automated Teller Machines (ATM), etc. In all these machines, a click of a button starts a process inside the machine which follows a predefined work flow based on which button has been pressed. Broadly speaking, the machine receives certain data from the user, processes it and gives the output (washed clothes, cooked food or currency notes). This is exactly a computer system does.

1.A.1 Basic Computer Organisation:

A computer is an electronic device (hardware) that processes input data and produces result (output) according to a set of instructions called program (software). In this section we shall try to outline the basic structure and function of a modern digital computer. Digital computer systems came in to being during 1940s with the design of ENIAC, EDVAC and Harvard Mark I. except ENIAC;

all other systems are based upon a common technology of “stored program” proposed by John von Neumann. All modern digital computer systems till date are based upon that von Neumann technique. Every system performs the following five basic operations to accomplish a job. Though the internal architecture of different computers varies from one system to another, the basic organization remains the same. The block diagram given here gives a basic layout of a modern digital computer system.

1. **Input:** It accepts data and instructions by way of inputs.
2. **Store:** It stores data and instructions for the use of the system as and when required.
3. **Processing:** It processes (carries out arithmetic or logical operations on) data and convert them to useful information.
4. **Output:** It presents the final result for the user in form of output (printed or visual display format).
5. **Control:** It controls (co-ordinate and synchronize) all four operations described above.

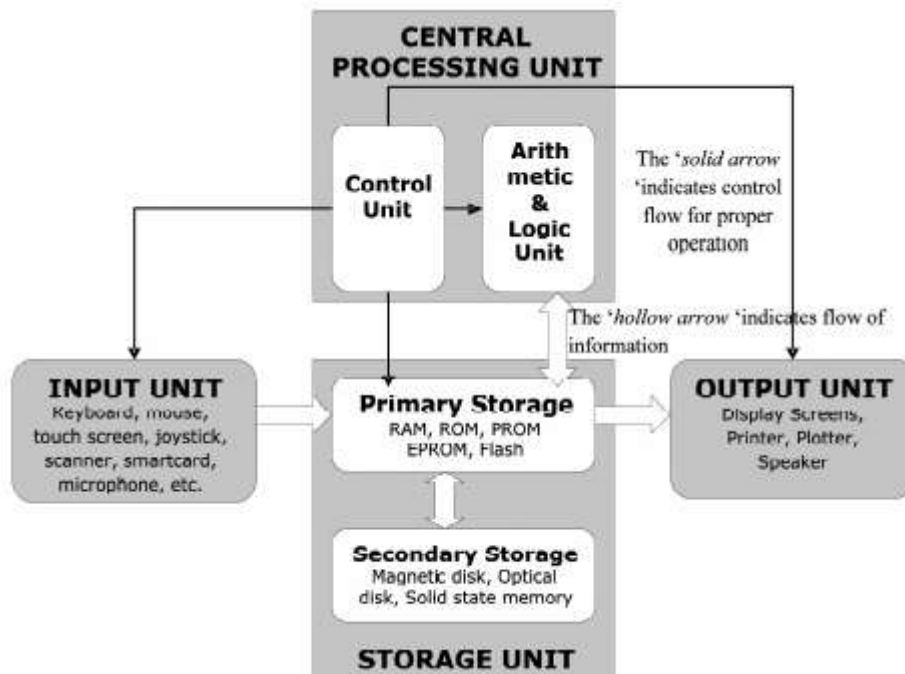


Figure 1.A.1 : Block diagram of functional units of a computer Introduction to Computer System

1.A.2 Hardware Concepts :

We shall now discuss briefly the structure and working of different sections of a digital computer as shown in the previous diagram. Hardware represents the physical and tangible components of the computer itself, i.e. components which can be seen and touched. It includes

1. Input devices
2. Central Processing Unit
3. Output devices
4. Memory / Storage Unit

Central Processing Unit :

The Central Processing Unit (CPU), or simply the processor, is a component in a digital computer which is capable of executing a program. It is considered to be the 'brain' of a computer system. It interprets program instructions and processes data. All calculations, comparisons are carried out inside the CPU and CPU sends instructions to other parts of the computer for their smooth and synchronized operations. CPUs provide the digital computer, trait of programmability. It is one of the necessary components found in computers of any era, along with primary storage and input/output facilities. A CPU that is manufactured as a single integrated circuit is usually known as a microprocessor (A microprocessor is a programmable device, that takes in numbers, performs arithmetic and logical operations on them according to the instructions stored in the memory and produces other numbers as a result). Beginning in the mid-1970s, microprocessors of ever-increasing complexity and power gradually supplanted other designs, and today the term "CPU" is usually applied to some type of microprocessor. Some of the processors, presently in use are shown here.



Figure 1.A.2 : Microprocessors

As has been shown previously, the CPU contains the Control Unit and the Arithmetic and Logic Unit.

(i) **Control Unit :**

This unit manages and co ordinates the functions of the entire computer system. It obtains instructions from the program stored in main memory, decodes and interprets them and issues signals to other components to execute them (Remember, this unit does not process any data!). To carry out all these operations the Control Unit has some special purpose **registers** (A register is nothing but a collection of special memory units to hold the information on a temporary basis; they facilitate speedy transfer of information between different parts of a CPU) and a decoder. The special purpose registers are: **Instruction Register & Program Counter Register**. The first register holds the current instruction and later one holds the address of the next instruction in the main memory. The decoder consists of the necessary circuitry to decode and interpret the meaning of every instruction supported by the CPU. Each instruction is accompanied by an **op-code**, which suggests the nature of its execution.

(ii) **Arithmetic Logic Unit :**

This unit performs all arithmetic (addition, subtraction, multiplication and division) and logic (less than, greater than, equal to etc.) operations as and when instructed by the Control Unit. Like Control Unit, this unit has also some special purpose registers like **Accumulator Register, Data Register** and some general purpose Temporary Registers.

The size of a register equals the number of bits required by a CPU to store a character or a number. Thus, the registers are often called as 8-bit, 16-bit, 32-bit or 64-bit registers. The bigger the word size, the faster the computer can process a set of data. Different CPUs possess registers of different sizes and their number also varies from processor to processor. Most recent microprocessors are using 64-bit registers. Each register is assigned to hold a particular type of information.

Working of a CPU :

The fundamental operation of most CPUs, regardless of the physical form they take, is to execute a sequence of stored instructions called a program. The program is represented by a series of numbers that are kept in some kind of computer memory. There are four basic steps that nearly all von Neumann CPUs use in their operation: fetch, decode, execute, and write back.

- (1) **Fetch:** The instruction is retrieved (by the control unit) from the memory and fed in to the Instruction Register where the operation and address parts are separated.
- (2) **Decode:** The control unit then sends the operation part to the decoder and the address part to the MAR.
- (3) **Execute:** The decoder interprets the instruction, and accordingly the control unit sends signals to the appropriate units to carry out the specified task. For example, for any arithmetic or logic operation the signal is sent to the ALU. Besides instructing ALU to carry out a particular operation, the control unit ensures that the address part of the instruction is loaded to a suitable register in the ALU from where it can read the data. The ALU carries out the necessary operation and sends a signal to the control unit as soon as the operation is finished.
- (4) **Write back:** The result is written back either to a register in the CPU for further processing or a slower but cheaper and larger main memory of the computer. As each instruction is executed, the address of the next instruction is automatically loaded in to the PC register and step 1 to 4 is repeated.

Clock Speed :

The above operations take place inside a computer with incredible speed. In order to synchronize each operation every CPU has a built-in electronic clock inside it. This clock emits millions of electric pulses at precise and regular intervals and is called as a system clock. The periodic pulses (signals) are termed as clock cycles. One cycle is the time it takes to perform one operation, such as moving a byte of data from one memory location to another. Therefore, several clock cycles are required to completely execute a single program instruction following the above 4 steps. The shorter the clock cycle, the faster is the processor. Hence, the speed with which an instruction is executed is directly related to the clock speed which is measured as the number of pulses emitted per second (hertz). Presently, most of the microprocessor clocks possess a speed of 1- 2 Giga hertz (10⁹ Hz). A CPU having faster clock speed can process data faster.

Main Memory :

The main memory of a computer system is otherwise called as Primary Storage Unit. When we feed data and instructions to a computer for processing,

it gets stored in its main memory. The main memory is outside the CPU. It possesses the following characteristics;

- The storage capacity of this memory is much larger compared to the registers in the CPU (which barely stores a byte of information!), so that it can store a complete program and other information.
- The storage mechanism is electronic in nature, in such a way, that data retrieval does not require any physical movement of the components. Thus the access time is much faster in comparison to the magnetic disk storage units.
- Most part of this memory is volatile in nature, i.e. the stored information gets wiped out, the moment power supply is off. Some portions are non volatile – which retain the stored information (all are in binary form) even if the power supply is off.
- The most important characteristic of this memory is that, each site (location) is equally accessible. The data retrieval time is independent of its storage position. Such a facility is often called as **Random Access**; because one can randomly access data from any location of the memory circuitry with equal ease.
- Because of this random access property, the main memory is often called as Random Access Memory or RAM or simply the **memory** of a computer.
- When information is read from the memory locations, the contents of the locations remain intact-called as memory read operation. When some new information is written to these locations the previous data is over written-called as memory write operation.

The capacity of memory is defined as the number of bytes it can store. It is expressed in terms of kilobytes ($\text{kB} = 2^{10}$ bytes = 1024 bytes), megabytes ($\text{MB} = 2^{10}$ kB = 1048576 bytes) or gigabytes ($\text{GB} = 2^{10}$ MB = 1073741824 bytes). Remember kilo; mega and giga are decimal suffixes for 1000, 1000000 and 1000000000 times. Here we are using these suffixes, just because of the closeness of the corresponding binary values (1000:1024; 1000000:1048576; 1000000000:1073741824)!

Memory Partition- RAM, ROM, PROM, EPROM :

The key benefit of **RAM** over other types of storage is its short retrieval time and consistency. *Short* because no physical movement is necessary, and

consistent because the time taken to retrieve a piece of data does not depend on its current distance from a physical head; it requires practically the same amount of time to access any piece of data stored in a RAM chip. Because of this speed and consistency, RAM is used as main memory' or primary storage. In most personal computers, the RAM is not an integral part of the motherboard or CPU—it comes in the easily upgraded form of modules called *memory sticks* or



Figure 1.A.3 : RAM sticks

RAM sticks. These can quickly be removed and replaced should they become damaged or too small for current purposes. Empty memory slots are available for this purpose in motherboards. A RAM basically stores information until it is needed by different parts of the system. As of today, there are two types of RAM architectures being used in different computer systems- Static RAM (SRAM) and Dynamic RAM (DRAM). These two technologies are different from each other in their memory cell architecture, function and power consumption. Let us briefly outline the features of both these RAMs.

Static RAM:

The word 'static' indicates that the memory retains its contents as long as power is on. Each memory location of SRAM consists of six transistors. These are used in personal computers, workstations, routers and peripheral equipment: internal CPU caches, hard disk buffers, router buffers, etc. LCD screens and printers also employ static RAM to hold the image displayed (or to be printed). Small SRAM buffers are also found in CDROM and CDRW drives; usually 256 KB or more are used to buffer track data, which is transferred in blocks instead of as single values. The same applies to cable modems and similar equipment connected to computers. The so called CMOS RAM on PC motherboards was originally a battery-powered SRAM chip.

Dynamic RAM :

It is a type of random access memory that stores each bit of data in a separate capacitor within an integrated circuit. Since real capacitors leak charge, the information eventually fades unless the capacitor charge is refreshed

periodically. Because of this refresh requirement, it is a *dynamic* memory as opposed to SRAM and other *static* memory. Its advantage over SRAM is its structural simplicity: only one transistor and a capacitor are required per bit, compared to six transistors in SRAM. This allows DRAM to reach very high density. Like SRAM, it is in the class of volatile memory devices, since it loses its data when the power supply is removed. A type of solidstate DRAM is called as synchronous DRAM is called as SDRAM are also being used. It is called synchronous in a sense that it waits for the clock signal before responding to its control input. The available modules of packaged DRAMs are SIMM-30pin, SIMM-72pin, DIMM-168pin etc.

Software can partition a portion of a computer's RAM, allowing it to act as a much faster hard drive that is called a RAM disk. Unless the memory used is non-volatile, a RAM disk loses the stored data when the computer is shut down. However, volatile memory can retain its data when the computer is shut down if it has a separate power source, usually a battery.

If a computer becomes low on RAM during intensive application cycles, the computer can resort to so-called virtual memory. In this case, the computer temporarily uses hard drive space as additional memory. Constantly relying on this type of backup memory it is called **thrashing**, which is generally undesirable, as virtual memory lacks the advantages of RAM. In order to reduce the dependency on virtual memory, more RAM can be installed.

No matter, every location of the main memory is randomly accessible; all of them are not volatile like RAM chips. Non volatile and random access memory do constitute a part of the main memory; the information stored in these locations is permanent, which can be read, whenever required, but cannot be changed. These are called as **Read Only Memory (ROM)**. The data is stored in ROM chips by using fuse links and the process is called as burning in the data . ROMs are mainly used to store programs and data which do not change and are frequently used by the computer. Micro programs (special programs written by the manufacturer to substitute some complex electronic circuitry to carryout complex and repetitive tasks) are generally stored in these ROMs. These micro programs



Figure 1.A.4 : ROM chip

which essentially substitute a hardware component of a computer are called as **firmware**. An example of this micro program is system boot program which make the computer system ready for use when the power is switched on. It contains a set of start-up instructions to check the status of memory, input output devices and other part of the computer and looks for an operating system and loads its core part in the volatile RAM to produce the initial display-screen prompt.

There are two types of ROMs-one manufacturer programmed and other to be programmed by the user. The later one which is left to be programmed by the user permanently is called as **Programmable Read Only Memory (PROM)**. In order to customize a system a user may convert his/her program in to micro programs and store it in the PROM chip. PROMs are programmed to record information using a special device, known as PROM-programmer. However, once the chip has been programmed, the recorded information can not be changed. A PROM is a non-volatile storage part of the main memory.

Other than ROM and PROM another type of memory chip used in the main memory is called as **EPROM (Erasable and Programmable Read Only Memory)**. This is a non-volatile memory and stores information permanently. However, in worst cases, the information can be altered either by exposing the chip for some time to Ultra Violet light or by using high voltage electric pulses. The EPROMs in which the data is erased by UV radiation are known as UVEPROM and the EPROMs which employ the second technique to erase the data are called as EEPROMs. Generally the EPROMs are used by R&D personnel while configuring new system hardware. Due to the ease with which the stored programs can be erased by using electric pulses, the EEPROMs are also called as **Flash Memories**.

Cache Memory :

As discussed earlier, information stored in main memory (RAM) can be accessed by the processor in very short time- around 60 nanoseconds (ns). But the processor executes that information in just 2 ns! This means, the processor seats idle for 58ns during one operation! It makes the processing slower and decreases the efficiency of a computer. The need to overcome this limitation gave rise to the concept of Cache memory. (Cache' is a French word means to hide).

Cache memory or Cache is a small but faster (access time: 10ns to 30ns) supplementary memory system, that temporarily stores frequently used

instructions and data during a program execution for quicker processing. It is actually an extension of computer main memory and random access in nature.

Secondary Storage Devices :

The main memory has two distinct limitations- short storage capacity and volatility. The maximum storage capacity of a RAM chip (main memory) is around 4GB, whereas the modern computers require a storage capacity in excess of 10GB. Because of volatility, the main memory loses the stored information once the power is off. These two limitations of the main memory are taken care of by the so called **auxiliary memory** or **secondary storage**. These are non volatile, high capacity, cheap but slower memory devices. They are used to store a large volume of data on a permanent basis. As and when required, main memory extracts data from these secondary storage devices for processing. Normally hard disk of computer is used as secondary memory but this is not portable so there are many other secondary storage media in use.

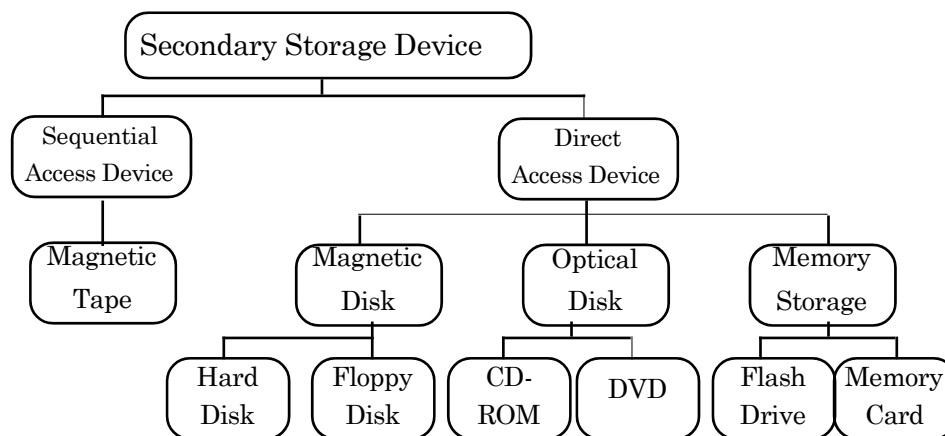


Figure 1.A.5 : Classification of Secondary Storage Devices

Hard Disk Drive : It consists of a stack of magnetic disks and read/write assembly. The disks are made up of a non-magnetic material, usually aluminum or glass, and are coated with a very thin layer of magnetic material. Older disks used iron-oxide as the magnetic material, but current disks use a cobalt-based alloy. The magnetic surface of each platter is divided into many small sub-micrometre-sized magnetic regions, each of which is used to encode a single binary unit of information. The disks are mounted on a spindle, with spacers in

between, and a motor on the bottom end of the spindle for spinning of the disks. To read and write to the surface of the disks, the drive uses a small electro-magnet assembly, referred to as a head, located on the end of an actuator arm. There are as many heads as the number of platter surfaces. The read/write head moves back and forth in a radial pattern with the help of a voice coil (the same technique used to move the cone of a speaker on your stereo). The read element of the head is magneto resistive, whereas the write head is typically thin film inductive. Hard disk drives are sealed to prevent dust and other sources of contamination, from interfering with the operation of the hard disks heads. Using rigid disks and sealing the unit allows much tighter tolerances than in a floppy disk drive. These drives are fitted in the system's cabinet.

This is a high capacity storage device ranging from 1GB to few Tera Bytes nowadays.

HDDs were originally developed for use with computers. In the 21st century, applications for HDDs have expanded beyond computers to include digital video recorders, digital audio players, personal digital assistants, digital cameras, and video game consoles. In 2005 the first mobile phones to include HDDs were introduced by Samsung and Nokia.

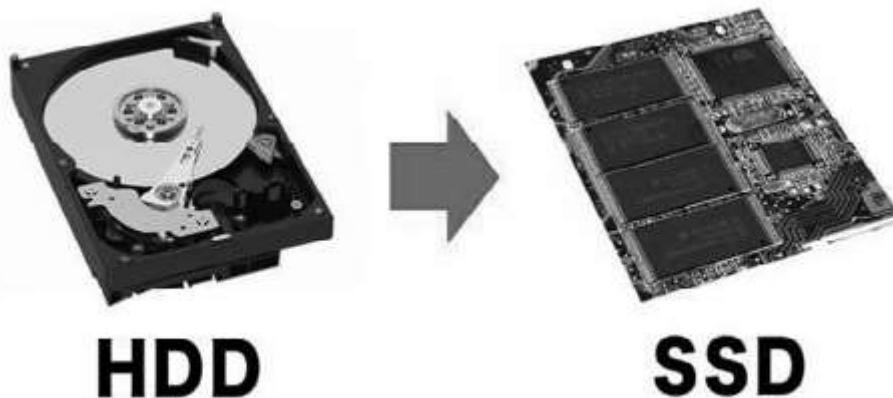


Figure 1.A.6 : Hard Disk Drive & Solid State Disk(showing flash memory units)

Solid State Disk: An SSD does much the same job functionally (e.g., saving data while the system is off, booting the system, etc.) as an HDD, but instead of a magnetic coating on top of platters, the data is stored on interconnected **flash memory** chips that retain the data even when there's no power present.

Flash memory is a type of EEPROM chip, which stands for Electronically Erasable Programmable Read Only Memory. It has a grid of columns and rows with a cell that has two transistors at each intersection. It has no moving parts, so it's noiseless. It allows faster access and smaller in size. This memory transfers data at a faster rate of 30 Mega Bytes per second!



Figure 1.A.7 : Solid State Disk

The chips can either be permanently installed on the system's motherboard (like on some small laptops and ultrabooks), on a PCI/PCIe card (in some high-end workstations), or in a box that's sized, shaped, and wired to slot in for a laptop or desktop's hard drive (common on everything else). These flash memory chips differ from the flash memory in USB thumb drives in the type and speed of the memory. The capacity ranges from 100 GB to 4 TB.

Floppy Disk: A floppy disk consists of a special and flexible plastic disk, coated on either side, with a magnetic material. The disk is encased in a square plastic or vinyl jacket cover. Because the disk is flexible, it is referred to as floppy disks or simply 'floppies'. Floppy disks were first introduced in 1972 by IBM Inc. USA. At present, most commonly used floppies are of 3.5" size. These are double sided disks and store data on both surfaces. The storage capacity of these floppies varies from 720kB to 2.88MB. These are the cheapest off line secondary storage devices for small computers.



Figure 1.A.8 : Floppy Disk

Optical Disks: The secondary storage disk, which stores binary data in form of **lands** and **pits** to produce constructive and destructive interference of laser light are termed as Optical Disks. Most of these disks are read only memories, because the lands and pits formed once on the disk become permanent. The basic difference between a magnetic disk and the optical disk is their storage pattern. A magnetic disk has concentric circular tracks whereas optical disks possess a spiral track from centre towards the outer edge. Due to the similarity

with audio CD, the optical disks are called as CD-ROMs (compact disk- read only memory). Capacity of standard 120mm CD is 700MB. It is a thin optical disk which is commonly used to store audio and video data. Transfer speed is mentioned as multiple of 150 KB/s. 4x means 600 KB/s. In this magnetic coatings are stored as data on a thin tape. Earlier this medium was used for archive purposes.

Magnetic Tape: It is sequential storage medium used for data collection, backup and archiving. Like videotape, computer tape is made of flexible plastic with one side coated with a ferromagnetic material. Tapes were originally open reels, but were superseded by cartridges and cassettes of many sizes and shapes.



Figure 1.A.9 : Magnetic Tape Cartridge

DVD: Digital Versatile Disc or Digital Video Disc. This is an optical disc storage device. Each writable layer of a DVD has a spiral track of data. On single-layer DVDs, the track always circles from the inside of the disc to the outside. The data track and the pits are incredibly tiny. The tracks of a DVD are separated by just 740 nanometers while that of a CD-ROM, the separation is around 1600nm. Similarly the pits are also smaller in case of DVD as well. The minimum pit length is 400nm for a DVD in comparison to 830nm for a CD-ROM. The width of a DVD track is 320nm and the height of bumps is 120nm.



Figure 1.A.10 : Digital Versatile Disc

A DVD has the same dimensions as a standard CD but cannot be read by a standard CD player; although a DVD player can read standard CDs. DVD players use a higher-power red laser (0.65 micrometer of wavelength) that enables smaller pits and separation tracks to be used. It can be recorded on single side or on double side. Its capacity may range from 4.7 GB to 8.5 GB

USB Drive: This is a class of flash memory, which is small, portable and can be plugged into a computer with USB Port. They have capacity lesser than hard disk but much larger than a floppy or CD. They are more reliable also. They are also called as pen drives.



Figure 1.A.11 : USB Pen Drive

Memory Cards: A memory card (sometimes called a flash memory card or a storage card) is a small storage medium used to store data such as text, pictures, audio, and video, for use digital cameras, computers, mobile phones, music players, video game console etc. Most of the current products use flash memory, although other technologies are being developed. There are a number

of memory cards on the market, including the SD card (secure digital card), the CF card (CompactFlash card), the SmartMedia card, the Memory Stick, and the MultiMediaCard (MMC). These cards are of varying sizes, and each is available in a range of storage capacities. The CompactFlash card is about the size of a matchbook, while the MultiMediaCard and Secure Digital card are each about the size of a postage stamp. Most



Figure 1.A.12 : Micro SD Card

available cards have constantly powered nonvolatile memory, which means that data is stable on the card, is not threatened by a loss of power source, and does not need to be periodically refreshed. Because memory cards are solid state media, they have no moving parts, and therefore, are unlikely to suffer mechanical difficulties. Earlier removable storage media, such as the PC card, the smart card, and similar cards used for gaming systems can also be considered to be memory cards. However, the newer cards are smaller, require less power, have higher storage capacity, and are portable among a greater number of devices. Memory cards offer a number of advantages over the hard disk drive: they're much smaller and lighter, extremely portable, completely silent, allow more immediate access, and are less prone to mechanical damage.

Communication Bus:

It is a cable, (also called as **System Bus**) which provides parallel data communication path over which information is transferred between the functional areas of the computer, such as CPU, Memory, I/O devices and other computer systems. The information is transferred byte by byte. The CPU logically controls the transfer of data, instructions, and commands between them.

The direction of signal flow for the different buses is indicated on figures in the computer's technical manuals. The direction may be unidirectional or bidirectional depending on the type of bus and type of computer. The name of the bus or its operation usually implies the type of signal it carries or method of operation.

- (i) **Control Bus:** The control bus is used by the CPU to direct and monitor the actions of the other functional areas of the computer. It is used to transmit a variety of individual signals (read, write, interrupt, acknowledge, etc.) necessary to control and coordinate the operations of the computer.
- (ii) **Address Bus:** The address bus contains the connections between the microprocessor and memory that carry the signals relating to the addresses which the CPU is processing at that time, such as the locations that the CPU is reading from or writing to. The width of the bus determines the number of memory locations that can be addressed. For example a system with 64-bit address bus can address 2^{64} memory locations. An address is defined as a label, symbol, or other set of characters used to designate a location or register where information is stored. Before data or instructions can be written into or read from memory by the CPU or I/O sections, an address must be transmitted to memory over the address bus.
- (iii) **Data Bus:** This is used for the exchange of data between the processor, memory and peripherals, and is bi-directional in nature. Sometimes called as the **memory bus**, it handles the transfer of all data and instructions between functional areas of the computer. It transmits data in one direction at a time. The data bus is used to transfer instructions from memory to the CPU for execution. It carries data (operands) to and from the CPU and memory as required by instruction translation. The data bus is also used to transfer data between memory and the I/O section during input/output operations.

The number of wires used in the data bus (sometimes known as the 'width') can differ. Each wire is used for the transfer of signals corresponding to a single bit of binary data. Therefore, a greater width allows greater amounts of data to be transferred at the same time.

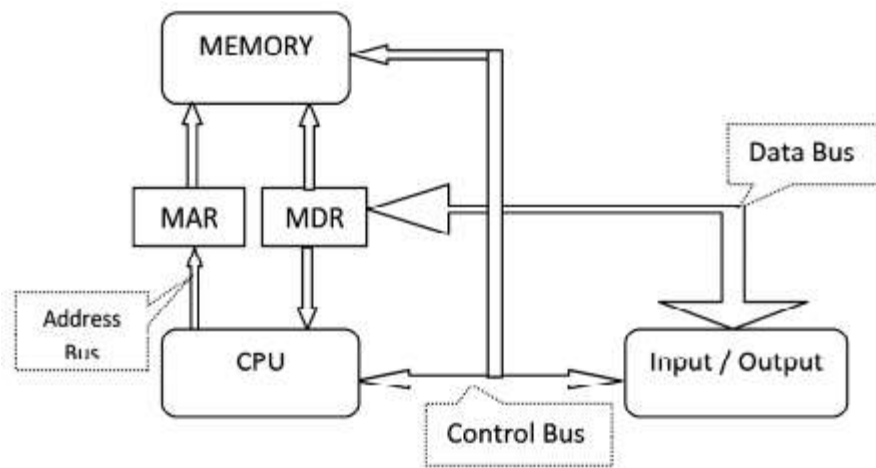


Figure 1.A.13 : System Bus (Address Bus, Control Bus & Data Bus)

Instructions to be executed by a CPU are retrieved from main memory, interpreted by it and executed. CPU is connected to main memory by a set of parallel wires called Address Bus, which carries address bits to the **MAR** (Memory Address Register) and the Data Bus, which carries data/instructions from CPU to **MDR** (Memory Data Register) of main memory. The control bus carries instructions to carry out operations such as Read/Write from or to memory and also input/output operations.

Communication Ports:

A motherboard has a set of connection points called ports to connect units such as disk, VDU, keyboard etc. In a parallel port data-bits are transmitted in parallel (16 or 32 bits simultaneously) to peripherals via a set of parallel wires (called ribbon cables). Serial ports transmit single bits serially, one after another. Faster peripherals such as hard disk are connected to parallel ports. Slower devices such as keyboard are connected to serial port. A standard serial port is known as Universal Serial Bus (USB).

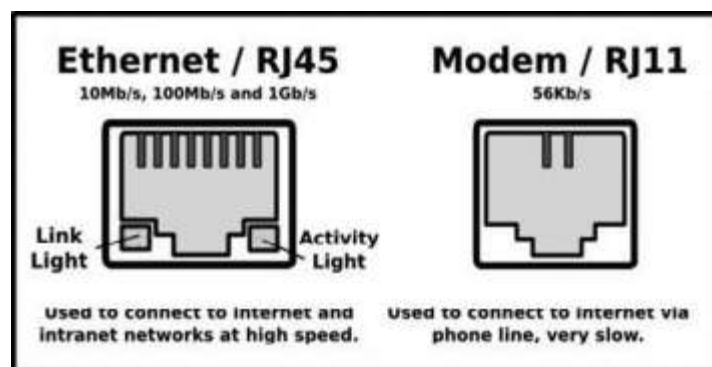
A communication port is mounted in a slot on the computer for easy plugging/unplugging of a peripheral device.

- (i) **Serial ports:** Through this port the information travels in and out one bit at a time. Serial ports come in the form of 9-pin or 25-pin male connector. These ports are often known as communication (COM) port. Mouse, modem etc. are connected using serial port though now mostly they are been replaced by USB port.



Figure 1.A.14 : Serial & Parallel Ports

- (ii) **Parallel Port:** Through this port the several data signals are sent simultaneously over several parallel channels. Parallel ports come in the form of 25 holes and commonly connect the devices to system. Performance capability of these ports is much faster as compared to serial ports and other types of ports. These ports are built in mother board and consist of 25 wires, 8 of those wires are responsible to transfer data and control complete circuit. These ports are used to connect printer, scanner etc.
- (iii) **RJ45 Port:** This port is used for Ethernet connections and can be used between computer and any networked device, such as a cable modem or a network hub. It is an eight wire connector.



- (iv) **RJ11 Port:** This port is for connecting to a telephone line. It has six-wire connector in it and is smaller than RJ45

- (v) **USB Port:** USB stands for Universal Serial Bus, used for short distance digital data communications. This port allows data transfer between devices with little electric power. Flash drives are biggest examples for these ports, which are connected on these ports only. These ports are also helping Smartphone users, as they can now transfer data very easily from computer to their phones.

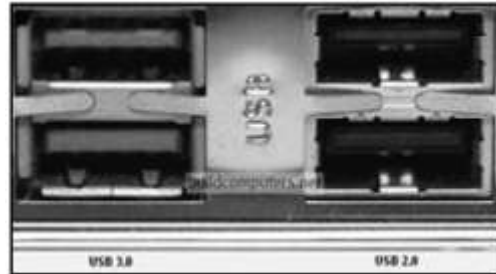


Figure 1.A.15 : USB 3.0 & USB 2.0 Ports

- (vi) **PS/2 Port:** These are most simple and oldest type of ports, and are still working in market with systems. These are 6-pin, low-speed serial connection ports which are used with mouse and keyboards mostly. In system two ports of this kind are found, although they look similar but they are not interchangeable. It is important to be extremely careful to attach the keyboard and mouse to their respective PS/2 port.

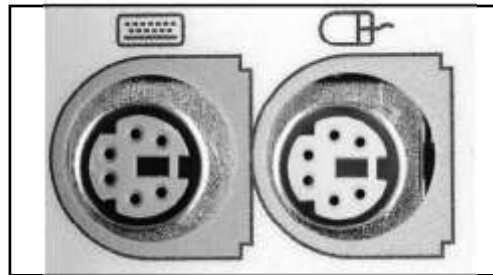


Figure 1.A.16 : PS/2 Port

- (vii) **VGA Monitor Port:** It is used to display Video Graphics Array in a system. This port uses an analog monitor which forward signals towards display adapter. All monitors and LCDs accept these signals, but some flat displays prefer to use digital signal interface. VGA resolution typically refers to the original resolution of 640×480 pixels and 16 colors, but it is not preferred to use for small monitors or displays.



Figure 1.A.17 : VGA Monitor Port

INPUT DEVICES :

The devices which are used to enter data and program instructions in to a computer for storage or processing are called as input devices. Punched Card and paper-tape readers were extensively used for inputting data in earlier systems. Now a days, those have been supplanted by type writer like key boards, interactive graphic devices like mouse, light pen, and joystick and touch screens. Other direct entry devices include optical laser scanners (bar code reader, OCR, OMR), smart cards, voice recognition devices capable of understanding simple verbal commands, digital camera and some auxiliary storage devices like floppy disks, CD-ROMs and Pen drives.

We shall discuss few of these input devices, which are widely used not only for scientific purposes but also for business, industry, education, etc.

(i) Keyboard:

This is the most common input device which uses an arrangement of buttons or keys. Keyboards can have different keys depending on the manufacturer, the operating system, and whether they are attached to a desktop computer or part of a laptop. In a keyboard each press of a key typically corresponds to a single written symbol. However some symbols require pressing and holding several keys simultaneously or in sequence. While most keyboard keys produce letters, numbers or characters; other keys or simultaneous key presses can produce actions (Do/Undo; Copy / Paste, etc.) or computer commands. In normal usage, the keyboards are used to type text and numbers while in a modern computer; the interpretation of key press is generally left to the software. Keyboards are also used for computer gaming, either with regular keyboards or by using keyboards with special gaming features. Apart from the Alphabet Key Pad (Type writer keys), there are several other keys for various purposes such as

1. **Number keys** – There are ten keys (0-9) on each keyboard. Sometimes, there are two sets of these keys; for laptops & others, they are in one line just above the Alphabet Key Pad & in detached key boards, they are arranged in an additional Numeric Key Pads to the right of the Alphabet Key Pad. This is generally found in adding machines or Desktop Calculators.
2. **Direction keys** - There are four arrow keys [left (←), right (→), up (↑) and down (↓)] arranged in an inverted T formation between the

typing keys and numeric keypad. They move the cursor on the screen in small increments along the direction of keys. Unlike alphabet and number keys, these keys do not display anything.

3. **Function keys** - There are generally twelve functions keys *F1-F12*. Application programs and operating systems can assign specific tasks to the function keys. Just like direction keys, these too do not print anything.

Control keys - These keys like, *Home, End, Insert, Delete, Page Up, Page Down, Ctrl, Alt* and *Esc* provide screen control. These keys have special tasks and do not print anything.

4. **Other keys** - There are several other non-printable keys for various different purposes. These include *caps lock, tab, pause, backspace, spacebar, shift; enter* etc which are used for special purposes.



Figure 1.A.18 : Key Board showing different keys

(ii) Mouse:

Interaction with computers was initially restricted to text mode only. The development of Graphical User Interface (GUI), in which a user interacts with the computer programs through graphic icons and menus, demanded an input device, which can be used directly to point and select the icons / menus on the computer screen. A host of such pointing devices were developed subsequently; they are mouse, joystick, trackball etc.

Invented by Douglas Engelbart of Stanford Research Center in 1963, and pioneered by Xerox in the 1970s, the **mouse** is one of the great breakthroughs in computer ergonomics. It is a small object which can roll along a hard, flat

surface. Its name is derived from its shape, which looks a bit like a mouse, its connecting wire that one can imagine to be the mouse's tail, and the fact that one must make it scurry along a surface. It controls the movement of the cursor or pointer on a display screen. A mouse contains at least one button and sometimes as many as three, which have different functions depending on what program is running. The *scroll wheel* of a mouse helps in scrolling through long documents. The applications which run by the clicking of a mouse button are often called *point-and-click* programs. The mouse is also useful for graphics programs that allow you to draw pictures by using the mouse like a pen, pencil, or paintbrush.

There are three basic types of mice:

1. **Mechanical Mouse:** Has a rubber or metal ball on its underside that can roll in all directions. Mechanical sensors within the mouse detect the direction the ball is rolling and move the screen pointer accordingly.
2. **Opto-mechanical Mouse:** Same as a mechanical mouse, but uses optical sensors to detect motion of the ball.
3. **Optical Mouse:** An optical mouse uses a tiny camera to take 1,500 pictures every second. It works on almost any surface. It has a small, red **light-emitting diode** (LED) that bounces light off that surface onto a complimentary metal-oxide semiconductor (CMOS) sensor. The CMOS sensor sends each image to a digital signal processor (DSP) for analysis. The DSP, operating at 18 MIPS (million instructions per second), is able to detect patterns in the images and see how those patterns have moved since the previous image. Based on the change in patterns over a sequence of images, the DSP determines how far the mouse has moved and sends the corresponding coordinates to the computer. The computer moves the cursor on the screen, based on the coordinates received from the



Figure 1.A.19 : Optical Mouse

mouse. This happens hundreds of times each second, making the cursor appear to move very smoothly.

By default, a mouse is configured to work for the right hand. If you are left-handed, the settings can be changed to suit your needs. Laptops are equipped with a small flat surface or sometimes with a very short stick for performing same job as mouse. Using left button of mouse different operations like selection, dragging, moving and pasting can be done. With the right button we can open a context menu for an item, if it is applicable.

Mouse connects to PCs in one of several ways: Serial mouse connects directly to an RS-232C serial port or a PS/2 port. This is the simplest type of connection. A PS/2 mouse connects to a PS/2 port, whereas an USB mouse connects to an USB port. A Cordless mouse uses infrared or radio waves to communicate with the computer.

(iii) Joystick:

A joystick is a personal computer peripheral or general control device consisting of a handheld stick that pivots about one end and transmits its angle in two or three dimensions to a computer. It works on the same principle as a mouse. A joystick is generally configured so that moving the stick left or right signals movement along the X axis, and moving it forward (up) or back (down) signals movement along the Y axis. In joysticks that are configured for three-dimensional movement, twisting the stick left (counter-clockwise) or right (clockwise) signals movement along the Z axis.

Joysticks are often used to control video games, and usually have one or more push-buttons whose state can also be read by the computer. The term joystick has become a synonym for game controllers that can be connected to the computer since the computer defines the input as a “joystick input”.

Apart from controlling games, joysticks are also used for controlling machines such as aircraft, cranes, trucks, powered wheelchairs and some zero turning radius lawn mowers. More recently miniature joysticks have been adopted as navigational devices for smaller electronic equipment such as mobile phones.

Additionally joysticks often have one or more fire buttons, used to trigger some kind of action. These are simple on/off switches.

Joystick elements:

1. Stick
2. Base
3. Trigger
4. Extra buttons
5. Auto fire switch
6. Throttle
7. Hat Switch (POV Hat)
8. Suction Cup

Most I/O interface cards for PCs have a joystick (game control) port. Modern joysticks use a USB interface for connection to the PC.



Figure 1.A.20 : Joystick

(iv) Light pen:

A light pen is an input pointing device shaped like a pen and is connected to a Display Unit (CRT only). The tip of the light pen contains a light-sensitive element (photo-electric cell) which, when placed against the screen, detects the light from the screen enabling the computer to identify the location of the pen on the screen.



Figure 1.A.21: Light Pen

This device is just a plastic cylinder housing a phototransistor, a light-activated switch..When the pen is held to the screen, the beam of electrons which light up the screen, triggers the photo-electric cell, causing it send a signal to the computer that the beam has been detected. Since the computer knew exactly where the electron beam was at that instant, it could figure out the location, where the pen was pointing

The user can point to displayed objects or draw on the screen in a similar way to a touch screen but with greater positional accuracy. It is commonly used to select or otherwise modify text or data on a screen. It is also used to draw pictures or to select menu options.

(v) Graphics Tablet

A graphics tablet is a hardware input device that enables an artist to draw or sketch digitally using a pen or stylus. They are helpful because they provide a more natural and precise feel than a standard computer mouse.

In a graphics tablet, thin sheets of a special film are used in place of the variable resistors. When you press down at a point on the film, a current flows, with the amount of resistance depending on where on the film you press.

Similar to an artist drawing with a pencil and paper, a user draws on the graphics tablet with a stylus. The computer will convert the drawing strokes into digital form, displaying them on the computer screen. The graphics tablet can also be used to capture users' signatures. This use is similar to the signature pads found at many retail stores, where you would write your signature after using a credit/debit card to make a purchase.



Figure 1.A.22 : Graphics Tablet Pen

(vi) Touchscreen

A touch screen sensor is a clear glass panel with a touch responsive surface. The touch sensor/panel is placed over a display screen so that the responsive area of the panel covers the viewable area of the video screen. There are several different **touch sensor technologies** on the market today, each using a different method to detect touch input. The sensor generally has an electrical current or signal going through it and touching the screen causes a voltage or signal change. This voltage change is used to determine the location of the touch to the screen. Some touchscreens also use beams across the screen to create a grid that senses the presence of a finger.

A touchscreen allows the user to interact with a device without a mouse or keyboard and is used with smart phones, tablets, computer screens, Bank



Figure 1.A.23 : Touch Screen of a Tablet PC

ATMs and information kiosks (at Railway stations, Urban Local Body Offices & others).

There are three basic touch sensor technologies those are used to recognize a person's touch; *Resistive*, *Capacitive* & *Surface Acoustic Wave*. Other technologies such as *Infrared*, *Near Field Imaging* are also being implemented

Resistive touchscreens work a bit like “transparent keyboards” overlaid on top of the screen. There's a flexible upper layer of conducting polyester plastic bonded to a rigid lower layer of conducting glass and separated by an insulating membrane. When you press on the screen, you force the polyester to touch the glass and complete a circuit—just like pressing the key on a keyboard. A chip inside the screen figures out the coordinates of the place you touched.

In the **capacitive** system, a layer that stores electrical charge is placed on the glass panel of the monitor. The inner layer conducts electricity and so does the outer layer, so effectively the screen behaves like a capacitor. When a user touches the monitor with his or her finger, some of the charge is transferred to the user, so the charge on the capacitive layer decreases. This decrease is measured in circuits located at each corner of the monitor. The computer calculates, from the relative differences in charge at each corner, exactly where the touch event took place and then relays that information to the touch-screen driver software. One advantage that the capacitive system has over the resistive system is that it transmits almost 90 percent of the light from the monitor, whereas the resistive system only transmits about 75 percent. This gives the capacitive system a much clearer picture than the resistive system.



Figure 1.A.24 : BankATM Touch Screen



Figure 1.A.25 : Touch Screen of a Smartphone

In case of **Surface Acoustic Wave** technology, the screen detects your fingers using sound instead of light. Ultrasonic sound waves are generated at the edges of the screen and reflected back and forth across its surface. When you touch the screen, you interrupt the sound beams and absorb some of their energy. The screen's microchip controller figures out from this, where exactly you touched the screen.

Note : Not all touchscreens act the same, and any of the below actions may react differently on your device.

Tap - A gentle touch or tap of the screen with a finger once to open an app or select an object.

Double-tap - Depending on where you are a double-tap can have different functions. For example, in a browser double-tapping the screen zooms into where you double-tapped. Double-tapping text in a text editor selects a word or section of words.

Touch and hold - Gently touching and holding your finger over an object selects or highlights the object. For example, you could touch and hold an icon and then drag it somewhere else on the screen or highlight text.

Drag - After you have touched and held anything on the screen while keeping the your finger on the screen drag that object or highlight in any direction and then lift your finger off of the screen to drop it in place or stop highlighting.

Swipe - Swiping your finger across the screen scrolls the screen in that direction. For example, pressing the finger at the bottom of the screen and quickly moving it up (swiping) scrolls the screen down.

Pinch - Placing two fingers on the screen and then pinching them together zooms out and pinching your fingers together and then moving them away from each other zooms into where you started.

(vii) Microphone

Sometimes abbreviated as mic, it is an input device that allows the users to input audio into the computers.

Most microphones connect to the computer using the "mic" port on the computer sound card. Higher quality



Figure 1.A.26 : Microphone

microphones or microphones with additional features can connect to the USB port in the computer system.

It is used to convert sound into electrical signal. The analog electrical signal is digitized and fed to the computer; Sound recording software does the necessary editing.

(viii) Scanners

Optical scanning involves, illuminating an object by an incident light source and then converting the reflected light into electrical signal by suitable electronic circuitry(Charge Coupled Device / Photo Multiplier Tube). A scanning device optically scans an image and converts it to a digital image. The digital image is then, suitably manipulated by the application software for the specific use in the computer systems. Since these are direct data entry devices, they require a very good quality of input documents for proper processing. The accurate dimension and ink specification are other two requirements for the purpose. However, image scanners may not require the same restrictions as they do not need any data processing. Scanning devices are used for different purposes as outlined below

- (1) Image Scanner
- (2) OCR scanner
- (3) OMR scanner
- (4) MICR scanner
- (5) Bar Code Scanner

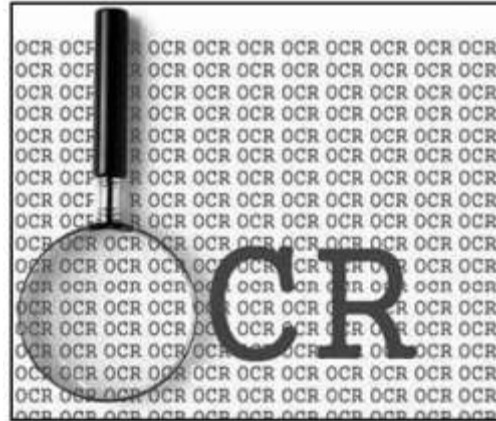
Image Scanner:

An image scanner translates a paper document in to an electronic format for storing in a computer system. It helps in preserving the paper document in electronic form for years without any physical damage. With the help of available software the images can be manipulated in desired manner. There are different types of image scanners-such as, **flat bed** scanner, **sheet fed** scanner, **hand held** scanner and **drum** scanner.

OCR Scanner

Optical Character Recognition (OCR) is the electronic conversion of images of typed, handwritten or printed text into machine-editable text.

The OCR scanner is equipped with character recognition software which converts the images of characters to equivalent ASCII codes (machine encoded text). The scanner first creates the image of the document, and the software (OCR software) translates the array of grid points in to ASCII text, which the computer can interpret as letters, numbers or special characters. The scanned document is stored in the system as a bit-mapped file in TIF format.



It is widely used as a form of data entry from printed paper data records, whether passport documents, invoices, bank statements, computerized receipts, business cards, mail, printouts of static-data, or any suitable documentation. It is a common method of digitizing printed texts so that it can be electronically edited, searched, stored more compactly, displayed on-line, and used in machine processes such as machine translation, text-to-speech, key data and text mining. It is widely used to convert books and documents into electronic files, to computerize a record-keeping system in an office, or to publish the text on a website. OCR is a field of research in pattern recognition, artificial intelligence and computer vision.

OMR Scanner

Optical mark recognition is the process of capturing data by contrasting reflectivity (low reflection: high reflection) at predetermined positions on a page. By shining a beam of light onto the document, the scanner is able to detect a marked area because it reflects less light than an unmarked surface. It is generally distinguished from optical character recognition by the fact that a recognition mechanism is not required. That is, the marks are constructed in such a way that there is little chance of not reading the marks correctly.

One of the most familiar applications of optical mark recognition is the use of HB pencil bubble optical answer sheets in multiple choice question examinations. Students mark their answers, or other information, by darkening circles or ovals or squares on a pre-printed sheet. Afterwards the sheet is automatically graded by a scanning machine.

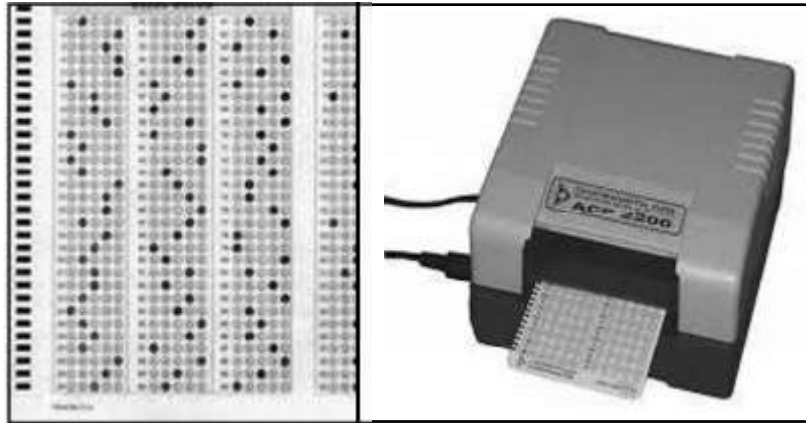


Figure 1.A.27 : OMR Sheet & Scanner

Other examples of OMR are the Magnetic Ink Character Recognition (MICR) & Bar Code Reader. We shall discuss these two very important applications below.

MICR Scanner

Magnetic Ink Character Recognition (MICR) is the common machine language specification for the paper-based payment transfer system. It consists of magnetic ink printed characters of a special design which can be recognized by high speed magnetic recognition equipment. This process excludes any other markings, like human signatures or bank seals, etc. on the cheques.

A series of readable characters (The MICR devices can read only a select group of characters; E-13B and CMC-7 Font) on a cheque provides the receiving party with information needed for processing. These include: the cheque number, bank routing number, cheque account number and in some cases the amount of the cheque. A sample cheque, which employs MICR technology, is shown here.

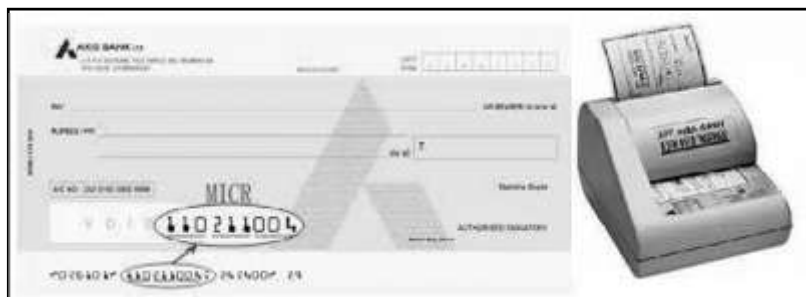


Figure 1.A.28 : A Bank Cheque with MICR Number; MICR Scanner

MICR code consists of 9 numerical digits. First three digits represent the postal code of the city, in which the Bank is situated. For example; for **Bhubaneswar** the PIN Code starts with **751**; so for any Bank having approved MICR facility would have its 1st, 2nd and 3rd digit as **751**. The middle three digits represent the Bank Code allotted by the Reserve Bank of India. For example; the State Bank of India has been allotted the code 002. The last three digits represent the Branch Code of the particular Bank

Bar Code Scanner

When information are encoded in form of parallel lines of varying thickness and spaces between them, they are called as bar codes. These codes are being used for unique identification of different types of goods in retail stores, books, postal packages, tags, etc.

A barcode reader (or barcode scanner) is a computer peripheral for reading barcodes printed on various surfaces. It consists of a light source (a laser beam), a lens and a photo conductor translating optical impulses into electrical ones. In fact a laser beam after striking a pattern of bars reflects in a particular pattern which is sensed by the photo conductor. The electrical pulse so generated goes to a decoder circuitry which analyzes the barcode's image data and sends the barcode's content to the scanner's output port.

(ix) Smart Card Reader

It is used to access the microprocessor of a smart card. There are two broad categories of smart cards - Memory cards and Microprocessor cards. Memory cards contain only non-volatile memory storage components, and some specific security logic. Microprocessor



Figure 1.A.29 : Bar Code Scanner



Figure 1.A.30 : Smart Card Reader

cards contain volatile memory and microprocessor components. The card is made of plastic, generally PVC (Poly Vinyl Chloride). Smart cards are used in large companies and organizations for strong security authentication.

(x) Biometric Sensor

A biometric sensor is an input device that recognizes individuals based on their physical or behavioral traits (fingerprint, voice, face, iris etc.). It converts any of these traits in to electrical signals by measuring the quantities like; pressure, temperature, light, speed, electrical capacity or other kinds of energies. Different technologies can be applied to achieve this conversion using common digital cameras or more sophisticated combinations or networks of sensors.



Figure 1.A.31 : Bio-metric Sensor

Biometric sensor is used to mark attendance of employees/students in organizations/institutions. It is also popular as a security device to provide restricted entry for secured areas.

(xi) Web Camera:

It is a video camera that captures image as data for a computer system and is normally connected directly to a computer through a USB port.



Figure 1.A.32 : Web Camera

OUTPUT DEVICES :

An electromechanical device, which accepts data from a computer and translates them in to a form suitable for the user is called an Output Device. These peripheral devices are classified in to following categories;

- (i) Monitors
- (ii) Printers
- (iii) Plotters

- (iv) Screen Image Projectors
- (v) Voice Response System (Speakers)

Output devices generate output in two formats: Soft copy and Hard copy.

Soft copy: A soft copy is an output, which is not produced on a paper or some material, which can be touched or carried to another place in physical forms. These are temporary in nature and vanish after the use. Examples of soft copy are output on a monitor or sound produced by a voice response system.

Hard copy: It is an output on a paper or any other material which can be touched and carried for being shown to others. These are permanent in nature. Outputs of printer and plotters are hard copy outputs.

1. Computer Monitor :

A computer display monitor, usually called a monitor, is a piece of electrical equipment which displays viewable images generated by a computer without producing a permanent record. The monitor displays instructions we send to the computer and the instruction and results from the computer back to us.

Several different hardware technologies exist for computer monitors, such as: Cathode ray tube (CRT), Liquid crystal display (LCD), Plasma display, Surface-conduction electron-emitter display (SED), Video projector - implemented using LCD/CRT/ other technologies, Organic light-emitting diode (OLED) display.

A monitor comprises the display device, circuitry to generate a picture from electronic signals sent by the computer, and an enclosure or case. A monitor may be within the computer as an integral part (laptops, notebook PC, PDA) or a plugged-in device for desktop or other form of computers.

(i) CRT Monitor:

The CRT or cathode ray tube is the picture tube of a monitor. The back of the tube has a negatively charged cathode. The electron gun shoots electrons down the tube and onto a charged screen. The screen is coated with a pattern of dots



Figure 1.A.33 : CRT Monitor

that glow when struck by the electron stream. Each cluster of three dots (for Red, Green & Blue color), constitute a picture element (pixel).

The image on the monitor screen is usually made up from at least tens of thousands of such tiny dots glowing on command from the computer. The closer together the pixels are, the sharper the image on screen. The distance between pixels on a computer monitor screen is called its **dot pitch** and is measured in millimeters. Most monitors have a dot pitch of 0.28 mm or less.

(ii) LCD Monitors

Liquid crystal display technology works by blocking light. An LCD is made up of two pieces of crossed polarized glass (called as substrates) that contain a liquid crystal material between them. Liquid crystals are liquid chemicals whose molecules can be aligned precisely when subjected to an electric field. When properly aligned, liquid crystals allow light to pass through. A backlight creates light that passes through the first substrate. At the same time, electrical currents cause the liquid crystal molecules to align to allow varying levels of light to pass through to the second substrate and create the colors and images that you see.



Figure 1.A.34 : LCD Monitor

Most LCD displays use active matrix technology. A thin film transistor (TFT) arranges tiny transistors and capacitors in a matrix on the glass of the display. To address a particular pixel, the proper row is switched on, and then a charge is sent down the correct column. Since all of the other rows that the column intersects are turned off, only the capacitor at the designated pixel receives a charge. The capacitor is able to hold the charge until the next refresh cycle.

2. Printers:

Printers are most commonly used output devices, producing hard copies of computer output. They are classified in to two categories-Impact Printers and Non Impact Printers.

- (i) *Impact Printers:* These printers form images by a mechanism of making contact with the paper through an ink-coated ribbon. There are two different ways of forming images on a paper. In one case, there are character shaped print hammers, which strike the paper through the ribbon to produce good quality output. In other case there is a print head containing a row of pins that produce a pattern of dots in the form of characters or other images. Drum Printers, Band / Chain Printer, Dot-Matrix Printer are the Impact Printers.
- (ii) *Non Impact Printers:* Most non impact printers form images from a matrix of dots; but they employ different techniques. The non- impact printers include-Laser Jet Printer, Inkjet Printer, LED printer, Ion deposition printer etc.

Dot-Matrix Printer :

A dot matrix printer or impact matrix printer refers to a type of computer printer with a print head that runs back and forth on the page and prints by impact. A vertical column of hammers strike an ink-soaked cloth ribbon against the paper, much like a typewriter. Unlike a typewriter, letters are drawn out of a dot matrix, and thus, varied fonts and arbitrary graphics can be produced. Because the printing involves mechanical pressure, these printers can create carbon copies and carbonless copies.



Figure 1.A.35 : Dot-matrix Printer

Ink Jet Printer:

Any of a class of non impact printers, which form characters and different kind of images by spraying small drops of ink on to the paper, are called as Ink Jet Printers. The dots are extremely small (the diameter of these dots varies between 50 and 60 microns; the diameter of a human hair is 70 microns!). The dots are positioned very precisely, with resolutions of up to 1440 dots per inch (dpi). The dots can have different colors combined together to create photo-quality images. The print head of an ink jet printer contains up to 64 tiny nozzles,

which can be selectively heated up in a few micro seconds by an IC controlled resistor. When the resistor heats up, the ink near it vaporizes and is ejected through the nozzle on to the paper. This is a character printer, i.e. a selective group of nozzles are heated at a time to print a character on the printer. The selection is made by the computer program. The print head moves horizontally to print a document or image.

Laser Printers:

Any of a class of non impact printers which create electrostatic image with the help of laser beams and imprint them on a paper or a plastic sheet by hot pressing the toner (black or coloured pigment) are called as Laser Printers. A laser printer prints one page at a time. The first laser printer was produced by Xerox. In 1971, researcher Gary Starkweather at Xerox Inc. modified a copier to produce this printer.

The technology of the laser printers has been renovated each passing year. However, some of the basic components and functions remain the same. Few of them are outlined here.

A laser printer consists of the following components:

- (i) A corona wire or a charged roller
- (ii) A photo conducting drum.
- (iii) A laser beam, mirror, lens assembly
- (iv) Toner
- (v) Fuser
- (vi) Rubber blade



Figure 1.A.36 : Laser Printer

Working of a laser printer:

A laser printer works in following six steps:

- (1) *Charging*: A corona wire (in older printers) or a primary charge roller projects a positive electrostatic charge onto the a revolving photosensitive drum or belt which is capable of holding an electrostatic charge on its surface as long as it hasn't been exposed

to certain electromagnetic radiation of a particular wavelength (laser beam)

- (2) *Writing:* The laser beam, being controlled by an image processor circuit is aimed at a moving mirror which directs the laser beam through a system of lenses and mirrors onto the revolving, charged drum. Wherever the laser strikes the photo conducting drum the charge is reversed (becomes negative), thus creating a latent electro-photographic image on the drum surface.
- (3) *Developing:* The surface containing the latent image is exposed to toner. The toner are very fine particles of dry plastic powder mixed with carbon black or coloring agents. The positively charged toner particles get electro statically attracted to the drum surface where the laser wrote the latent image with negative charges.
- (4) *Transferring:* The drum is rolled over the paper (which is given a strong negative charge to attract the toner particles from the drum) to transfer the image on to it.
- (5) *Fusing:* The paper then passes through a fuser assembly. Here a roller provides heat and pressure (up to 200 degrees Celsius) and bonds the plastic powder to the paper by melting.
- (6) *Cleaning:* When the print is complete an electrically neutral rubber blade cleans any excess toner from the photo conducting drum and deposits it into a waste reservoir. A discharge lamp removes the remaining charge from the drum.

Colour Laser Printers:

Essentially, color printers work the same way as monochrome printers, except they go through the entire printing process four times - one pass each for cyan (blue), magenta (red), yellow and black. By combining these four colours of toner in varying proportions, the printer generates the full spectrum of colours (VIBGYOR).

3. Plotters:

The printers discussed above are capable of producing graphics output, but lag in some cases like, engineering design, architecture plan of a building, design of mechanical components of a car etc. These are the cases where high

quality and perfectly proportioned graphics are required on large sheets of paper. A special type of output device used for this purpose is called a Plotter.

Two commonly used plotters are drum plotter and flat bed plotter.



Figure 1.A.37 : Flat-bed Plotter

4. Speakers:

Speakers are output devices used with computer systems to produce sound. They receive audio input from the computer's sound card and produce audio output in the form of sound waves. Most computer speakers are active speakers having an internal amplifier which allows one to increase the volume of the sound. Speakers usually come in pairs, which allow them to produce stereophonic sound from two separate audio channels.



Figure 1.A.38 : Computer Speaker

5. Digital projector:

A digital projector is a device which connects to a computer and is used to project the video output from the computer onto a wall or screen. In classrooms they are often used with electronic whiteboards. The projector can be portable and placed on a stand or it can be permanently fixed to the ceiling. They are used in:



Figure 1.A.39 : Digital Projector

- Classroom teaching
- Staff training sessions
- A presentation to an audience
- Home theaters

Memory Units: Register

A computer system possesses two types of memory units: - Primary/Main memory and Secondary/auxiliary memory; a brief description of these memory units have already been discussed earlier in this chapter. Here we shall discuss some special type of memory units, which reside at the core of a computer processor: the Registers.

A collection of special memory units to hold the information on a temporary basis inside the CPU (not in main memory) constitute a Register. These are electronic memory units and facilitate speedy transfer of information between different parts of a CPU.

The size of a register equals the number of bits required by a CPU to store a character or a number. Thus, the registers are often called as 8-bit, 16-bit, 32-bit or 64-bit registers. The number of bits (or numeral places) a CPU uses to represent numbers is often called “word size” or “bit width” or “data path width” or “integer precision”. The bigger the word size, the faster the computer can process a set of data. Different CPUs possess registers of different sizes and their number also varies from processor to processor. Most recent microprocessors are using 16-bit registers. Each register is assigned to hold a particular type of information. Notable among them are

- (i) **Memory Address Register:** This register holds the address of the memory location, where the current instruction is stored.
- (ii) **Memory Buffer Register:** The contents of the memory word are held in this register. A memory word is either an instruction or a data, which is read from or written to a memory location. An instruction word in MBR gets transferred to the instruction register, whereas a data word goes either to the Accumulator register for processing or gets transferred to the I/O register. Any word to be stored in a memory must go through the MBR.
- (iii) **Program Counter Register:** This register holds the address of the next instruction to be executed. The instructions of program are generally stored in consecutive memory locations, and read and executed sequentially, unless a branching instruction is encountered. In case of branching, the address of the branch instruction gets transferred to PC register to become address of the next instruction.
- (iv) **Accumulator Register:** This register holds the data for any operation. The intermediate results and the final result of a processing are also held in this register. In most of the CPUs there

are more than one AC register. The final result of a processing goes from this register via MBR to the memory.

- (v) **Instruction Register:** The current instruction, which is being executed, is held here. Every instruction has two parts -an operation part and an address part. These parts are separated in this register; the address part is sent to MAR and the operation part is sent to the control unit. In the Control Unit it is decoded and interpreted to generate the command signals specified in the instruction.
- (vi) **Input-Output Register:** All information (instruction and data) coming from an input device or the information before going to any Output device are placed in this register.
- (vii) **Data Register:** It holds the memory operands.

Summary:

- ÷ A computer is a machine, which takes data and instructions as input and performs computations based on those instructions to give some output.
 - ÷ The main functional units of a computer system are - input devices, CPU, memory and output devices.
 - ÷ There are various input and output devices for receiving input and for giving output.
 - ÷ CPU is divided into ALU and CU.
 - ÷ Primary memory iare of two types: RAM and ROM.
 - ÷ Communication bus is used to transfer data between computer and peripherals.
 - ÷ Different ports are used to connect peripherals to a computer.
 - ÷ For future use and permanent storage secondary storage devices are used.
-

EXERCISE**Multiple Choice Questions :**

1. When the electrical power is disrupted or cut off, data and programs are lost in
 - (a) ROM
 - (b) hard disk
 - (c) RAM
 - (d) secondary storage
2. Read Only Memory (ROM)
 - (i) Is volatile
 - (ii) Is programmable
 - (iii) Is mounted on the mother board
 - (iv) Contains the bootstrap loader
 - (a) i and iii
 - (b) i, ii and iv
 - (c) ii, iii and iv
 - (d) iii and iv
3. CD-ROM stands for
 - (a) Compactable Read Only Memory
 - (b) Compact Data Read Only Memory
 - (c) Compactable Disk Read Only Memory
 - (d) Compact Disk Read Only Memory
4. ALU is
 - (a) Arithmetic Logic Unit
 - (b) Array Logic Unit
 - (c) Application Logic Unit
 - (d) None of the above
5. MICR stands for
 - (a) Magnetic Ink Character Reader
 - (b) Magnetic Ink Code Reader
 - (c) Magnetic Ink Cases Reader
 - (d) None of the above
6. Which operation is not performed by computer?

- (a) Inputting
- (c) Controlling

- (b) Processing
- (d) Understanding

7. Pick the one that is used for logical operations or comparisons such as less than equal to or greater than.
(a) Arithmetic and Logic Unit (b) Control Unit
(c) Both (a) & (b) (d) None of these
8. Which of the following is a secondary memory device?
(a) Keyboard (b) Disk
(c) ALU (d) All of these
9. The difference between RAM and Secondary Storage is that RAM is ... and Secondary Storage is ...
(a) Temporary, permanent (b) Permanent, temporary
(c) Slow, fast (d) All of these
10. Which of the following registers holds the memory operands?
(a) Accumulator Register (b) Data Register
(c) Instruction Register (d) Input Register

Short answer type questions :

1. What is the basic function of BIOS Chip?
2. Why is it necessary to use ROM in a computer?
3. What is the function of a bus?
4. Explain the block diagram of a computer organisation.
5. Pair the following equivalent units
10GB, 1GB, 1000MB, 10MB, 104KB, 107KB

Long answer type questions :

1. Describe the Basic Computer Organisation and working of different components.
2. Briefly describe different input devices used in a computer system.
3. Briefly describe different output devices used in a computer system.
4. What is a computer register? Briefly describe the types of registers used in a computer system.
5. Briefly describe different types of ports used in a computer system.
6. What is the function of a Communication Bus? Briefly describe the types of Buses used in a computer system.



CHAPTER - 2

TYPES OF SOFTWARE

1.B TYPES OF SOFTWARE

Introduction

As we all know, a computer is a programmable electronic device. It can carry out mathematical calculations and logical operations. A large amount of data can be stored, processed and retrieved very quickly in a computer system. This machine is composed of hardware components, which are tangible and software (instruction in electronic form) enables it to work properly.

The hardware components include- Central Processing Unit (mother board, microprocessor, electronic-memory, etc.), Storage devices (Hard disk, Solid State memory unit, etc.), Input devices (mouse, keyboard, touch screen, microphone, etc.) and Output / display devices (Monitors, Printers, etc.).

Software is nothing but a set of structured instructions, which can be stored electronically in a computer system and runs its hardware components in a desired pattern. In this chapter we shall discuss in brief, about the software component of a computer system.

An electronic device operates through electric pulses i.e. whether the electric current is flowing through a circuit or not. We denote 'flowing condition' as 1 & 'no flowing condition' as 0. Thus a computer system can understand instructions that consist of only two 'characters' namely 1 and 0. This special language, which is used for an electronic device is popularly known as Binary language or Machine language. The Binary Digits are also known as Bits (Binary Digits). A group of 8 or 16 or 32 or 64 bits is called a Byte, used for storing characters and numbers in computer memory.

It is extremely difficult to learn, remember and use machine language and write proper instructions. So we use some form of electronic interpreter to translator to convert our instructions into machine language and vice versa. The language we use to write instructions for running of a computer system is popularly known as High Level Language. An ordered set of instructions written in High Level Language constitute a program.

A set of programs that governs the operation of an entire computer system and its related devices is known as Software.

Software is not only the basic requirement of a computer system; it makes it more powerful and useful. Software enables the hardware components of a computer system to work in a desired pattern, through which we are now able to make railway/flight reservations, exchange emails, listen music, etc. in addition to basic scientific research and database management. Software can be broadly divided into two categories such as

1. System Software
2. Application Software

1.B.1 SYSTEM SOFTWARE

The functions of all the physical components of a computer system are guided by some instructions or program collectively known as System Software. It controls all internal activities inside a computer system and between all attached components of a computer system. It also controls all internal computer operations as summarized below:

- (i) Reads data and instructions from input devices;
- (ii) Translates all data and instruction into computer understandable form and vice versa;
- (iii) Controls all devices attached to the computer system;
- (iv) Processes and generates the result on the output devices;

System software is a collection of programs that governs the entire hardware operation and supports the application software.

System software helps run the computer hardware and computer system. It is a large amount of software that is produced by the computer manufacturer or software enterpriser. Some common examples of System Software are:

- BIOS
- Operating System
- Device Drivers
- Language Processors

BIOS (Basic Input / Output System)

Commonly known as the System BIOS, it is basically a booting firmware (small program that controls various electronic devices attached to the main computer system). It is designed to be the first set of instructions run by a Computer when powered on. The initial function of the BIOS is to initialize system devices such as the RAM, hard disk, CD/DVD drive, video display card, and other hardware components. The BIOS sets the machine hardware into a known state to help the operating system to configure the hardware components. This process is known as booting, or booting up. BIOS programs are stored on a chip (BIOS chip) which is fitted in to the motherboard of a computer system.

[Boot firmware is the ROM-based software that controls a computer from the time that it is turned on until the primary operating system has taken control of the machine. The main function of boot firmware is to initialize the hardware and then to “boot” (load and execute) the primary operating system. Secondary functions include testing the hardware, managing hardware configuration information, and providing tools for debugging in case of faulty hardware or software.]



Figure 1.B.1 : BIOS ROM chip

Operating System

It is a collection of programs, which provides an interface between the user and computer system (hardware). Just like traffic police it manages every operation of a computer system. It is stored on the hard disk or any other external storage device. It is the first program to be executed on a computer after the BIOS. Every computer must have an operating system to operate all its components and run other programs. Operating system is a set of system programs that controls and coordinates the operations of a computer system.

The core part of the operating system, the **kernel**, stays loaded into main memory all the time. The operating system performs important tasks like receiving input, sending information to output device, keeping track of files and directories in the memory, as well as controlling the various peripheral units such as printers, monitors, keyboards etc. Following picture describes the overview of the functions of an Operating System.

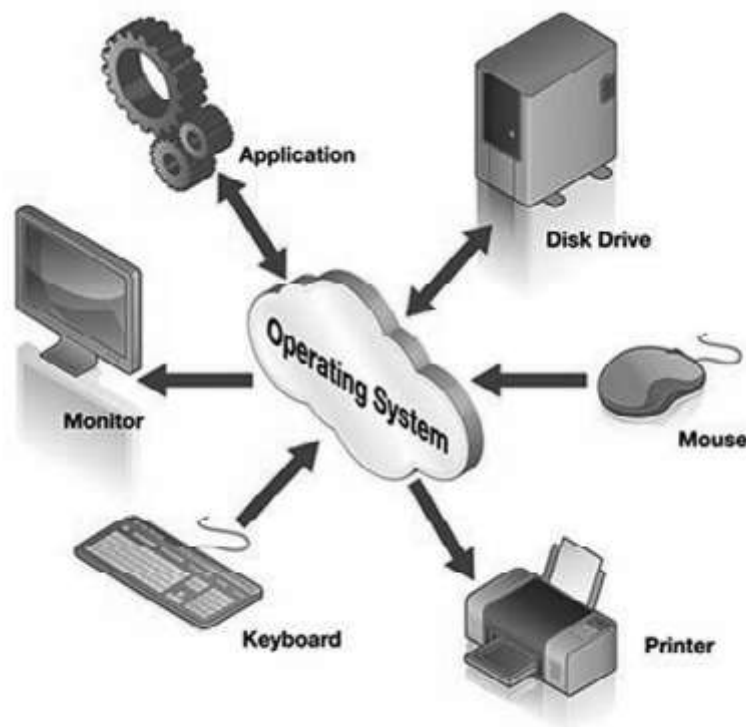


Figure 1.B.2 : Diagram showing control of I/O devices by an of an Operating System

Need for an Operating System

Operating system provides a software platform, on top of which, other programs, like Application programs are run. The application programs must be written to run on the environment of a particular operating system. Our choice of operating system, therefore, depends to a great extent on the CPU and the other attached devices and the applications we want to run. Some of the most popular operating systems are DOS, Microsoft Windows, Linux, Mac OS, Solaris, JBoss, etc.

All operating systems are designed to perform the following basic operations:

1. Send information to the computer through the keyboard
2. Display result on the monitor
3. Send information to a printer
4. Read / write files on to secondary storage.

Major Functions of an Operating System:

An operating system is responsible for the following functions:-

1. Communicate with hardware and the attached devices [Device Manager]
2. Manage different types of memories [Memory Manager]

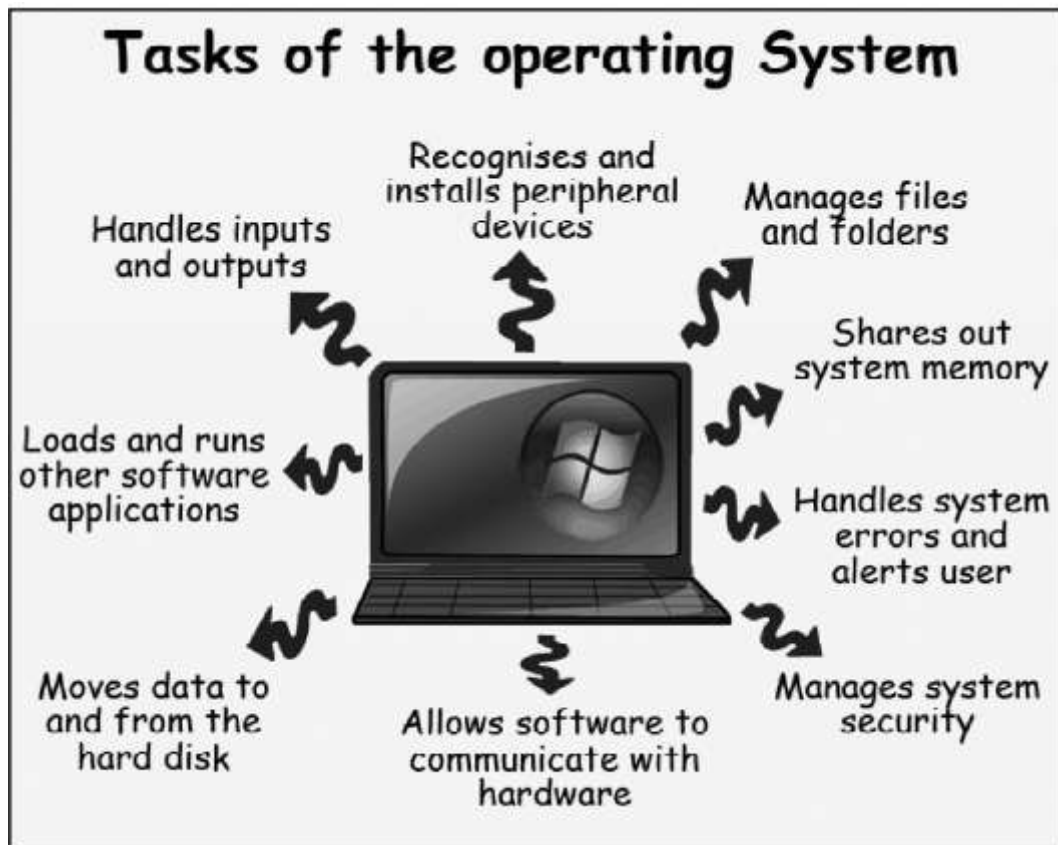


Figure 1.B.3 : Functions of an OS

3. Provide a user interface [Interface Manager]
4. Provide a structure for accessing an application [Program Manager]
5. Enable users to manipulate programs and data [Task Manager]
6. Manage the files, folders and directory systems on a computer [File Manager]
7. Provide basic networking structure for LAN and Internet [Network Manager]
8. Provide a minimal security to the computer system through authorization (user name) and authentications (password) [Security manager]

(i) Device Management

The computer system that we are using might be attached to peripheral device like keyboard, printer, modem, and mouse etc. All these devices communicate with the computer for getting the job done, which is monitored by the operating system.

(ii) Memory Management

Memory is the term usually referred to the main memory of a computer, which is random access in nature. When jobs are running, operating system retrieves it from the disk and loads it into its memory. But, before it is loaded, the OS must check whether memory is available for it or not, and if available it allocates it to the specific job. Once the execution of the job is over, it gets removed from the memory and this free memory is allocated to another job.

(iii) User Interface

There must be a way to interact with the system from outside world because as you know that computer understands the language 0's and 1's. The gap between the machine understandable language and human understandable language is bridged with the help of OS command-interpreter. The user interface helps in bridging the gap between languages [Machine and Human].

(iv) CPU Management

Operating system handles the problem of assigning various jobs for processing. This function of the operating system is also called job scheduling. CPU gets the job from the keyboard or storage medium puts it into memory,

processes the job and passes the result onto the predetermined place and makes itself ready for the next job.

(v) Disk Management

The operating system helps you to store and retrieve files from disks whenever, required by the user. It helps in partitioning and formatting the storage disks. The activities in connection with disk management includes; Free space management, Storage allocation and Disk scheduling. The speed of operation of a computer depends on the disk (hard disk, secondary disk) space management.

Types of Operating System

Different types of operating systems are available and used depending upon the primary purpose and application and the type of hardware attached to the computer; they are:

1. Single User Operating System.
2. Multi User Operating System
3. Batch Operating System
4. Networking Operating System
5. Distributed Operating System

Further an Operating Systems may be

- (i) Multi-programming
- (ii) Multi-tasking
- (iii) Multi-threading
- (iv) Multi-processing
- (v) Time-sharing
- (vi) Real Time / On-Line

1. Single User Operating System

This operating system allows one user to operate a computer and can run different programs; but only one program can be supported and executed by it at any point of time. The CPU executes only one program at a time. Microsoft's Disk Operating System [DOS] is an example of single user operating system.

2. Multi-User Operating System

A multi-user operating system allows more than one user to access a single computer system at the same time. This operating system is used to be installed in a network of computers (created by attaching a number of terminals with a Server or Host computer) to be operated by more than one user at a time. Unix, Linux, Windows are common examples of multi-user operating system.

3. Batch Operating System

This is the earliest operating system (also called Control Program) is a collection of commonly required utility programs stored in the main memory. Batch processing environment requires grouping of similar jobs, consisting of programs, data and system commands.

This OS uses a large computation time with no need of user interaction / involvement. Some examples of such programs include Payroll, Forecasting, Statistical analysis and large Scientific calculations. Users are not required to wait while the job is being processed. They can submit their programs to operators and return later to collect them. Essentially it provides a non-Interactive environment; it allows little or no interaction between user and executing programs. Users have no control over intermediate results of a program. This type of arrangement does not create flexibility in software development.

4. Networking Operating System

A networking operating system is a collection of software and associated protocols that allow a set of autonomous computers interconnected by a computer network to be used together in a convenient and cost-effective manner. In a networking operating system, the users are aware of existence of multiple computers and can log in to a remote machine from another machine.

A network operating system makes all the resources available to the users; even if they are attached to remote hosts. Shared files are available on the Central Computer. Different computers can store information in different formats. Network operating system supports communication between the users and system. It offers many capabilities, which includes authorized access to various resources of the network host, up-to-the minute network documentation on-line, etc. Windows NT is an example of networking operating system

5. Distributed Operating System

A distributed operating system is one that looks to its user like an ordinary centralized operating system but runs on multiple independent CPUs. The key concept here is transparency. In other words, the use of multiple processors should be invisible to the user. Another way of expressing the same idea is to say that user views the system as virtual uni-processor but not as a collection of distinct machines. Distributed operating system, users are not aware of where their programs are being run or where their files are residing; they should all be handled automatically and efficiently by the operating system. An example of distributed operating system is Novell

Multiprogramming is a rudimentary form of parallel processing in which a number of programs belonging to different users are kept simultaneously in distinct area in the main memory. Main memory of the computer should be large enough to accommodate several programming. Multiprogramming operating system is an attempt to increase the utilization of CPU. This technique reduces the idle time of the CPU. Several programs are resident of main memory but the CPU executes one programming at a given time. Multi programming approach has some distinct **disadvantages**: Supposing there is a small error in the first program, and there is no error in the next, which occupies the CPU for a long time. The CPU takes the first program, detects the error and sends it out for correction. The monitor sends the next program to the CPU. Since the second program engages the CPU completely, it is executed without any interruption. Therefore even after the mistake in the first program has been corrected, the user of that program must for the CPU time. This draw back has been overcome by the introduction of the concept of time-sharing operating system.

Multitasking, in an operating system, is allowing a user to perform more than one computer task (such as the operation of an application program) concurrently. The operating system is able to keep track of where you are in these tasks and go from one to the other without losing information. Microsoft Windows 2000, IBM's OS/390, and Linux are examples of operating systems that can do multitasking (almost all of today's operating systems can). When you open your Web browser and then open 'Word' at the same time, you are causing the operating system to do multitasking. Being able to do multitasking doesn't mean that an unlimited number of tasks can be juggled at the same time. Each task consumes system storage and other resources. As more tasks are started, the system may slow down or begin to run out of shared storage.

Multithreading is the ability of a program or an operating system process to manage its use by more than one user at a time and to even manage multiple requests by the same user without having to have multiple copies of the programming running in the computer. Each user request for a program or system service (and here a user can also be another program) is kept track of, as a thread with a separate identity. As programs work on behalf of the initial request for that thread (path of a particular job) and are interrupted by other requests, the status of work on behalf of that thread is kept track of until the work is completed.

Multiprocessing is the coordinated processing of programs by more than one computer processor. Multiprocessing is a general term that can mean the dynamic assignment of a programs to one of two or more computers working in tandem or can involve multiple computers working on the same program at the same time (in parallel). In this type of system there is more than one processor and they all are running parallel to each other. They are also known as parallel system or tightly coupled system. All processor have common bus, clock, memory and input-output devices. User interaction when the process is in operation is very restricted. In this an identical copy of the operating system is running parallel to each other.

Multiprocessing should not be confused with multiprogramming or the interleaved execution of two or more programs by a processor. Today, the term is rarely used since all but the most specialized computer operating systems support multiprogramming. Multiprocessing can also be confused with multitasking, the management of programs and the system services they request as tasks that can be interleaved, and with multithreading, the management of multiple execution paths through the computer or of multiple users sharing the same copy of a program.

In a **time-sharing** operating system, many video terminals may work simultaneously with a CPU. In this a single CPU executes multiple jobs simultaneously by switching among them. If there are 50 terminals, then 50 users can operate the computer simultaneously. In this the operating system allows more than one user to share the computer system. So the CPU time is divided among the number of user in a small time slot or time-slice- period or time quantum. The switching process between the job is so fast that the user can not feel there is a time gap and realize as if the system is dedicated to him/her only and feel number of processes are running concurrently. The jobs are executed by the CPU in a Round Robin Fashion.

On-line or Real-time operating system

To get immediate response from the computer for any request of the user, it is possible to share large volume of data in disc storage so that the data can be retrieved fast. To retrieve the data available in the disc immediately, system software has been designed and this software is called an on-line operating system. For example, suppose in a computerized bank, the accounts of the all the customers are stored in a disc. Using the on-line operating system, the account of any customer can be retrieved immediately from the disc and transaction done without any delay. The on-line operating system is used in the computerized reservation counter of railways and airlines. In Real time system the user has the capacity to set the priority to perform a particular task over others. The system responds to input instantly. The OS has the time rigidity on the operation of a process. This is mainly used for scientific experiments, robotics, satellite launching, weapon systems, automatic engine, fuel injection systems etc. LYNX and Windows CE are examples of real time operating systems.

Sometimes operating systems are also categorized as:

- (a) PC Operating System
 - (b) Server Operating System.
 - (c) Mainframe Operating System.
 - (d) Hand held Operating System.
 - (e) Mobile Operating System.
- (a) **Personal Computer Operating System:** The concept of micro computer was introduced during mid 1970's. A microprocessor with VDU and a detachable keyboard constitute a micro computer. It is also referred as personal computer. That is the computer which is dedicated to a single user at a time is usually referred as personal computer. The cost of hardware items decreased and the capacity of storing and speed increases. DOS is the most popular operating system used by these computers. With the increase in technology the another operating system named UNIX came into existence around 1980 which includes the facilities of a single user as well as the multiuser operating system. Now days to connect personal work station in a large personal computer these operating system are widely in use such as WINDOWS-NT, IBM OS/2 and the MACINTOSH operating system. They full fill requirement of a single user and if require can be shared by the other users having LAN facilities. For example,

computers used in universities, large business organizations, companies etc. Windows, Linux, MacOS, ChromeOS, Ubuntu, etc are some Operating Systems for PCs and Laptops.

- (b) **Server Operating System:** In case of LAN generally an individual computer is not directly connected with other. They communicate with each other by a central computer called a Server. The server provides the network facilities, the security, traffic routing from one network to another network in case of parallel computers. To communicate with different LAN differ in their topology we use bridges, gateways and routers. The operating system, which enables to perform the above, mentioned tasks are called server-operating system. These operating systems are also used for engineering design, order processing, and payroll calculations in companies, continued existence etc. These applications are called mission critical application. This is a well-secured multitasking, multithreading and open in nature. The commonly used server operating systems are WINDOWS NT, OS/2, and UNIX etc.
- (c) **Mainframe Operating System:** With the increase in use of mainframe and minicomputer, the computer manufacturer has developed their own operating system, known as mainframe operating system. In addition to the operating system they have their own applications that ran top of them. These are called proprietary systems because they are all owned by a single proprietor. These operating systems have better processing facilities and security. The entire task required by a modern operating system can be performed by using this operating system.
- (d) **Hand held Operating System:** Windows-CE (similar to Windiws-95) having a small keyboard and a touch sensitive screen which is miniature used for hand held computer, Pamtop computers and the Operating system which enables them to work with is called hand held operating system.
- (e) **Mobile Operating System:** A mobile operating system, also called a *mobile OS*, is an operating system that is specifically designed to run on mobile devices such as mobile phones, smart phones, PDAs, tablet computers and other handheld devices.

A mobile OS typically starts up when a device powers on, presenting a screen with icons or tiles that present information and provide application access. Mobile operating systems also manage cellular and wireless network connectivity, as well as phone access. The operating system is

responsible for determining the functions and features available on your device, such as thumbwheel, keyboards, wireless application protocol (WAP), synchronization with applications, email, text messaging and more.

Examples of mobile device operating systems include Apple's **iOS**, Google's **Android**, Research in Motion's **BlackBerryOS**, Nokia's **SymbianOS**, Hewlett-Packard's **webOS** (formerly Palm OS), Microsoft's **Windows Phone OS**, Samsung's **BADA** and MWC (Mobile World Congress)'s **MeeGo**. Microsoft's **Windows 8**, functions both as a traditional desktop OS and a mobile operating system. **FirefoxOS** and **Ubuntu for mobiles** are new entrants in this segment.

Device Driver

A device driver is system software that that operates or controls a particular type of device that is attached to a computer. It provides a software interface to hardware devices, enabling operating systems and other computer programs to access hardware functions without needing to know precise details of the hardware being used.

A driver communicates with the device through the computer bus or communications subsystem to which the hardware connects. When a calling program invokes a routine in the driver, the driver issues commands to the device; once the device sends data back to the driver, it invoke routines in the original calling program. Drivers are hardware dependent and operating-system-specific. They usually provide the interrupt handling required for any necessary asynchronous time-dependent hardware interface.

Device drivers simplify programming by acting as translator between a hardware device and the applications or operating systems that use it. Programmers can write the higher-level application code independently of whatever specific hardware the end-user is using.

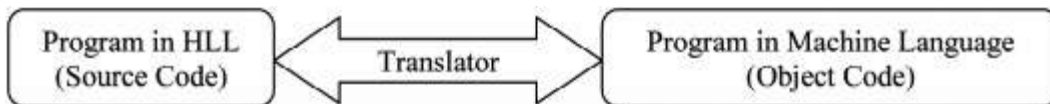
All computer accessories like Printer, Scanner, Web Camera, etc come with their own driver software. These driver softwares help the operating system and other application software to communicate with those devices for optimal use.

Language Processor

A computer system understands the instructions only in binary language (also called as machine language) or known as Low Level Language (LLL).

This language is extremely difficult to learn, remember and implement by a programmer. Thus there was a need for some special language that is easy to learn and understand for the programmer in order to interact with the computer system. These special languages or set of commands are collectively known as High Level languages (HLL). Basic, C, C++, JAVA, etc are some examples of High Level Programming Languages. When we write a program in high level language and feed it to a computer it cannot comprehend the codes, unless they are converted to binary forms (machine level language).

The system software, which converts a program written in HLL to Machine Language and vice versa, is called as Language Processor or **Translator**.



Basically, there are three types of translators

- (i) Assembler
- (ii) Compiler
- (iii) Interpreter.

Assembler - A programming language which is very close to the low level language is called as Assembly language. This language consists of mnemonic codes, is difficult to learn and is machine dependent. A language processor, which translates a program written in assembly language into machine language is called an Assembler.

Compiler - A compiler is a language processor which converts (or translates) the **entire program** written in high level language into machine language in one go. If it fails to convert the program because of error(s) present in the program, all errors are reported together along with the line numbers for debugging. After all the errors are removed, the program can be recompiled to obtain the object program. After the compilation process is completed, the object program can directly be executed, without the intervention of the compiler thus saving memory.

Interpreter - This language processor converts a high level language program into machine language **line by line** as well as executes it. If there is any error in the program, translation and execution stops and the error is reported for debugging. The conversion and execution resumes only after that

error is rectified. This is a slow process and consumes high memory as the interpreter is also required to execute the correct code along with reporting the errors, but is very useful for debugging and thus suitable for novice programmer. Debugging is the process of removing all errors from a computer program.

Difference between compiler and interpreter:

- Compiler saves the machine codes permanently for future reference, whereas the machine codes produced by interpreter are not saved.
- A compiler is a bigger program as compared to an interpreter. Thus the former occupies more memory space than the later.
- Though it occupies more memory space, a compiler is much faster than an interpreter.
- A compiler system, includes a (built in or separate) **linker**, generates a stand-alone machine code program, while an interpreter performs only the actions described by the source program.

Linker: In high level languages, some built in header files or libraries are stored. These libraries are predefined and these contain basic functions which are essential for executing a program. These functions are linked to the libraries by a program called *Linker*. If linker does not find a function in the library, then it informs to compiler and then compiler generates an error. The compiler automatically invokes the linker as the last step in compiling a program. Besides built in libraries, it also links the user defined functions to the user defined libraries. Usually a longer program is divided into smaller subprograms called modules. And these modules must be combined to execute the program. The process of combining the modules is done by the linker.

Loader: Loader is a program that loads machine codes of a program into the system memory. In computing, a loader is the part of an Operating System that is responsible for loading programs. It is one of the essential stages in the process of starting a program. Because it places programs into memory and prepares them for execution. Loading a program involves reading the contents of executable file into memory. Once loading is complete, the operating system starts the program by passing control to the loaded program code. All operating systems that support program loading have loaders. In many operating systems the loader is permanently resident in memory.

Example : Apex Language Processor, developed by Centre for Development of Advanced Computing (CDAC), is a multilingual word processor running under

DOS and UNIX that allows typing of all Indian scripts through the common INSCRIPT keyboard overlay.

Application Software :

Application software is a set of programs to carry out a specific task like word processor, spreadsheet, presentation tools, library management system, railway reservation, antivirus software, etc. Generally application software can perform only one specific job and cannot be used for something else. For instance, library management software cannot be used for railway reservation system or word processing software is generally not used as a spreadsheet. Application Software can be divided into different categories depending upon their uses as follows:

1. Utility Software
2. General Purpose Application Software
3. Specific Purpose Application Software
4. Developer Tools

Utility Software :

After all the basic and necessary software like Operating System and Device Drivers have been installed, we also require some additional software to keep our computer system efficient and trouble free. Generally these software come bundled with the Operating System Software but we can also use utility software provided by other vendors. Few examples of utility software are as follows:

- (i) Compression
- (ii) Back Up
- (iii) Disk De-fragmentation
- (iv) Anti-virus
- (v) Text Editor

Compression software: Using this software, one can reduce (compress) the storage size of any computer program/file while not in use. This utility comes in handy when we want to transfer a big program or computer file from one computer to another either through internet or using storage devices like Pen Drive, CD or DVD.

Backup software: Though computer is in general a dependable device but it is always advisable to take regular back up of important data and programs stored in the computer. In case of any damage to the system, the back-up files can be restored and the important data can be recovered from the back-up files. This utility software facilitates you to take regular back-up of important files and folders stored in a drive into another storage device like a Pen drive or CD or a DVD or another computer. This backup data can be restored in case of any unforeseen situation.

Disk de-fragmentation software: When computer system finds a file too large to store in a single location, it splits the file and stores it in pieces (called fragments), which are logically linked. This simply means that different parts of the file are scattered across the hard drive in noncontiguous locations. This type of fragmented file requires some extra time to access and slows down the system. Disk de-fragmentation utility software speeds up the system by rearranging such fragmented files stored on a disk in contiguous locations in order to optimize the system performance. Defragmenting a disk minimizes head movement, which reduces the time it takes to read files from and write files to the disk. It also reduces system start up times.

Following figure shows three fragmented files named AK (stored in 6 fragments), RK (stored in 4 fragments) and DP (stored in 5 fragments), then running the defragmentation utility reorganize the file contents in consecutive locations.

AK	AK	RK	RK	<i>free</i>
<i>free</i>	DP	DP	DP	<i>free</i>
<i>free</i>	RK	AK	AK	AK
AK	<i>free</i>	<i>free</i>	DP	<i>free</i>
<i>free</i>	RK	<i>free</i>	DP	<i>free</i>

Figure 1.B.4 : Fragmented Disk

AK	AK	AK	AK	AK
AK	DP	DP	DP	DP
DP	RK	RK	RK	RK
<i>free</i>	<i>free</i>	<i>free</i>	<i>free</i>	<i>free</i>
<i>free</i>	<i>free</i>	<i>free</i>	<i>free</i>	<i>free</i>

Figure 1.B.5 : De-fragmented Disk

Antivirus software: A computer virus is a computer program intended to hamper the performance of a computer system. These viruses are copied into the system through some other infected programs (copied into the system) or

downloaded from the internet. This utility software provides the user with a virus free work environment by restricting the entry of any unwanted program into the system.

Text Editor: This utility software helps one to create, store or edit a basic text files. A text file generally stores English type text and can also store numeric and special characters with little formatting. Popular examples of text editors are Notepad, Notepad2, Notepad++, Gedit and KWrite.v

A text editor (like notepad) is where you simply make some quick changes to the text. Remember that you cannot do much formatting with text editors. Whichever font or formatting style you use in the document stays throughout the document; you cannot change fonts or formats within the document. A word processor (like MS Word), on the other hand has a variety of options to format text within the document like inserting special symbols, changing colors, line spacing, tables and a whole lot of other things that you cannot do with a normal text editor.

General Purpose Application Software

Software required for running and maintaining the basic components of a computer system come under the category of system software whereas software required to solve some specific task of daily use is generally called application software. An operating system is an example of system software while documentation tool, a presentation tool, a spreadsheet tool are all examples of application software. Even your favorite computer game is an example of application software.

Being an electronic device, a computer as such cannot perform anything of its own. Some of the application softwares are designed for general day to day applications and uses. They are often termed as general purpose Application Softwares. They may be classified as;

- (i) Word Processor software
- (ii) Spreadsheet software
- (iii) Presentation software
- (iv) DBMS software
- (v) Integrated Development Environment

Word Processor: Word Processor is a general purpose application software that facilitates the creation of text documents with extensive formatting.

The user can not only create a document and add lines into it but can also use different types of fonts of various sizes along with features like underlining or making a certain part of the text bold. One can also add clipart and other graphics into the document. Therefore we can use word processing software for various tasks from writing a simple document to designing special art effect.

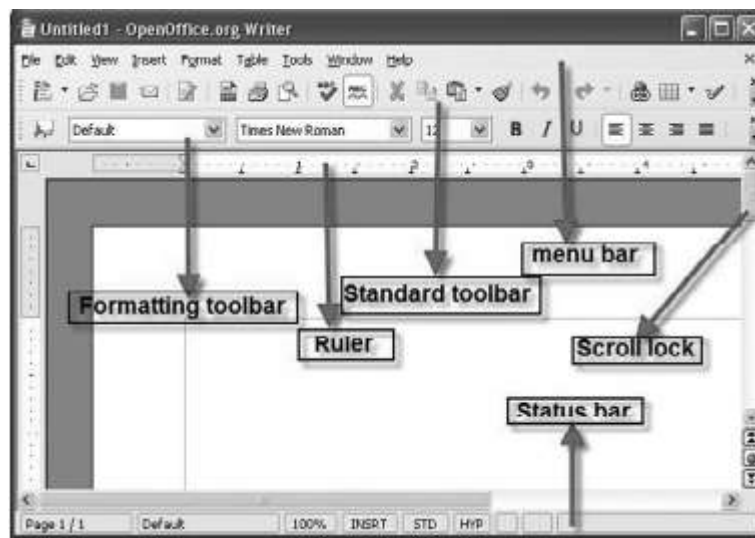


Figure 1.B.6 : Open Office's "Writer" window

Preparing a common letter for different addressee (using mail merge feature), writing stories, applications and designing posters (using clip art and graphics) are some of the common applications of a word processor. Popular examples of Word processing software are Microsoft Word and **Writer** (Open Office).

Spreadsheet Tools: Spreadsheet Tool is a general purpose application software that facilitates creation of tabular forms where some text and numerical values can be stored. A spreadsheet tool not only allows one to create a document and add data into it but also allows creation of different types of charts and graphs based upon the numerical data stored in a worksheet. Furthermore, all common mathematical and statistical formulae can be used on the stored numeric data and various text functions can be applied on the texts stored in the cells of a worksheet. Popular examples of Spreadsheet tools are Microsoft Excel and Calc (Open Office). A spreadsheet tool can be used by a class teacher to maintain the marks scored by different students. This enables the teacher to statistically

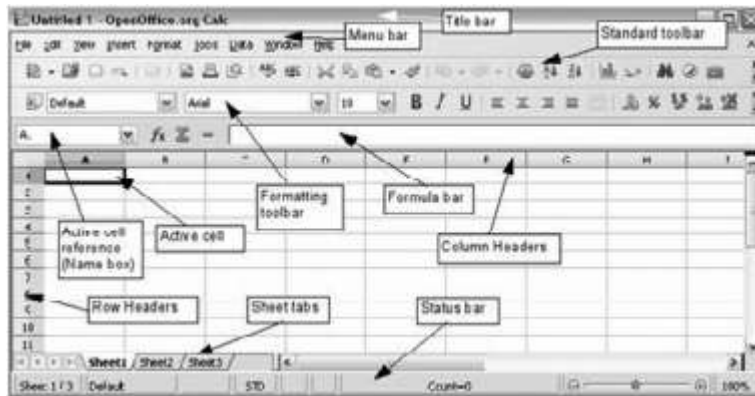


Figure 1.B.7 : Open Office’s spread sheet tool “Calc” window

analyze the performance of students both individually and collectively. Similarly spreadsheet is used by almost all professionals to maintain and statistically analyze data.

Presentation Tools: Presentation Tool is general purpose application software that facilitates the creation of presentations on any particular topic like Uses of Internet, Global Warming, Social networking or any topic for class room teaching or of social interest and importance. It allows one to not only create a presentation and add slides into that but also allows use of various formatting features like adding different types of background, different fonts, animations, audio, video, clipart and other graphics. Popular examples of Presentation tools are Microsoft Power Point and Impress (Open Office).



Figure 1.B.8 : Open Office’s presentation tool “Impress” window

Database Management System: It is an application software that facilitates creation of computer programs that control the creation, maintenance, and the use of database for an organization and its end users by generating desired Reports. It allows the user to not only store data but also control the addition, deletion, management, and retrieval of data in a database. It also allows importing and exporting the data to many formats including Excel, Outlook, ASCII, dBase, FoxPro, Oracle, SQL Server, ODBC, etc. Popular examples of Database Management System are **Base** (Open Office) and Microsoft Access.

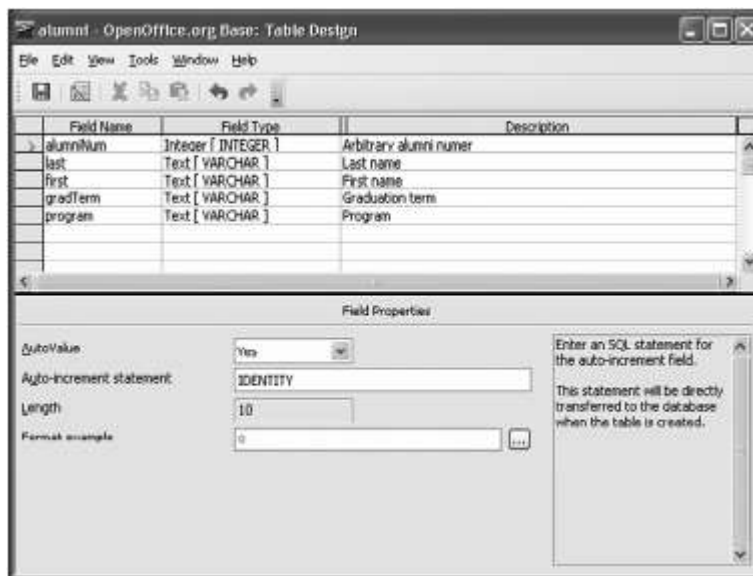


Figure 1.B.9 : Open Office's database tool "Base" window

Integrated Development Environment

When a programmer starts the process of writing a program to develop software for any type of application, he/she requires a series of software developing tools like code editor, debugger and compiler. A platform where all these software developing tools are bundled into a package is known as Integrated Development Environment (IDE). An Integrated Development Environment (IDE) is an application program that consists of all required software developing tools for developing software as part of a single interface. It typically consists of the following tools:

1. Source Code Editor
2. Graphical User Interface (GUI) builder
3. Compiler / Interpreter
4. Debugger
5. Build Automation tool

Example: Netbeans IDE, Visual Basic IDE

Specific Purpose Application Software

There are some application softwares, which are tailor made to suit the specific needs of the government or private-run institutions, corporate offices, business establishments, etc. come under the category of specific purpose application software. The usage of some specific purpose application softwares are explained below:

Inventory Management System: Inventory management software is used for tracking inventory levels, orders, sales and deliveries. This software is made up of several key components, all working together to create a cohesive inventory for many organizations' systems. It can also be used in the manufacturing industry to create a work order, bill of materials and other production-related documents. Companies use inventory management software to avoid product overstock and outages. It is a tool for organizing inventory data that was previously stored in hard-copy form or in spreadsheets. Companies often use inventory management software to reduce their carrying costs, track products and parts as they are transported from a vendor to a warehouse, between warehouses, and finally to a retail location or directly to a customer.

It is generally used in departmental stores or in an institution to keep the record of the stock of all the physical resources. For example, a school keeps record of the number of computers, printers, printing sheet, printer cartridge available in the school's computer department. Maintaining this kind of data also helps the administration to place purchase order when the current stocks of consumables like printing sheet or printer cartridge is less than the critical limit.

Human Resource Management System: Human Resources Management System (HRMS) refers to a system in which the human resource (the attributes of persons in relation to an organization like the Bio-Data; CCR tracking, Payroll, achievement, history of posting, etc.) is managed through the

use of Information Technology. Some basic modules, those are essential components of this software are discussed below.

- **Payroll module** automates the pay process by gathering data on employee time and attendance, calculating various deductions and taxes, and generating periodic pay cheques and employee tax reports.
- **Time and Attendance module** provides labor distribution capabilities and data analysis features. Cost analysis and efficiency metrics are the primary functions.
- **Benefits Administration module** provides a system for organizations to administer and track employee participation in benefits programs like, insurance, compensation, profit sharing and retirement.
- **HR management module** is a component covering many other HR aspects from application to retirement. The system records basic demographic and address data, selection, training and development, capabilities and skills management, compensation planning records and other related activities. Human resource management function involves the recruitment, placement, evaluation, compensation and development of the employees of an organization. Initially, businesses used computer-based information systems to:
 - **Online recruiting** has become one of the primary methods employed by HR departments to garner potential candidates for available positions within an organization. Talent management systems involves:
 - ☞ Analyzing personnel usage within an organization;
 - ☞ Identifying potential applicants;
 - ☞ Recruiting through company-facing listings / through online recruiting.
 - **Applicant Tracking System** maintains an organized recruitment effort, cross-posting within and across general or industry-specific job boards and maintaining a competitive exposure of availabilities.
 - **Training module** provides a system for organizations to administer and track employee training and development efforts as training courses, books, CDs, web based learning or materials to develop their skills.

- **Employee self-service module** allows employees to query HR related data and perform some HR transactions over the system. Employees may query their attendance record from the system without asking the information from HR personnel.

Many organizations have gone beyond the traditional functions and developed human resource management information systems, which support recruitment, selection; hiring, job placement, performance appraisals, employee benefit analysis, health, safety and security, while others integrate an outsourced applicant tracking system that encompasses a subset of the above.

Payroll Management System: Payroll Management System software is used by almost all modern organizations to encompass every employee of the organization who receives a regular salary / wage or other compensation. All types of payment methods are calculated by the payroll software and it ensures the electronic transfer of funds to the employees' bank accounts, Provident Fund Accounts, Insurance funds, etc.

Financial Accounting System: Financial accounting is a specialized branch of accounting that keeps track of a company's financial transactions. Using standardized guidelines, the transactions are recorded, summarized, and presented in a financial report or financial statement such as an income statement or a balance sheet. Accounting software describes a type of application software that records and processes accounting transactions within some functional modules such as accounts payable/receivable, payroll, etc. It functions as an accounting information system and can be developed in-house by the organization or purchased from a third party, or may be a combination of a third-party application software package with local modifications. Accounting software may be on-line based, or may be desktop based. It varies greatly in its complexity and cost. Financial Accounting System generates the following general-purpose, external, financial statements:

1. Income statement (sometimes referred to as "profit and loss [P&L] statement")
2. Balance sheet (sometimes referred to as "statement of financial position")
3. Statement of cash flows (sometimes referred to as "cash flow statement")
4. Statement of stockholders' equity

Some of the accounting packages are: *Fresh Books, Sage, QuickBooks, Zoho, KashFlow, PayChex Online, and Kashoo*

Financial Management System: It is a methodology and software that an organization uses to oversee and govern its income, expenses, and assets with the objectives of maximizing profits and ensuring sustainability. An effective financial management system improves short- and long-term business performance by streamlining invoicing and bill collection, eliminating accounting errors, minimizing record-keeping redundancy, ensuring compliance with tax and accounting regulations, helping personnel to quantify budget planning, and offering flexibility and expandability to accommodate change and growth. Other significant features of a good financial management system include:

- (i) Keeping all payments and receivables transparent.
- (ii) Amortizing prepaid expenses.
- (iii) Depreciating assets according to accepted schedules.
- (iv) Keeping track of liabilities.
- (v) Coordinating income statements, expense statements, and balance sheets.
- (vi) Balancing multiple bank accounts.
- (vii) Ensuring data integrity and security.
- (viii) Keeping all records up to date.
- (ix) Maintaining a complete and accurate audit trail.
- (x) Minimizing overall paperwork.

Reservation System: Commonly seen at railway reservation offices, this software helps the concerned department to automatically check the availability of the seats or berths of any train for any particular date very quickly. Now a days using the internet and this software one can book or reserve tickets of any train / flight / bus for any dates with in no time.

Summary

- ☞ Software is a set of programs that governs the operation of a computer system and its related devices.
- ☞ Software can be broadly divided into two categories - System Software & Application Software.

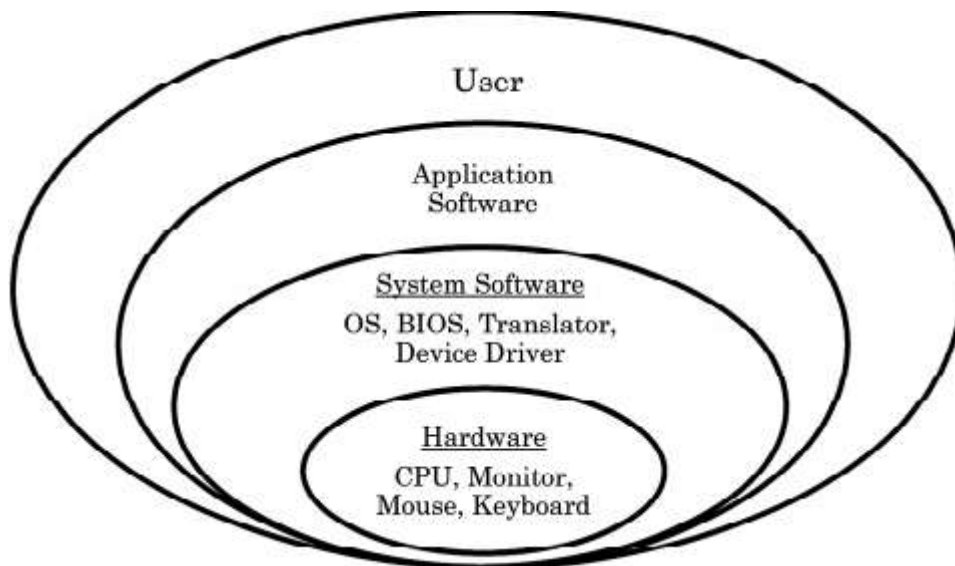


Figure 1.B.10 : Diagram showing the relationship between hardware and the different types of softwares

- ☞ System Software controls all internal activities of a computer system and between all attached components of a computer system.
- ☞ BIOS-The basic input/output system is the built-in software that contains the code required to control the keyboard, monitor, disk drives, communications ports, and other functions independently of the computer operating system.
- ☞ Operating system is set of system programs that control and coordinate the operations of a computer system. It is the interface that acts like a bridge between a user and the hardware components of a computer.
- ☞ Major functions of an operating system are Device Manager, Memory Manager, Interface Manager, Program Manager, Task Manager, File Manager, Network Manager, Security Manager.
- ☞ Different Types of Operating system include Single User, Multi-user, Multiprocessing, Multitasking, Multithreading and Real time.
- ☞ A device driver is system software that acts like an interface between the Device and the user or the Operating System.

- Application software is a set of programs to carry out a specific task like word processor, spreadsheet, presentation tools, library management software, railway reservation etc.
 - Utility Software are used to keep your computer system efficient and trouble free.
 - Word processor is general purpose application software that facilitates creation and formatting of text documents.
 - Presentation tools are general purpose application software that facilitate creation of presentations on any particular topic.
 - Spreadsheet is general purpose application software that facilitates creation of worksheets that stores text and numerical data in tabular form. Performing basic statistical analysis including graphs is the main utility of this software.
 - Debugging is the process of removing all errors from a program.
 - An interpreter converts as well as executes a high level language program into machine language line by line.
 - A compiler is a language processor which converts (or translates) the entire program written in high level language into machine language in one go.
 - An Integrated Development Environment (IDE) is a platform or an application program that consists of all required software developing tools required for developing an application at one place. The various tools are arranged as separate windows integrated into one big environment.
-

EXERCISE**Multiple Choice Questions :**

1. Interpreter converts a HLL program into machine language by converting and executing:
 - (a) In one go
 - (b) line by line
 - (c) Group of Five lines
 - (d) None of these
2. Which of the following Operating System is suitable for handling more than one user at one time?
 - (a) Multiprocessing OS
 - (b) Multiuser OS
 - (c) Multiprogram OS
 - (d) Multiprogramming OS
3. 'Calculator' is a
 - (a) Package
 - (b) Utility Software
 - (c) Customized software
 - (d) Developer Tool
4. A bit stands for
 - (a) Boolean Digit
 - (b) Binary Digit
 - (c) Binary Decimal
 - (d) Byte Digit
5. Which of the following is not an application software?
 - (a) Word Processor
 - (b) Anti Virus Program
 - (c) Operating System
 - (d) Railway Reservation System
6. Which of the following is not an utility software?
 - (a) Text Editor
 - (b) Anti Virus Program
 - (c) Compression software
 - (d) Railway Reservation System
7. Which of the following is not a component of Open Office?
 - (a) Word
 - (b) Impress
 - (c) Calc
 - (d) Writer
8. A program written in high level language is known as:
 - (a) Source Program
 - (b) Main Program
 - (c) Object Program
 - (d) Image Program

9. Which of the following is a Language Processor?
- (a) Text Editor (b) BackUp Software
(c) Both (a) & (b) (d) None of these
8. Which of the following is a Special Purpose Application Software?
- (a) Word Processor (b) Presentation Tool
(c) Pay Roll module (d) Build Automation Tool

Short Answer type Questions :

1. What do you mean by a bit and a byte?
2. Define the term Software.
3. Explain the relationship between hardware and software with the help of a suitable example.
4. Define the term System Software.
5. What do you mean by BIOS?
6. Write down the four major functions of an operating system.
7. Differentiate between the following:
(a) Single user & Multiuser OS
(b) Compiler & Interpreter
8. Write short note:
(a) Utility Software b) Device Driver
9. What do you mean by presentation software?
10. What are the uses of Payroll Management System?
11. Explain different types of language processors.
12. Explain the different types of Operating System.
13. Define the term Application software.
14. Explain the term Integrated Development Environment.

Long Answer type Questions :

1. What is a System Software? Describe four basic types of this software and their functions.
2. What is an Operating System? Discuss different types of OS presently being used in personal computers.
3. What is a Language Processor? Give brief descriptions of different types of LPs.
4. What is a Utility Software? Discuss various types of these softwares.
5. Discuss various types of Specific Purpose Application softwares.



CHAPTER - 3

OPEN SOURCE CONCEPTS

1.C OPEN SOURCE CONCEPTS

Introduction:

In the previous sections we have discussed different types of softwares- System software and Application software. These softwares are nothing but a collection of programs (written in high level programming languages) to carry out specific jobs. The instructions written for a program (or per se a Software package) are called as Source Code.

Source Codes are generally written by a single programmer or a group. In case the license is owned by a vendor company to sell the package at a price, is called Proprietary software; the software which is free is said to be an 'Open Source Software'.

There are competing advantages and disadvantages of both types of softwares. Proprietary softwares come with built-in-facilities, which are handy for new users; for any glitches, the user has no other way but to report to the vendor, who in turn would debug the code for necessary repair. But the user of an Open Source package can debug the code and fix it instantly. Another advantage of Open Source over the Proprietary software is that, it is developed, used and guarded by millions of users, which makes it versatile, hack-proof and user-friendly (a user can modify its source code for any specific purpose, which is not possible in case of proprietary ones). In this section, we shall discuss some commonly used Open Source Softwares.

Open Source Software (OSS)

Open Source Software is a package available with source code, open for any required change/edit/re-distribution, without any specific authorization

(password). It allows free re-distribution of the software without royalties or licensing fees to the authors. The source code can be distributed with the software or otherwise made available for no more than the cost of distribution. Further, it allows anyone to modify the software or derive other software from it, and to redistribute the modified software under the same terms.

Examples of Open Source Software

1. *Office Automation software* : Open Office, KOffice, and Neo Office.
2. *Operating System* : Ubuntu, Fedora, Linux, Android
3. *Web Browser* : Mozilla Firefox
4. *Web Design* : NVU, Bluefish,
5. *E-Commerce* : Zen Cart, Virtue Mart
6. *Content Management Systems* : Joomla, Drupal
7. *Internet related Software* : Mozilla Foundation (Firefox and Thunderbird)
8. *Programming Related Software*: PHP, Perl, Python, MySQL, JAVA, Netbean IDE etc.

Examples of Proprietary Software

1. *Office Automation software*: - Microsoft Office Suite (MS-Office).
2. *Operating System* : Windows, iOS, MacOS
3. *Web Browser*: Internet Explorer, Chrome
4. *Audio & Video playback*: Window's Media Player
5. *Programming Related Software*: Visual Basic, C, Visual Basic IDE etc.

Criteria of OSS:

The distribution terms of open-source software must comply with the following criteria

1. **Free Redistribution:** The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

2. **Source Code:** The source code must be the preferred form in which a programmer would allow modify and distribute the program.
3. **Derived Works:** The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.
4. **Integrity of the Author's Source Code:** Sometimes the license of OSS may restrict source-code to be distributed after modification. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.
5. **No Discrimination against Persons or Groups:** The license must not discriminate against any person or group of persons.
6. **No Discrimination against Fields of Endeavour:** The license must not restrict anyone from making use of the program in a specific field of Endeavour. For example, it may not restrict the program from being used in a business, or from being used for scientific research.
7. **Distribution of License:** The rights attached to the program must apply to all, to whom the program is redistributed without the need for execution of an additional license by those parties.
8. **License must not be Specific to a Product:** The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.
9. **License must not Restrict Other Software:** The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.
10. **License must be Technology-Neutral:** No provision of the license may be predicated on any individual technology or style of interface.

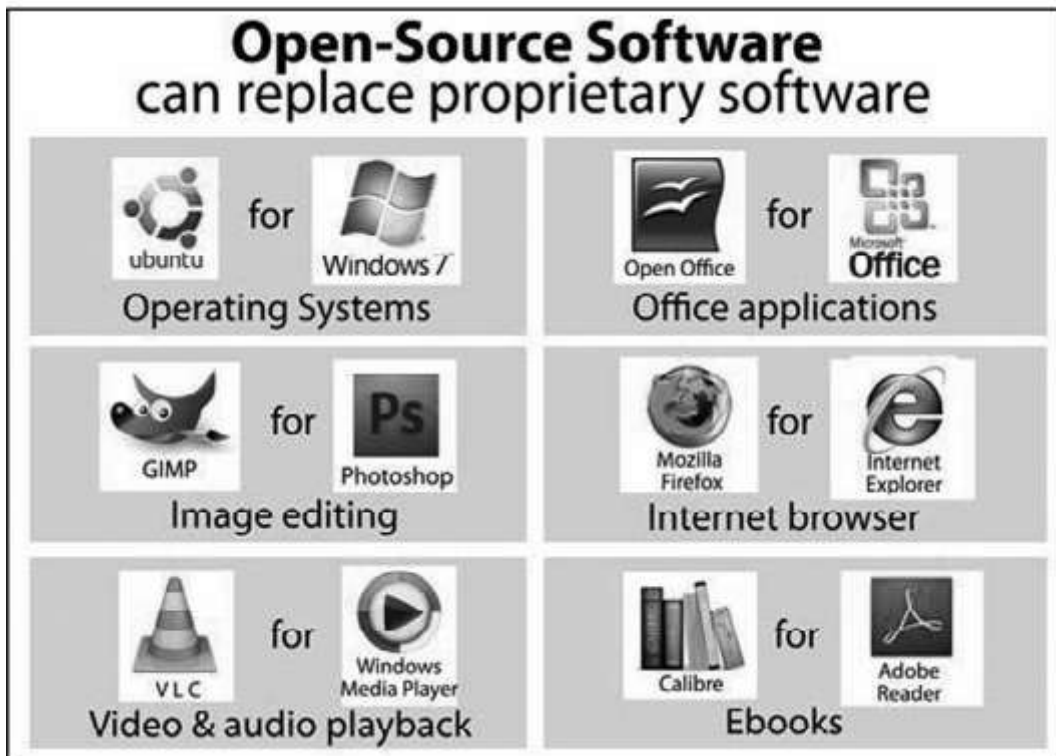


Figure 1.C.1 : Open Source Software as against the Proprietary softwares

Open Source Initiative (OSI): The OSI is a nonprofit corporation with a mission of education and advocacy for the open source community. Anybody can call their program open source, but this organization maintains compliance and trademark information that the industry at large recognizes in a cooperative effort. In other words, everyone plays nicely with the help of the OSI.

[**Freeware** is a type of software available free of cost and allows copying and further distribution but does not allow modification as its source code is not available; **Shareware** is available for redistribution for stipulated time but after some time some license fee is required to be paid.]

FOSS / FLOSS: FOSS stands for *Free Open Source Software* and FLOSS stands for *Free Libre Open Source Software*. The word libre or livre means freedom. These softwares are free as well as open. That is any one is freely licensed to use, copy, study and change the software in anyway and source code is openly shared, so that the developer or programmer are encouraged to voluntarily improved the design of the software.

FLOSS is aimed to avoid talking, whether it is a 'Free Software' or an 'Open Source Software'. Few of these softwares are outlined below.

1. **GNU:** It is an Open Source-operating system developed by Richard M. Stallman in 1983. Unlike Linux, it supports the GNU Hurd kernel and is intended to be a 'Complete UNIX Compatible Software System'. GNU's design is Unix-like but differs from UNIX by being free software and containing no Unix code. Programs released under the auspices of the GNU project are called GNU packages or GNU programs. Now, it offers a wide range of softwares, including Applications apart from Operating System.
2. **Linux:** It is a popular Open Source-operating system originally written by Linus Torvalds, a student of Finnish University in 1991. It is a part of popular web service program LAMP (Linux, Apache, MySQL, and PHP). Linux can be downloaded from www.linux.org. As an operating system, its aim is to make it inter-operable (work together) with other operating systems and establish computing standards.
3. **Mozilla** is popular free and open source software. It was developed to create internet suites like, web browser, e-mail client, HTML editor, IRC (internet relay chat) client, etc. Development of Mozilla was initiated by Netscape Communication Corporation, but now a separate Mozilla foundation is looking after it and is responsible for its development and maintenance. It is one of the most freely downloaded software on the Net (www.mozilla.org).
4. **Open Office:** It is an office application suite made to be in competition with Microsoft Office. Open Office is based on the source code of older version of 'Star Office', which was acquired and made open source by Sun Microsystems. Open Office includes the following components
 - (a) Writer (Word processor)
 - (b) Calc (Spreadsheet)
 - (c) Draw (Graphic software)
 - (d) Impress (Presentation program)
 - (e) Base (Database program)
 - (f) Math (Editor for mathematical formula), etc.

It can be integrated with databases, such as MySQL and Postgre SQL by configuring it. Open Office can be downloaded from www.openoffice.org.

5. **Java:** It is a programming language originally developed by James Gosling at Sun Microsystem in 1991. The language was first called Oak but later renamed as Java; it was released as a free and open source software product of Sun Microsystems. The language derives much of its syntax from C and C++ but it has fewer low-level facilities than either of them. Java is a general-purpose, concurrent, class based, object oriented language that is specifically designed to have a few implementation dependencies as possible. It is intended to let application developers. “Write Once, Run Anywhere” (WORA), meaning the code that runs on one platform does not need to be recompiled to run on another. Java can be downloaded from www.java.com.
6. **NetBeans:** NetBeans is an Integrated Development Environment (IDE) for developing applications in programming languages like, Java, PHP, C/C++ and HTML. It is also an application platform framework for Java Desktop Applications and others. NetBeans IDE written in Java can run on platforms like, Windows, OSx, Linux, Solaris, etc compatible with Java Virtual Machine.

This platform offers reusable services common to desktop applications, allowing developers to focus on the logic, specific to their application. NetBeans can be downloaded from <https://netbeans.org>.
7. **MySQL:** MySQL is perhaps the most used open source Relational Database Management System (RDBMS) that runs as a server providing multi-user access to a number of databases. MySQL development project has made its source code available under the terms of the GNU general public license, as well as under a variety of proprietary agreements. It supports many languages, such as C, C++, PHP, Ruby, Java, Python, etc. It can work on many platforms, such as MS-Windows, Linux, Mac OS, etc. It is used in many high profiles, large scale World Wide Web products, including Google, Facebook, Twitter, and YouTube. It can be easily downloaded from www.mysql.com.

Common Open Standards

The open standard refers to internationally accepted technical standards. These standards are freely available and do not have any restrictions. The file or information created using these standards are accessible, irrespective of any change in technology. The data and information in open standards are accessible across the platform and applications. The open standards follow the principles like;

- (i) *Availability*: Open Standards are available for all to read and implement.
- (ii) *Maximise End-User Choice*: It creates a fair, competitive market for implementations.
- (iii) *No Royalty*: Open standards are free for all to implement, with no royalty or fee.
- (iv) *No Discrimination*: OpenStandards-Administrators do not favor one implementer over another for any reason other than the technical standard-compliance.
- (v) *Predatory Practices*: Open standards may employ license terms that protect against subversion of the standard by embrace and extend tactics.

Features of Open Standards

1. Available free of cost.
2. Ensures data is application and platform independent.
3. No hidden information.
4. Diversity and interoperability in the industry, i.e. it enables business and people to go for any technology of their choice as per their needs and budget.
5. Offers diverse choices for users. e.g. ASCII characters, HTML file, Joint Photographic Expert Group (JPEG), Portable Network Graphic, etc.
6. Makes the data accessible to all, there is no restriction for using particular software.

Examples of Common Open Standards

1. **World Wide Web (www)** is an open source information space where documents and other web resources are identified by URL (Uniform Resource Locator), interlinked by hypertext links, and accessed via the internet. By using a web browser, one can view web pages that may contain text, images, videos and other multimedia and navigate between them *via* hyperlinks. The Internet is a global system of interconnected computer networks.

2. **HTML** (Hyper Text Markup Language) is the standard language for designing a web page. The HTML is a platform-independent format (can be used with any operating system). A web browser reads an HTML documents and compose them into visible / audible web pages. The browser does not display the HTML tags but uses the tags to interpret the content of the page. HTML allows images and objects to be embedded and can be used to create interaction forms. HTML elements form the building blocks of all websites.
3. **DHTML** (Dynamic HTML) is a collective term for a combination of Hypertext Markup Language tags and options that can make Web pages more animated and interactive than previous versions of HTML. Much of dynamic HTML is specified in HTML 4.0. Simple examples of dynamic HTML capabilities include having the color of a text heading change when a user passes a mouse over it and allowing a user to “drag and drop” an image to another place on a Web page. Dynamic HTML can allow web documents to look and act like desktop applications or multimedia productions. A feature called **dynamic fonts** lets web page designers include font files containing specific font styles, sizes, and colors as part of a Web page and to have the fonts downloaded with the page. That is, the font choice no longer is dependent on what the user’s browser provides. Dynamic HTML includes the capability to specify style sheets in a “cascading style sheet” fashion (that is, linking to or specifying different style sheets or style statements with predefined levels of precedence within the same or a set of related pages).
4. **XML** (eXtensible Markup Language) is a markup language that defines a set of rules for encoding documents in a format that is both human readable and machine readable.

Many Application Programming Interfaces (APIs) have been developed to aid software developers with processing XML data and several schema systems exist to aid in the definition of XML based languages. The ODF (Open Document Format) is an XML based file format for representing electronic documents. These documents could be text document, spreadsheet, etc. It was developed with the aim of providing a universal document format that could be used with any office software suite.

This standard was developed by a technical committee in the Organisation for the Advancement of Structured Information Standards (OASIS) consortium. The specifications were originally developed by Sun Microsystem as an XML format for OpenOffice.org Office Suite.

5. **ODF (Open Document Format):** The Open Document Format (ODF) is an open source standard for office documents (text, spreadsheets, presentations etc.). ODF is the default file format for applications like Open Office 2.0, Office 1.5, Star Office, IBM Workplace and many others. A document in ODF can be represented either as a single XML document or a collection of sub documents in a package (i.e. Zip) file.

Most commonly used file extensions with ODF documents are as follows:

- **.odt** for word processing (text) documents, e.g. file.odt
 - **.ods** for spreadsheet files, e.g. file.ods
 - **.odp** for presentation files, e.g. file.odp
 - **.odg** for graphics files, e.g. file.odg
 - **.odb** for database files, e.g. file.odb
6. **TCP/IP:** The TCP/IP suite of networking standards provides the foundation for the network infrastructure of the Internet. It is a two-layered packet-switching specification in which data to be communicated between two end-points on a network is first broken up into smaller data packets that are then individually routed through the network from the source to the destination points. The higher layer, Transmission Control Protocol (TCP), manages the disassembling of the data into smaller packets at the source and the reassembling at the destination point upon receipt of the data packets. The lower layer, Internet Protocol (IP), handles the addressing and routing of each packet so that it gets to the correct destination.

Indian Language Computing

Indian Constitution identifies 22 languages, of which 6 languages (Hindi, Telugu, Tamil, Bengali, Marathi and Gujarati) are spoken by at least 50 million people within the boundaries of the country. Indian language computing refers to the ability of interaction in different Indian languages on electronic systems. But, it is most complicated, given the fact that each major Indian language has its own script and grammar.

To fructify efforts of making use of ICT (Information and Communication Technology) reach to poor and rural people of India, Technology Development for Indian Languages (TDIL) Programme initiated by the Department of Electronics & Information Technology (DeitY), Ministry of Communication & Information Technology (MC&IT), Govt. of India has the objective of developing Information Processing Tools and Techniques to facilitate human-machine

interaction without language barrier; creating and accessing multilingual knowledge resources; and integrating them to develop innovative user products and services.

The Programme also promotes Language Technology standardization through active participation in International and national standardization bodies such as ISO, UNICODE, World-wide-Web consortium (W3C) and BIS (Bureau of Indian Standards) to ensure adequate representation of Indian languages in existing and future language technology standards.

Character Coding

In order to use the computer, the computer should be able to recognise the alphabets, numerals, punctuation marks and other special characters. The method of assigning a unique number for all the characters and symbols used in writing scripts is termed as **Character Coding**. The most widely used alphanumeric code is **ASCII** (American Standard Code for Information Interchange), which is a 7 bit code, consisting of $2^7 = 128$ possible code groups. The ASCII code is used for transferring alpha numeric information between a computer and its input/output devices. It is widely used in most microcomputers, minicomputers and in many mainframes.

Unicode (Universal Code, developed by Unicode Consortium in 1991) defines the standards for representation and processing of written text to support the worldwide interchange, processing, and display of the written texts of the diverse languages and technical disciplines of the modern world irrespective of the operating platform or programming language. The current version of the Unicode standard, defines more than 107,000 characters covering around 90 languages. It uses UTF-8 encoding system. Character encoding standards not only identify each character and its numeric value, or code point, but also how this value is represented in bits.

Unicode has incorporated Indian scripts under the group of Asian scripts. These include, Devnagari, Bengali, Gurmukhi, Gujarati, Kannada, Malayalam, Odia, Tamil, Arabic and Telugu. [*A unicode chart of Odia characters is given at the end of this chapter for reference*]

The Unicode Standard has been adopted by many leading industries such as Apple, HP, IBM, Microsoft, Oracle and many others. Though Unicode files are very large (it takes 2 bytes to store each character), it possesses some distinct advantages as;

- Enables a single software product or a website across the multiple platforms, languages and countries without re-engineering.

- Allows transportation of data through different systems without corruption.
- Uses 16-bit system which can support many more characters.

Department of Electronics and Information Technology (DeitY), Ministry of Communications & IT, is the voting member of the Unicode Consortium.

ISCII (Indian Script Code for Information Interchange)

The ISCII was adopted by the Bureau of Indian Standards in 1991 for computing in Indian languages. It is an 8 bit code capable of handling 2^8 i.e. 256 characters. The lower 128 characters are the 7-bit ASCII Character set and the top 128 characters cater to all the ten Indian scripts based on the ancient Brahmi script. ISCII is used in all GIST (Geographic Information Support Team) products. It is also used by many other vendors for developing products and solutions in Indian language. The ISCII can be applied for all Indian language scripts including Devnagari, Gurumukhi, Gujarati, Telugu, Tamil, Odia and Kannada.

The 8-bit ISCII code retains the standard ASCII code; while the Indian script keyboard overlay is designed for the Standard English can co-exist with Indian scripts. This approach also makes it feasible to use Indian scripts along with existing English computers and software, so long as 8-bit character codes are allowed.

The ISCII code (Indian Script Code for Information Interchange), co-exists with the standard English code: It contains only the basic alphabet arranged in an order which conforms to that in most of the Indian dictionaries.

Fonts

Font refers to the character, which is displayable and printable with a specific **style** and **size**. In fact, a font is a collection of **glyphs** (shape, style and size of the character) used for visual depiction of characters on some display device. On the basis of technical specification fonts are divided into two types: - True Type Fonts (TTF) and Open Type Fonts (OTF).

- (i) **True Type Fonts:** These are widely used with Windows and MAC operating system. Basically, the True Type Fonts are developed by Apple Incorporation and they are licensed to Microsoft. In True Type Fonts, all information for previewing and printing is in **one file**. It makes font management easy. But it is not compatible with cross platform, i.e. different files are required for platforms like Windows and MAC TTFs are normally 8 bit fonts.

- (ii) **Open Type Fonts:** It is the extended version of True Type Font format. These are 16 bit fonts. It allows handling of large glyph sets using Unicode encoding and contains up to 65536 different glyphs, so we can have multiple languages within one font. The data and the glyph set in an open type font file are in tabular form. This table also contains information, how the final glyphs are formed.

Advantages of OTF

- Provides better quality of output.
- Manages font easily by the use of one file.
- Compatible with cross platform, i.e. same files can work on both (i.e. Windows and MAC) platforms.
- Read file names easily.

Disadvantage of OTF

- All applications do not support the extended features.

On the basis of Font Configuration, fonts are categorised into two types:

- (i) **Static Fonts:** In static fonts, after designing and digitising, characters are stored in font files. They will appear all times in identical shapes. Almost every font such as Times New Roman, Courier, Arial, etc. belongs to this category.
- (ii) **Dynamic Fonts:** The characters in dynamic fonts are redefined every time, they are displayed or printed. They are generated every time with slight variation. The hand written fonts such as hand written alphabets or calligraphic letters are dynamic fonts because of they generate individual variations.

Entering Indian Language Text

There are many softwares for entering the text in Indian language. Most of these softwares support two types of text entries namely, Phonetic text entry and Key-map based text entry.

Phonetic Text Entry

In this form of text entry, traditional keyboard with English keys are used. But while typing, the Indian alphabets are written phonetically (i.e., the way they sound as per the pronunciation) in English Script and then converted to corresponding language word.

For instance, to write “Odisha” in odia, we have to type ‘A’ ‘[’ ‘F’ ‘*’ ‘e’. It means when words typed according to their pronunciation in English and converting them to an Indian language is known as phonetic text entry of translation,

Note: The process in which we type Indian language words phonetically in English script and the tool(relevant phonetic key entry software) automatically converts them into corresponding language words is called as **transliteration**. Essentially, Transliteration is the conversion of a text from one script to another.

Key-map Based Text Entry

In this method the keyboard keys are mapped to specific characters using a keymap. The whole arrangement of mapping the keyboard keys to specific language characters is known as keymap. Under this arrangement usually the typewriter order of characters in respective language are stored in forms of multiple tables to respect the complete key mapping.

Among these tables, one table is for key-mapping without any additional key press (like shift), another could be with shift key pressed, and still another could be with control key pressed and so on.

InScript(India Script) is the standard keyboard layout for Indian scripts using a standard 104 or 105 key layout. This keyboard layout is built into most of the major operating system including Windows, Linux and Mac OS systems.

It is also available in some mobile phones and in Apple’s iOS 5 and higher. This keyboard layout is used for inputting text in Indian language written in Brahmic-scripts as well as the Shantali language written in non-Brahmic script.



Fig. 3 : InScript Keyboard lay out

UNICODE CHART FOR ODIA SCRIPT

Standard 8.0, Copyright © 1991-2015 Unicode, Inc. All rights reserved.

	0B0	0B1	0B2	0B3	0B4	0B5	0B6	0B7
0		ଏ	ଠ	ର	ଠ		ର	ୱ
1	ଠ		ଡ		ଠ		ଢ	ଝ
2	ଠ		ଢ	ଢ	ଠ		ଠ	
3	ଠ	ଠ	ଶ	ଢ	ଠ			୲
4		ଠ	ଡ					୲
5	ଅ	କ	ଅ	ବ				/
6	ଅ	ଶ	ଦ	ଶ		ଠ	ଠ	ୱ
7	ଝ	ଞ	ଝ	ଞ	ଠ	ଠ	ୱ	ୱ
8	ଝ	ଞ	ନ	ଞ	ଠ		୨	
9	ଝ	ଡ		ଝ			ଶ	
A	ଝ	ଠ	ଞ				ୱ	
B	ର	ଝ	ଞ		ଠ		ୱ	
C	ଝ	ଞ	ବ	ଠ	ଠ	ଠ	ୱ	
D		ଝ	ଞ	ଝ	ଠ	ଠ	ୱ	
E		ଝ	ଞ	ଠ			ୱ	
F	ଝ	ଞ	ଞ	ଠ		ଝ	ଞ	

1/4

1/2

3/4

1/16

1/8

3/16

Just put the **Row heading** next to the **Column heading** to get the code for any character.

For example;

The hexadecimal Unicode for the Odia Number '1' is '0B67'

The hexadecimal Unicode for the Odia Number '9' is '0B6F'

EXERCISE

Multiple Choice Questions :

1. Identify the open standard form the following,

(a) .win7	(b) .docx
(c) gimp	(d) .vlc
2. Which of the following is a voting member of Unicode Consortium?

(a) BSNL	(b) C-DOT
(c) C-DAC	(d) DeitY
3. The 'JAVA' programming language was developed by

(a) John G Kemeny	(b) Thomas A Kurtz
(c) Dennis M Retchie	(d) .James Gosling
4. PHP is

(a) an IDE	(b) an Open Standard
(c) an Operating System	(d) a Programming Language
5. 'MySQL' is

(a) an Open Source RDBMS	(b) A Proprietary RDBMS
(c) MS-Access	(d) An Apple RDBMS
6. DHTML is a

(a) Common Open Standard	(b) Programming Language
(c) File Extension	(d) Web Browser
7. Which of the following is a hexadecimal Unicode for an Odia character?

(a) B001	(b) 0B18
(c) 0B89	(d) 0B99
8. Open Type Fonts are

(a) 7-bit fonts	(c) 7-bit fonts
-----------------	-----------------

16-bit fonts (b)
(d)

8-bit
fonts
32-bit
font
s

Short Answer Type Questions :

1. Expand the terms: OSI, FLOSS, BIS, TDIL ISCII, OSS, HTML, OOo, DeitY, TTF, OTF, ODF
2. **.docx** file extension is used for Microsoft Word. It is an example of:
 - (a) Proprietary standard
 - (b) Open standard
3. Which protocol is responsible for delivering datagrams / data from the source host to the destination host?
4. Defiferentiate between Freeware and Shareware.
5. What is proprietary software?
6. What is Transliteration?
7. What is XML?
8. What is DHTML?
9. What are the advantages of Unicode?
10. What is open document format?

Long Answer Type Questions

1. Describe FOSS / FLOSS with examples.
2. Briefly describe different Common Open Standards.
3. Write short notes on
 - (a) TTF and OTF
 - (b) ASCII and ISCII
4. Write short notes on
 - (c) Phonetic Text Entry
 - (d) Key-map based Text Entry



Unit

2

INTRODUCTION TO PROGRAMMING

CHAPTER - 4

Programming using IDE

CHAPTER - 5

**Programming
Fundamentals**

CHAPTER - 6

Programming Guidelines

UNIT - 2

INTRODUCTION TO PROGRAMMING

CHAPTER - 4

PROGRAMMING USING IDE

2.1 INTRODUCTION

Language is a medium to express our views and feelings. To communicate with a computer, we need a language which a computer understands. Computer is a versatile machine capable of performing different tasks, from very simple mathematical operations to complex mathematical calculations and from simple designs to multimedia animations. A computer performs a series of pre-defined jobs as per the instructions fed in to it. A set of such structured instructions is called a **Program** which tells the computer what to do. The process of writing specific instruction in a computer language is called **programming**. Person who writes such programs is known as a **programmer** or **coder**. The language used in a computer program is called as Programming Language. BASIC, FORTRAN, Pascal, COBOL, C, C++ and JAVA are some of these programming languages. Programs should optimally, be functional, elegant, efficient, user friendly, and self-documenting.

What is Java?

Java (originally known as **oak**) is a computer programming language (which is also a high-level language because of its similarity to ordinary English language) has a set of rules that determine how the instructions are to be written. These rules are known as its *syntax*. Once a program has been written, the instructions, which are in high-level language form, gets translated into binary codes that computers can understand and execute.



History of Java

James Gosling (a Canadian Computer Scientist, Born 19th May 1955) initiated the Java language project in June 1991 for use in one of his many set-top box projects. The language, initially called Oak after an oak tree that stood outside Gosling's office, also went by the name Green and ended up later being renamed as Java, from a list of random words.

Sun released the first public implementation as Java 1.0 in 1995. It is a **Write Once, Run Anywhere**(WORA) type language, providing no-cost run-times on popular platforms. On 13 November 2006, Sun released much of Java as free and open source software under the terms of the GNU (GNU is an operating system which is completely free software system) General Public License (GPL). On 8 May 2007, Sun finished the process, making all of Java's core code free and open-source, aside from a small portion of code to which Sun did not hold the copyright.

Advantages / Characteristics of java.

Java is a High-level programming language with the following characteristics:

- **Object Oriented:** Java is purely objected oriented language and implements all the object oriented principles. So in java maintenance of code is very easy. It helps the programmer to develop more reliable and error free code.
- **Platform independent:** Languages can be platform dependent or independent, but Java is a Platform independent which is portable & easily executed on all operating systems (windows, Linux, solaris) or Java programs can run on any platform.
- **Simple:** Java is designed to be easy to learn. If you understand the basic concept of OOP Java would be easy to develop.
- **Secure:** As java generates an intermediate code called **ByteCode** which is very secure because nobody can understand anything if any one tries to access the code
- **Architectural-neutral:** Java compiler generates an architecture-neutral object file format which makes the compiled code to be

executable on many processors, with the presence of Java runtime system.

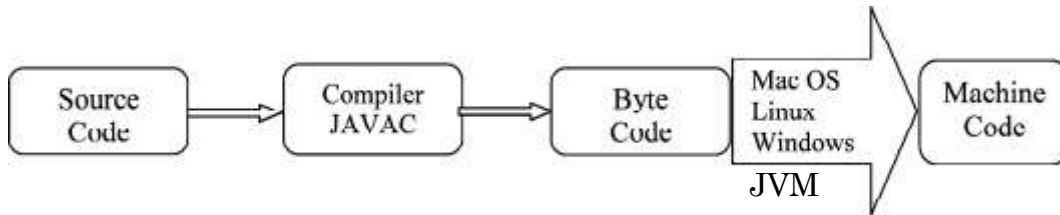
- **Portable:** Being architectural-neutral and having no implementation dependent aspects of the specification makes Java portable. Compiler in Java is written in ANSI C with a clean portability boundary which is a POSIX subset.
- **Robust:** Java makes an effort to eliminate error prone situations by emphasizing mainly on compile time error checking and runtime checking.
- **Multithreaded:** With Java's multithreaded feature it is possible to write programs that can do many tasks simultaneously. This design feature allows developers to construct smoothly running interactive applications.
- **Interpreted:** Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light weight process.
- **High Performance:** With the use of Just-In-Time compilers, Java enables high performance.
- **Distributed:** Java is designed for the distributed environment of the internet.
- **Dynamic:** Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry extensive amount of run-time information that can be used to verify and resolve accesses to objects on run-time.

Java Virtual Machine (JVM)

JVM is a run time environment which acts as an interpreter and translates the byte code into object code (machine language). It is a platform-independent execution environment that converts Java byte code into object code and executes it.

Most programming languages (like C/C + +, etc.) compile source code directly into machine code that is designed to run on any specific operating system such as Windows or Mac. This creates platform dependency. But the

developers of Java wanted it to be a platform independent language. So the concept of JVM was included in Java.



Source Code

The core program or text written in any computer language (like C, C++, Java, etc.) is called source code. Usually it is a collection of computer instructions written by using any human readable computer language. Source code files have the extension class.

Object Code

The program in the form of machine instructions or binary instructions (i.e., in computer readable form). It is generally produced by compiler/interpreter, but in case of Java, it is produced by JVM.

Byte Code

In Java, when a source code is compiled, it doesn't directly convert into object code, rather it converts into what, is known as byte code. So, a byte code is machine instruction that the Java compiler (Javac) generates.

Note :

- ☞ **Javac** –compiler that converts source code to byte code.
- ☞ **JVM**- interpreter that converts byte code to machine language code.
- ☞ **Source Code** (program.java) Æ compile Æ Byte Code (program.class)
- ☞ **Byte Code** (program.class) Æ JVM Æ Machine Code

2.2 RAPID APPLICATION DEVELOPMENT (RAD) USING IDE

Rapid application development is abbreviated as RAD. It is an object-oriented approach for software development, being introduced by James Martin in the 1980s. It includes a method of development as well as software tools commonly known as ICU. where,

- I-Iterative development.
- C-Construction of prototypes.
- U-Use of Computer-Aided Software Engineering (CASE) tools.

RAD includes analyzing, designing, building and testing phases the form of short iterative cycle which has a number of rare advantages over the traditional sequential development methods.

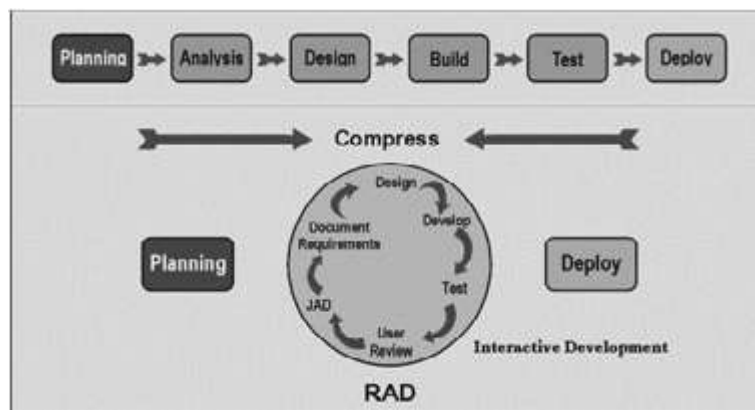


Figure 2.A.1 : Rapid Application Development

Rapid application development is a response of different Processes developed in the 1970s and 1980s, such as the Structured Systems Analysis and Design Method and other Waterfall Model.

Following are two most popular RAD systems;

- NetBeans IDE** is a cross-platform RAD tool for creating visual desktop, mobile and web applications for Linux, Windows and Mac OS X.
- MyEclipse** is a RAD environment, focusing on enterprise Java and web application development.

Some of the Advantages of RAD are as follows

- Increased speed of software development via including rapid prototyping, virtualization of system related routines, the use of CASE tools, and other techniques.
- Reduced complexity of development.
- More focus on simplicity and usability of GUI components.

Java Development Kit (JDK)

Java programs are of mainly two types-applets and applications. Applets are the Java programs executed by a web browser while applications are the programs that execute directly on your machine. JDK is a program development environment for writing Java applets and applications.

Sun decided to give away a Java Development Kit (JDK) that would provide basic tools needed for Java programming. JDK provides all the tools needed to write powerful Java applications or applets.

JDK includes JRE (Java Runtime Environment) and some other basic tools, like an interpreter/loader (Java), a compiler (javac), an archiever (jar), a documentation generator (javadoc), etc.

These tools are described below

1. JRE (Java Runtime Environment)

It is the environment which is used to run the predeveloped applets or applications in JDK.

2. Javac

Java compiler is named as Javac. The Javac takes input source code and converts them into compiled byte code.

3. Java

The Java interpreter/loader, normally called as Java. It can be used to execute Java applications. The interpreter translates byte code directly into program actions.

4. Java doc

It is used to generate our source code into HTML files documentation.

5. Jar

It is an archiever used to create jar files. A JAR file is a file that contains the class, image and sound files for a Java application or applet gathered into a single file and possibly compressed.

6. Applet Viewer

This small program provides a real Java environment for testing applets. It loads the HTML files (in which the applet has been embedded) and displays the application in a browser like window.

Integrated Development Environment (IDE)

An Integrated Development Environment is abbreviated as IDE. It is a software tool which provides a way to create and debug programs easily. IDEs are designed to maximize programmer productivity by providing developing of components with similar user interfaces.

An IDE brings all of the programming tools into one convenient place. There was a time when programmers had to edit files, save them, run the compiler and linker, build the application and then run it through a debugger and all these works were done separately.

IDEs overcome this problem. Today, IDEs bring editor, compiler, linker and debugger into one place along with project management tools to increase programmer productivity.

In other words, IDE is a software program that is designed to help programmers and developers to build software.

Most IDEs include

- ☞ A source code editor, similar to an HTML text editor. It is where programmers write the source code for their programs.
- ☞ A compiler which converts the source code into intermediate code (Byte code) and an interpreter (JVM) to execute that compiler generated byte code.
- ☞ Build automation tools help to automate the processes that need to happen with most software development like compiling debugging and deployment.
- ☞ A debugger helps to point out the exact spot where there is a problem in the source code.
- ☞ A Graphical User Interface (GUI) builder simplifies the creation of GUIs.

IDEs provide a user friendly framework for many modern programming languages, such as Visual Basic, Java and PowerBuilder, many people designing websites today *use an* IDE package such as Home Site, Dreamweaver, FrontPageetc. One of the most important features of an IDE tool is that the program code can be continuously parsed while it is being edited and provides instant feedback when syntax errors are occurred.

Some of the common examples of IDE tools are; Microsoft's Visual Studio, Oracle's NetBeans, Eclipse, Code Blocks, Aptana Studio etc.

NetBeans IDE

NetBeans IDE is a free, open source, integrated development environment (IDE) that enables you to develop desktop, mobile and web applications. The IDE supports application development in various languages, including Java, HTML5, PHP and C++. The IDE provides integrated support for the complete development cycle, from project creation through debugging, profiling and deployment. The IDE runs on Windows, Linux, Mac OS X, and other UNIX-based systems.

The IDE provides comprehensive support for JDK technologies and the most recent Java enhancements. It is the first IDE that provides support for JDK 8, Java EE 7, and JavaFX 2. The IDE fully supports Java EE using the latest standards for Java, XML, Web services, and SQL and fully supports the Glassfish Server, the reference implementation of Java EE.

NetBeans IDE 8.1 is a free and open-source Java Integrated Development Environment. It is the first free IDE providing support for JDK 8 previews, JDK7, Java EE 7 including its related HTML5 enhancements and JavaFX 2.

NetBeans IDE incorporates many tools to facilitate creating, managing and running a program, including features like a Graphical User Interface (GUI) for development, a fancy text editor, an output window, links for compilation errors, buttons to compile and run the program, a debugger and many more.

Integrated modules of a NetBeans IDE are

(i) NetBeans Profiler

It is a tool for the monitoring of Java applications. It helps developers to find memory leaks and optimize speed.

- **GUI Design Tool**

It enables developers to prototype and design Swing GUI by dragging and positioning graphical components.

- NetBeans JavaScript Editor

The NetBeans JavaScript editor provides extended support for JavaScript, Ajax, and CSS.

Some advantages of NetBeans IDE are

- (i) Best support for latest advanced Java technology
- (ii) Fast and smart code editing.
- (iii) Easy and efficient project management.
- (iv) Supports rapid application development mechanism.
- (v) Enable users to write bug free code.

(ii) Java Editors:

To write your Java programs, you will need a text editor. There are even more sophisticated IDEs available in the market. But for now, you can consider one of the following:

- **Notepad:** On Windows machine you can use any simple text editor like Notepad (Recommended for this tutorial), Text Pad.
- **NetBeans:** is a Java IDE that is open-source and free which can be downloaded from <http://www.netbeans.org/index.html>.
- **Eclipse:** is also a Java IDE developed by the eclipse open-source community and can be downloaded from <http://www.eclipse.org/>

2.3 FAMILIARIZATION OF IDE USING BASIC INTERFACE COMPONENTS

NetBeans IDE provides a way to interact with the different components of JAVA through visual graphical elements (Pictures, icons, symbols, etc.). To start NetBeans IDE follow the steps:-

Step1: Click Start Button Æ All programs Æ NetBeans IDE 8.1

Or

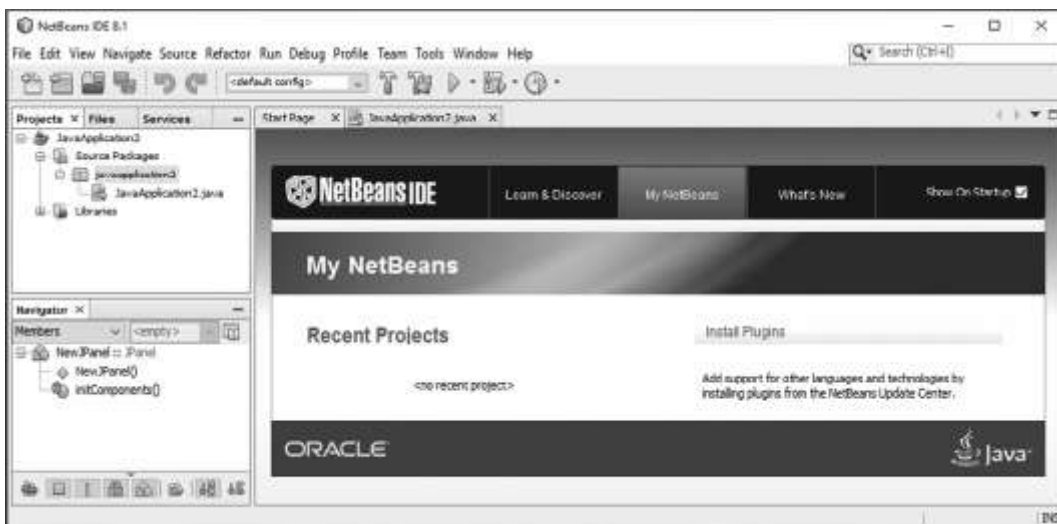
Double click on the shortcut icon of NetBeans IDE 8.1 on the Desktop.

Step2 : The loading screen of NetBeans IDE 8.1 will appear on your desktop.



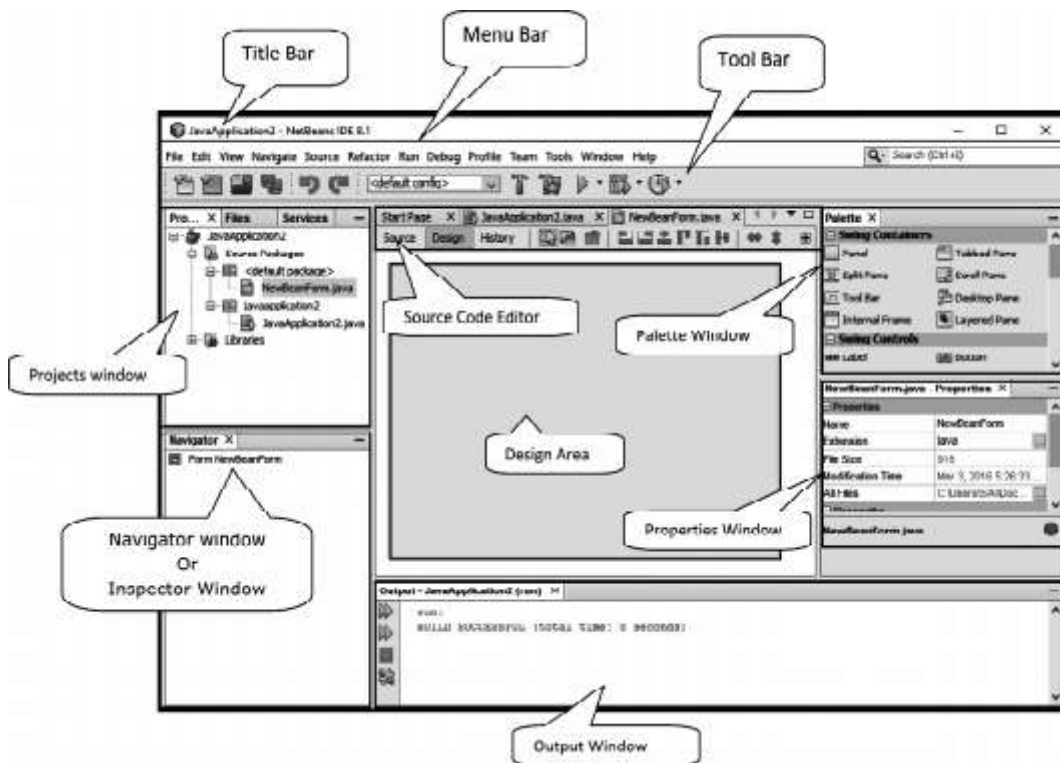
Loading Screen of NetBeans IDE 8.1

Step 3: After loading , the welcome screen of NetBeans IDE 8.1 will appear.



Welcome Screen of NetBeans IDE 8.1

Different parts of IDE windows



The Net Beans IDE contains various controls and menus via which, a user can easily interact and work according to their requirements.

1. **Title Bar:** It displays the name of the application or a program.
2. **Menu Bar:** It provides the various commands used to work with Net Beans IDE, such as file, edit, view, etc.
3. **Tool Bar:** It provides a way to quickly access the commonly used commands.
4. **Palette Window:** This window contains all the containers and other graphical components such as labels, panels, text fields etc.
5. **Projects Window:** This window shows a logical structure of different project contents.
6. **Inspector Window:** This window displays a hierarchical tree structure of all the components used in the current form.

7. **Properties Window:** This window displays the editable properties of the currently selected graphical component.
8. **Design Area:** It is the primary area used for creating and editing of Java forms.
9. **Source Editor:** It is a fully featured text editor program which includes a compiler, debugger and other components of IDE.
10. **Output Window:** This window shows the output messages from the Net Beans IDE. Output window automatically shows the messages after compilation of a program.

2.4 GETTING FAMILIAR WITH JAVA SWING USER INTERFACE COMPONENTS :

Components & Container

In Java, JFC (Java Foundation Classes) support GUI features. JFCs are pre-written code in the form of class libraries (coded routines) that give the programmer a comprehensive set of GUI routines to use. JFC is an extension of the original Java Abstract Window Toolkit (AWT).

Graphical User Interface (GUI)

It is a type of user computer interface that allows users to take advantages of computer graphical capabilities to make program easier. A GUI is developed with the help of graphical elements such as buttons, scroll bars, list and text fields. These elements are called components of GUI

Java GUI programming involves two packages, the original abstract windows kit and newer swing toolkit. Components present in AWT are all present in swing. Swing components have the prefix J to differ them from AWT. *For example*, JLabel instead of Label.

Java Swing

Java Swing is a group of classes that provides more powerful and flexible GUI (Graphical User Interface) components than does the AWT (Abstract Window Toolkit). Swing is built on the foundation of AWT and concentrate to the limitations present in the AWT.

The two key features of swing are— light weight components and a pluggable look and feel. Together they provide an elegant, yet easy to use solution to the problems occurred in AWT.

Components in Swing

A component is an independent visual control, such as a push button or slider. Swing components are derived from the `JComponent` class.

For example, `JComponent` supports the Pluggable Look And Feel (PLAF), which means that each swing component is provided by java rather than by native peers.

The look and feel of a component is under the control of swing. `JComponent` inherits the two AWT classes; `Container` and `Component`, to give GUI data item a better look.

There are two types of graphical components available in java swing

- (i) **Container Control** It is the control that holds some other swing controls. *For example*, `Panel`, `Frame` etc.
- (ii) **Child Control** Those controls which are available in the container control are known as child control.

For example, `Button`, `List` etc.

Some Basic Component of Swing

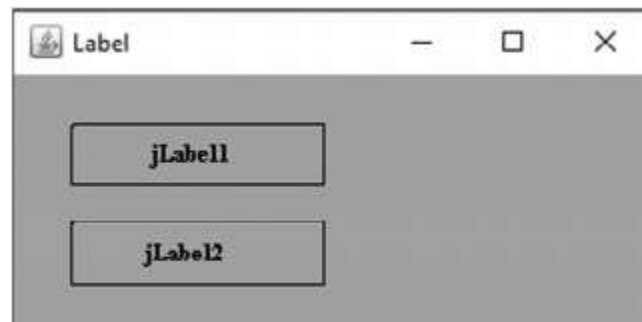
Swing components are represented by a class within the package `javax.swing`. This package must be imported into every swing program. The package `javax.swing` contains several classes that implements *the basic GUI swing components shown in the table below*

<code>JApplet</code>	<code>JButton</code>	<code>JCheckBox</code>	<code>JCheckBoxMenuItem</code>
<code>JColorChooser</code>	<code>JComboBox</code>	<code>JCheckBox</code>	<code>JComponent</code>
<code>JDialog</code>	<code>JEditorPane</code>	<code>JFileChooser</code>	<code>JDesktopPane</code>
<code>JFrame</code>	<code>JLabel</code>	<code>JLayeredPane</code>	<code>JInternalFrame</code>
<code>JList</code>	<code>JMenu</code>	<code>JMenuBar</code>	<code>JMenuItem</code>
<code>JOptionPane</code>	<code>JPopupMenu</code>	<code>JPanel</code>	<code>JPasswordField</code>
<code>JProgressBar</code>	<code>JRadioButton</code>	<code>JRootPane</code>	<code>JRadioButtonMenuItem</code>
<code>JScrollBar</code>	<code>JScrollPane</code>	<code>JSeparator</code>	<code>JSlider</code>
<code>JSpinner</code>	<code>JSplitPane</code>	<code>JTabbedPane</code>	<code>JTable</code>
<code>JTextArea</code>	<code>JTextField</code>	<code>JTextPane</code>	<code>JToggleButton</code>
<code>JToolBar</code>	<code>JToolTip</code>	<code>JTree</code>	<code>JViewport</code> , <code>JWindow</code>

The components given the table provide an easy way to create graphical inbuilt software programs. Few of them are described below with their *properties* and *methods*.

1. Label:

Label is the display area for a short text string or an image. It is created through JLabel component. It is easy to use component which does not react on the input actions performed by an user. A label displays a single line of read only text. The text can be changed by the application, but a common user cannot edit it directly.



Constructor

A java constructor has the same name as the name of the class to which it belongs. Constructor's syntax does not include a return type, since constructors never return a value. It is executed when we create new object of that class.

JLabel provides following constructors

JLabel (Icon i)

JLabel (String str)

JLabel (String str, Icon i, int align)

Here, **str** and **i** are the text and icon respectively used for the label. The align argument specifies the horizontal alignment of the data item within the dimensions of the label

It must be one of the following values

LEFT, RIGHT, CENTER, TOP and DOWN

Properties

- (a) *Background* :This property is used to set the background colour of a label. The colour of the label is displayed when opaque property is set to true. The default setting value is false.
- (b) *Enabled* :This property is used to set whether the label is active or not at run
a. time.
- (c) *Font* :This property is used to set a font style for the text displayed on the label.
- (d) *Foreground* :This property is used to set the foreground colour to be displayed on the label. The foreground colour of the label is displayed when opaque property is set to be true but the default setting value is false.
- (e) *Text* :This property is used to specify the text to be displayed on the label.

Methods

- (a) *voidsetText(String)* :This method is used to set the text which will be displayed with the label.
For example,<label_name>.setText(“Name”);
It would set the “Name” text on the specified label.
- (b) *String getText()* :This method is used to fetch the text displayed on the label. *For example*,String str = <label_name>.getText();
It returns the text value available on the specified label

2. Text Field

Text Field is the simplest text component which allows a user to feed input text at the run-time. It is derived from JTextField component, which provides the basic functionality common to swing components. JTextField allows you to edit one line of text. If you enter numeric data then you must have to convert it into text format because numeric data could not be displayed in a textfield.



JTextField provides following constructors

`JTextField (int cols)`

`JTextField (String str, int cols)`

`JTextField (String str).`

Here, **str** is the string to be initially presented and **cols** is the number of columns available in the textfield.

- ☛ If no string is specified, the textfield is initially empty.
- ☛ If the number of columns are not specified, it set as a specified strings length

Properties

- (a) **Background** : This property is used to set the background colour of the textfield.
- (b) **Editable** : This property is used to edit the contents of the textfield. If it is set to true then the text available in the text field can be edited by the user at run-time.
- (c) **Enabled** : It is used to set the textfield as activated or deactivated.
- (d) **Font** : This property is used to set a font style for the text to be displayed in the textfield.
- (e) **Foreground** : This property is used to set the foreground colour of the text to be displayed in the textfield.
- (f) **Text** : This property is used to specify the text to be displayed in the textfield.

Methods

- (a) **void setText(String)** This method is used to **set** the text displayed by the text field.
For example, <textfield_name>.setText(" Name");
It would set the "Name" text in the specified textfield.
- (b) **String getText()** This method is used to fetch the text displayed in a textfield.
For example, String str = <textfield_name>.getText()
- (c) **isEditable()** This method returns the current state of the textfield. If its value is true then the user can edit the text available in the specified textfield.
- (d) **isEnabled()** It indicates whether the textfield is in activated state or deactivated state.

3. Text Area

Text Area is a multi-line text component which enables a user to enter a large amount of text. It is created through JTextArea component class.



JTextArea provides these following constructors

```
JTextArea()
```

```
JTextArea(int numLines, int numChars)
```

```
JTextArea(string str).
```

Here, numLines specifies the height of the text area, and numChars specifies its width, in characters. Initial text can be specified by **str**.

Properties

- (a) **Background** : This property is used to set the background colour of the textarea.
- (b) **Foreground** : The property is used to the set foreground colour of the text available in the textarea.
- (c) **Enabled** : This property is used to set the textarea as enabled.
- (d) **Editable** Sets whether the textarea is editable or not. Default state/ value is true.

Methods

- (a) **String getText()** This method is used to fetch the text displayed by the textarea.

For example, String str=<textarea_name>.getText();

- (b) **void setText(String)** This method is used to set the text which will be displayed in the textarea.

For example, <textarea_name>.setText("Name");

It would set the "Name" text in the specified and textarea.

4. Button

A button is created through JButton class. JButton class provides the functionality of a push button (or simply say button). It allows an icon, string or both to be associated with the push button.

JButton provides following constructors

```
JButton(Icon i)
```

```
JButton(String str)
```

```
JButton(String str, Icon i)
```

Here, str and i are the string and icon used for the button

Properties

- (a) **Background** This property is used to set the background colour of the button. The colour of the button is displayed when it is set to be true.



- (b) **Enabled** This property is used to set whether the button is active or not at the run-time.
- (c) **Font** This property is used to set font style of the text displayed on the button.
- (d) **Foreground** This property is used to set the colour of the text displayed on the button.
- (e) **Text** This property is used to set the text specified on **the** button.
- (f) **Label** Used to set the label of the button.

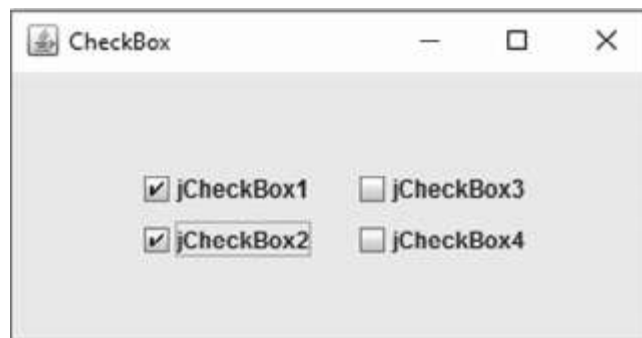
Methods

- (a) **void setText(String)** This method is used to set the text which will be displayed on the button.
For example, <button_name>.setText("Push Button");
It would set the "Push Button" text on the specified button.
- (b) **String getText()** This method is used to fetch the text displayed on the button.
For example, String str = <button_name>.getText();

5. Check Box

A check box is a graphical component that can be in one of the state as 'Selected' or 'Deselected'. It is created through JCheckBox class. JCheckBox is a labeled toggle switch whose super class is JToggleButton. Each time when the user clicks it, its state changes. A check box generates one item event and one action event per click.

- ☞ CheckBox1 Æ If checked, value = true/on.
- ☞ CheckBox3 Æ If unchecked, value = false/off.



JCheckBox provides following constructors

`JCheckBox(String str)`

`JCheckBox(String str, boolean selected)`.

The first constructor creates a checkbox that has the text specified by **str** as a label and the second constructor provides a way to prepare a selectedcheckbox.

Properties

- (a) **Button Group** Used to set a group of buttons in which the current control belongs.
- (b) **Font** This property is used to set the font which will be displayed with the checkbox.
- (c) **Foreground** This property is used to specify the font colour of the checkbox.
- (d) **Label** Used to set the checkbox label by replacing `setText(String)` method.
- (e) **Selected** This option is used to set the current value of a checkbox. If it is set to true, it is displayed as checked otherwise unchecked.
- (f) **Text** This property is used to set the text to be displayed with the checkbox.

Methods

- (a) **void setText(String Str)** This method is used to set the text which will be displayed with the check box.
For example, <Checkbox_name>.setText("New CheckBox");
It would set the "New CheckBox" text with the current checkbox.
- (b) **String getText()** This method is used to fetch the text displayed with the checkbox.
For example, String str=<Checkbox_name>.getText();
- (c) **boolean isSelected()** This method returns the current state of the checkbox. If it is checked, returns true otherwise false or unchecked.
- (d) **void setSelected(Boolean b)** This method is used to set the current state of the checkbox.

6. Radio Button

Radio button is a group of mutually exclusive buttons, in which only one button can be selected at a time. It is supported by the `JRadioButton` class, which extends `JToggleButton`.

JRadioButton provides following constructors

`JRadioButton(String str)`

Here, **str** is the label of the button.



Properties

- (a) **Background:** This property is used to set the background colour of the radio button. The colour of the radio button is displayed when it is set to be true.
- (b) **Button Group:** This property is used to set a name of the group of radio buttons, to which the current radio button belongs.
- (c) **Enabled :**This property is used to set whether the radio button is active or not at runtime.
- (d) **Font:** This property is used to set the font style of the text to be displayed with the radio button.
- (e) **Foreground :**This property is used to set the font colour of the text displayed with the radio button.
- (f) **Label:** This property is used to set the text displayed with any other control.
- (g) **Selected:** This property is used to set the current state of the radio button. If it is set to ture, it is displayed as checked otherwise unchecked.
- (h) **Text:** This property is used to set the text to be displayed with the radio button.

Methods

- (a) **void setText(boolean b)** This method is used to set the text, which will be displayed with the radio button. *For example, <radiobutton_name>.setText("New");*
It would set the 'New' text with the current radio button.
- (b) **String getText()** This method is used to fetch the text displayed with the radio button.
For example, String str=<radiobutton_name>.getText()
- (c) **boolean isSelected()** This method returns the current state of the radio button. If it is selected, it returns true otherwise false.
- (d) **void setSelected(boolean b)** This method sets current state of the radio button for selection process.

7. Frame

It is a top-level container to which components are added. It provides a home for the other components of the interface, and dictates the overall feel of the application.

A frame is created through `JFrame` component and `JFrame` is a subclass of `java.awt.Frame` and therefore, it inherits all the features of an AWT frame. `Frame` has decorations such as a border, a title, and supports button components that close or iconify the window. Applications with GUI usually include at least one frame.

JFrame provides the following constructors

- ☛ `JFrame()` Constructs a new frame that is initially invisible.
- ☛ `JFrame (string str)` Constructs a new initially invisible frame with the specified title.

Properties

- (a) **Background** This property is used to set the background colour of the frame.
- (b) **Title** This property is used to set the title of the frame.
- (c) **DefaultCloseOperation** This property is set to `hide-on-close`, a value taken from `windows constants`. This indicates that closing a `JFrame` window results in setting the frame invisible.

Methods

- (a) **`int getDefaultCloseOperation()`** Returns the operation that occurs when the user initiates a 'close' on this frame.
- (b) **`Void setDefaultCloseOperation()`** Sets the operation that will happen by default when the user initiates a 'close' on this frame.
- (c) **`string getTitle`** Gets the title of the frame.
- (d) **`void setTitle(String)`** Sets the title for this frame to the specified title.

8. Dialog Box

A dialog box is a small separate sub-window that appears on the screen for providing/requesting information to/from the users. According to the Java

documentation, a dialog window is an independent sub-window meant to carry temporary notice apart from the main swing application window. In Java language, the dialog box is supported by the Java Swing control.

It supports the following types of controls

(a) JDialog

The JDialog is a Swing window dialog which provides the normal behaviour of a window. It displays a dialog box which has minimize, maximize and close icons in the title bar of the dialog box. Most dialogs present an error message or warning to a user, but dialogs can present images, directory trees, or just about anything compatible with the main swing application that manages them.

(b) JOptionPane

In Java language, the JOptionPane is used to create a pop-up window with varied contents. It can display an alert message box or an input box.

JOptionPane contains classes used for a Graphical User Interface (GUI) that facilitates data entry and data output. This class contains methods to display a dialog box.

Swing JOptionPane allows you to

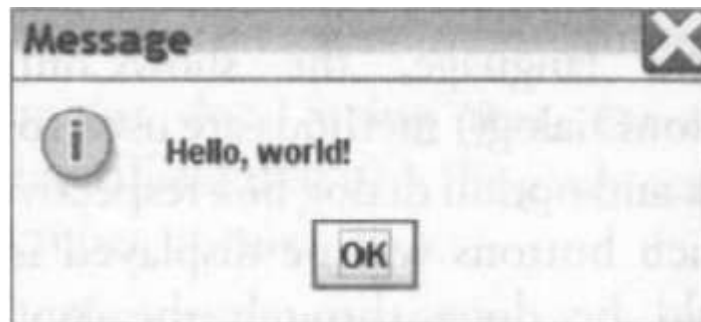
- ☞ Create a dialog box
- ☞ Display information
- ☞ Request information
- ☞ Give the user a choice with buttons.

Structure of JOptionPane

The basic elements of JOptionPane are as follows

- (i) Icon :** The icon determines the type of dialog box or message box. There are four default options available. These are error, information, warning and question.
- (ii) Message :** The message option is used to set the text information that we want to convey through the dialog box.
- (iii) Input Area :** The input area allows the user to provide a response in the form of an input. For user response, we can use JTextField, JComboBox or JList.

- (iv) **Button :** The button area displays a set of buttons such as OK/Cancel, Yes/No etc. The buttons can be customized according to the user's requirements.



Dialog Type





We can create four types of predefined dialog boxes. *These are as follows*

- (i) **Input Dialog:** This type of dialog box is used to enter the data. We can use `JTextField`, `JComboBox` or `JList` for taking input. There are two buttons; OK and Cancel. To create an input dialog box the `showInputDialog()` method is used.
- (ii) **Confirm Dialog:** This type of dialog box is used to ask from the user about the confirmation of any information. It includes buttons Yes/No, OK and Cancel etc. To create a confirm dialog box, the `showConfirmDialog()` method is used.
- (iii) **Message Dialog** The message dialog box is used to simply display some information to the user. It includes only a single OK button. To create a message dialog box, the `showMessageDialog()` method is used.
- (iv) **Option Dialog** The option dialog is a flexible type dialog box. It can be used to create a dialog box according to the user's need.

To create an option dialog box, the `showOptionDialog()` method is used.

Message Type

The icon of the `JOptionPane` is specified through the message type values. Following table shows the possible values.

IDE value	Equivalent Code	ICON
-1	<code>JOptionPane.PLAN_MESSAGE</code>	No Icon
0	<code>JOptionPane.ERROR_MESSAGE</code>	
1	<code>JOptionPane.INFORMATION_MESSAGE</code>	
2	<code>JOptionPane.WARNING_MESSAGE</code>	
3	<code>JOptionPane.QUESTION_MESSAGE</code>	

Option Type

In Java language, the `showConfirmDialog()` and `showOptionDialog()` methods are used to show the confirm dialog box and option dialog box respectively, have choice to select which buttons will be displayed in the dialog box. This could be done through the 'option type' in the properties window.

IDE value	Equivalent Code	Buttons Displayed
-1	<code>JOptionPane.DEFAULT_OPTION</code>	OK
0	<code>JOptionPane.YES_NO_OPTION</code>	YES and NO
1	<code>JOptionPan.es.YES_NO_CANCEL_OPTION</code>	YES, NO and CANCEL
2	<code>JOptionPane.OK_CANCEL_OPTION</code>	OK and CANCEL

Alternative Method for Creating JOptionPane

We can create the JOptionPane also through coding. For this purpose, *we have to follow these steps*

- (i) First we have to specify the import command to import the javax.swing API in the application as per following syntax

```
import javax.swing.*;
```

or

we can add following command to include the JOptionPane directly

```
import javax.swing.JOptionPane;
```

- (ii) Then, we have to determine whether the dialog box is to be used as message box, confirmation box, option box or input box and accordingly, *we have to use any of the following methods*

(a) **void showAlertDialog()** Displays a message and waits until the user acknowledges it by clicking Ok.

(b) **void showConfirmDialog()** Displays a message and waits for user response through button selection.

(c) **void showOptionDialog()** Displays a message and waits for user response from a group of options.

(d) **void showInputDialog()** Displays a message and waits for input from the user.

- (iii) Then, we need to specify the parent frame name.

- (iv) Then, we have to type the message which is to be displayed within the dialog box.

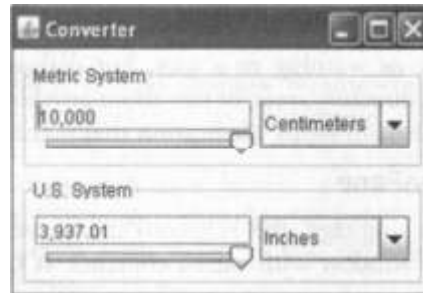
- (v) For selecting the icon to be displayed, we need to specify the IDE code.

- (vi) Then, we have to select a set, for the types of buttons to be displayed within the dialog box.

Note: Step (ii) to (vi) can be done simultaneously in following statement
`JOptionPane.showMessageDialog(null, "Message to be displayed", dialog);`

9. Panel

It is a lightweight container used for holding components which include JButton, JLabel, JList, etc. A panel is created through JPanel component. They are used to let the developer have more control over positioning and style of widgets on the screen and to give a better structure to the code. JPanels are fundamental in the creation of a professional GUI. JPanel provides the following constructors



- ☛ **JPanel()** Creates a new panel with a double buffer and a flow layout.
- ☛ **JPanel(boolean)** Creates a new panel with flow layout and the specified buffering strategy.
- ☛ **JPanel(LayoutManager)** Creates a view buffered panel with the specified layout manager.
- ☛ **JPanel(LayoutManager, boolean)** Creates a new panel with the specified layout manager and buffering strategy.

Methods

- (i) **String paramString()** Returns a string representation of this panel.
- (ii) **AccessibleContext getAccessibleContext()** Gets the accessible context associated with this JPanel.
- (iii) **void add(Component)** Adds the specified component to the panel.
- (iv) **int getComponentCount()** Gets the number of components in this panel.
- (v) **void remove(Component)** Removes the specified component.

10. Scroll Pane

A scroll pane is created through JScrollPane component. It provides a scrollable view of a component. When screen real state is limited, use a scroll pane to display a component that is large or one whose size can change dynamically. Scroll pane provides a way of scrolling horizontally and vertically if the widget becomes any bigger than the viewport size. JScrollPane is used primarily in JTextArea, JList, and JTable where the data displayed by the widget is changed dynamically and can really affect the size and dimension of the widget.

JScrollPane provides the following constructors

JScrollPane() Creates an empty (no viewport view) JScrollPane where both horizontal and vertical scrollbars appear when needed.

JScrollPane(Component view) Creates a JScrollPane that displays the contents of the specified component, where both horizontal and vertical scrollbars appear whenever the component's contents are larger than the view.

JScrollPane(Component view, int vsbPolicy, int hsbPolicy) Creates a JScrollPane that displays the view component in a viewport whose view position can be controlled with a pair of scrollbars.

JScrollPane(int vsbPolicy, int hsbPolicy) Creates an empty (no viewport view) JScrollPane with specified scrollbar policies.

11. Password Field

The password field is used to accept the password for an application during the execution of a program. It is created through JPasswordField component. The password field is similar to textfield but the text displayed in a password field is in the form of an echo character (echochar). The default character is *. It means whatever characters the user enters in this field during the execution of the program, an echo character (usually *) will be displayed for the same. Password field is used for the security purpose. It does not show the character that user types.

JPasswordField provides the following constructors

- ☞ **JPasswordField()** Constructs a new JPasswordField, with a default document, null starting text string, and 0 column width.
- ☞ **JPasswordField (Document doc, String text, int columns)** Constructs a new JPasswordField that uses the given text storage model and the given number of columns.
- ☞ **JPasswordField (int columns)** constructs a new empty JPasswordField with the specified number of columns.
- ☞ **JPasswordField (String text)** Constructs a new JPasswordField initialized with the specified text.
- ☞ **JPasswordField (String text, int columns)** Constructs a new JPasswordField initialized with the specified text and columns.

Properties

- (a) **Background** This control is used to set the background colour for the password field.
- (b) **Font** This control sets the font for the text of a password field.
- (c) **Foreground** This control is used to set the colour of the text to be displayed on password field.
- (d) **Text** This control is used to set the text for a password field.
- (e) **echochar** This property is used to set an echo character, *i.e.*, the character to be displayed in the password field.

Methods

- (i) **void setEchoChar(char)** This method is used to set an echo character for the password field *i.e.*, the character to be displayed in the password field, when the user press any key on the keyboard.
- (ii) **char getEchoChar()** This method returns the echo character of the password field.
- (iii) **char[] getPassword()** This method returns the text displayed in the password field.
- (iv) **void selectAll()** This method selects all characters of the text field.

12. Combo Box

A Combo Box, which lets the user choose one of several choices, can have two very different forms. The default form is the uneditable combobox, which features a button and a drop-down list of values. The second form, called the editable combobox, features a textfield with a small button abutting it. The user can type a value in the textfield or click the button to display a drop-down list. Here's what the two forms of comboboxes look like in the Java look and feel.

Combo boxes require little screen space, and their editable (textfield) form is useful for letting the user quickly choose a value without limiting the user to the displayed values. A JComboBox is created through JComobox component.

JComobox provides the following constructors

- ☞ **JComboBox()** Creates a JComboBox with a default data model. `JComboBox (ComboBoxModel<E>aModel)` Creates a JComboBox that takes its items from an existing `ComboBoxModel`.
- ☞ **JComboBox (E [items)** Creates a JComboBox that contains the elements in the specified array.
- ☞ **JComboBox(Vector<E>items)** Creates a JComboBox that contains the elements in the specified vector.

Properties

- (a) **Background** It is used to set the background colour of the combobox.
- (b) **ButtonGroup** Used to set a group of buttons in which the current control belongs.
- (c) **Editable** It is used to edit the contents of the textfield of the combobox. If it is set to true, then the text given in the combobox can be edited by the user at runtime.
- (d) **Enabled** It is used to set the combobox as activated or deactivated on runtime.
- (e) **Font** Used to set a font style for the text to be displayed.
- (f) **Foreground** Used to set the foreground colour of the text to be displayed.
- (g) **Model** This property is used to specify the data model of the combobox.
- (h) **SelectedIndex** This property is used to get or set the index specifying the currently selected item.
- (i) **SelectedItem** This property is used to get or set currently selected item in the combobox.
- (j) **Text** This property is used to specify the text to be displayed on the textfield of the combobox.

Methods

- (i) **Object getSelectedItem()** Returns the currently selected list object (or item) from the data model.

- (ii) **int getSelectedIndex()** Returns an integer specifying the currently selected list item, where **0** specifies the first item in the list, or **—1** if no item is selected or if the currently selected item is not in the list.
- (iii) **void setModel (ComboBoxModel)** Sets the data model that the JComboBox uses to obtain the list of items.

13. List

A List displays a list of objects and allows the user to select one or more items. A List is created through JListComponent. Lists can have many items, so they are often put in scroll panes.

In addition to JList, JComboBox swing components also present multiple selected items to the user. But the basic difference between list and combobox is that list allows you to select more than one item at a time, whereas combobox allows you to select only one item at a time.

JList provides the following constructors

- ☞ **JList()** Constructs a JList with an empty model.
- ☞ **JList (ListModel dataModel)** Constructs a JList that displays the elements in the specified, non-null model.
- ☞ **JList (Object[] listData)** Constructs a JList that displays the elements in the specified array.
- ☞ **JList (Vector listData)** Constructs a JList that displays the elements in the specified Vector.

Properties

- (a) **Background** This property is used to specify the background colour of the list.
- (b) **ButtonGroup** This property is used to set a group of button in which the current control belongs.
- (c) **Model** This control is used to set an associated object that stores the data displayed in the list.
- (d) **SelectedIndex** This control is used to select an item initially. By default it is set to **—1** *i.e.*, no item is selected.
- (e) **Editable** This property can be set true or false. If it is set true, then the text given in the list can be edited by the user at run time.

- (f) **Enabled** Sets true or false at run time display. If it is true then it does not function at run time.
- (g) **Font** It is used to set the font style and size of the component.
- (h) **Foreground** It is used to set the foreground or displayed text colour of the component.
- (i) **Text** This control is used to set the text for a list.
- (j) **SelectedItem** This control returns the selected items from the list.

Methods

- (a) **Object getSelectedvalue()** This method returns the selected cell index, when only one item is selected from the list. When multiple items are selected, it will return the first selected value. If no items is selected, then it will return **null**.
- (b) **ObjectgetSelectedvalues()** This method returns all selected values in the form of an array from the list, in increasing order based on their indices in the list. When no option is selected, it will return an empty array.

2.5 DEVELOPING GENERAL APPLICATION

Creating a new Project

To create a new application project called “JavaApplication1”:


1. Choose File \mathcal{A} New Project. Alternately, click the New Project icon in the toolbar.
2. From the Categories pane select Java and in the Projects pane, choose Java Application. \mathcal{A} Click Next.
3. Enter a name (in this case JavaApplication1) in the Project Name field and specify the project location by clicking on the Browse button. By default the project is saved in the NetBeans Projects folder in My Documents and so this is the default Project location displayed in this field.
4. Ensure that the Set as Main Project checkbox is selected and clear the Create Main Class field.
5. Click Finish.

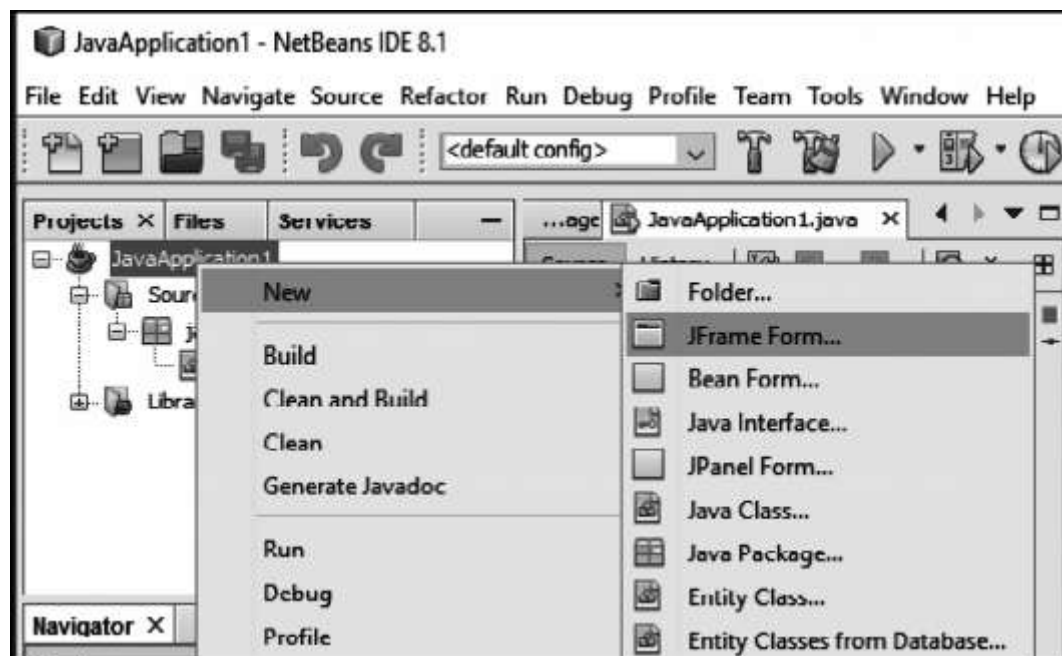
NetBeans creates the JavaApplication1 folder on your system in the designated location. This folder will contain all of the associated files of the project. The next step is to create a form.

To proceed with building our form, we need to create a container within which we will place the other required components of the form like a button. For all our applications we will choose the JFrame Form as the container to place other components.

Creating a new Form

To create a JFrame Form container:

1. In the Projects window, right-click the Book node and choose New  JFrame Form as shown in Figure
2. Enter Form Example 1 as the Class Name. This will be the name of your form.
3. Enter Book as the package. This should be the name given while creating the Project.
4. Click Finish.

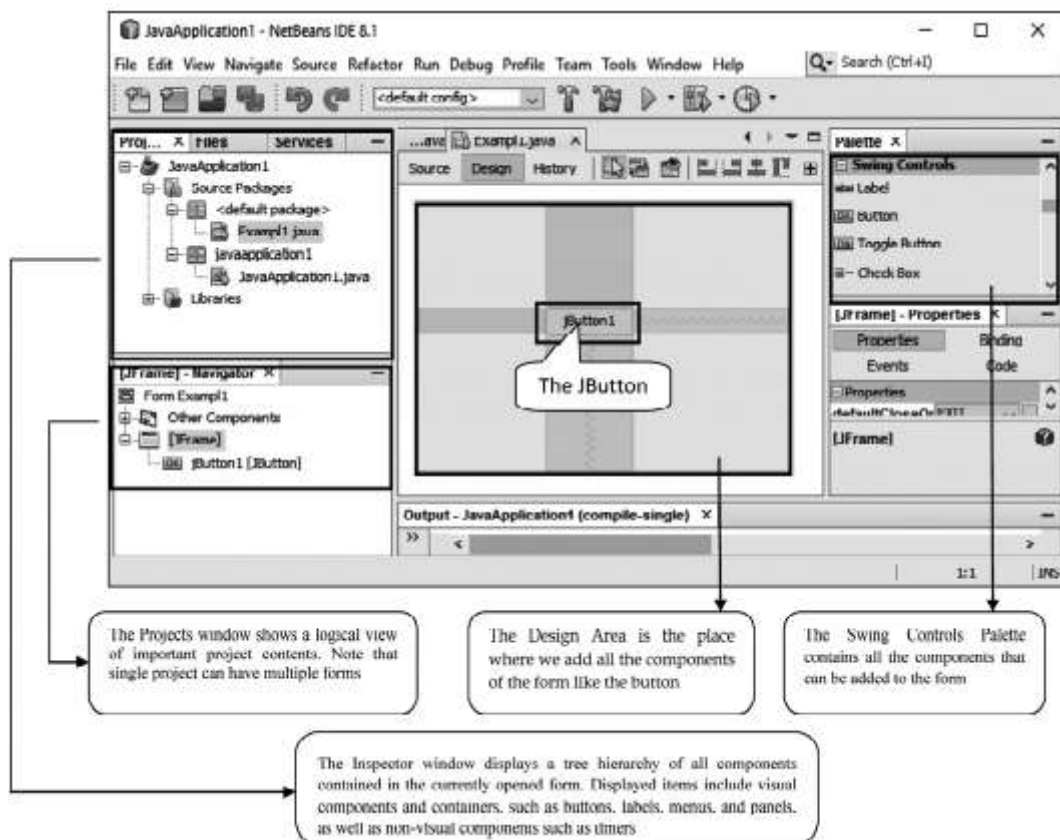


NetBeans creates the Form Example1 form within the application and opens the form in the Builder. Now we are ready to add components to our form.

Adding a Button Component to a Form

We want to add a button so follow the given steps to add a JButton to the form:

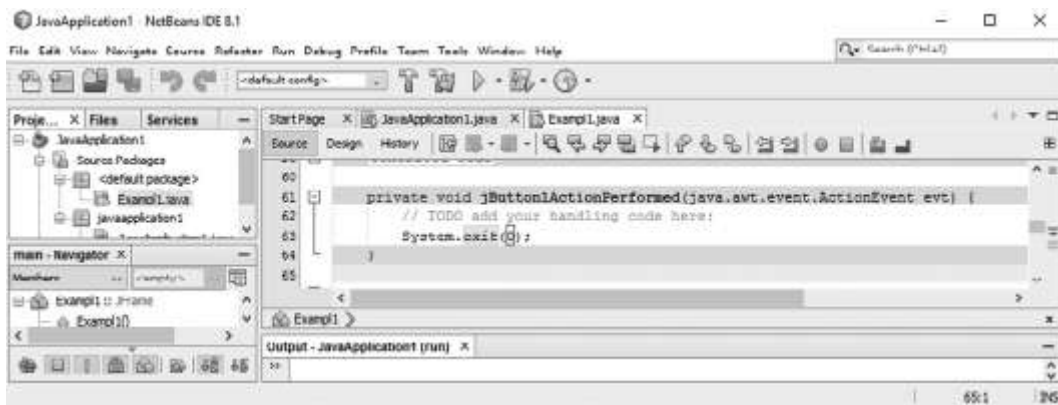
1. In the Palette window, select the JButton component from the Swing Controls category (displayed in Figure).
2. Move the cursor over the Form. When the guidelines appear (as displayed in Figure) indicating that the JButton is positioned in the desired location, click to place the button.



Adding a Button and Understanding the Different Windows

Attaching Code to a Form Component


After placing the button, the next step is to write a code to exit from the application on the click of this button. To do the same, double click on the button to attach a code with the event i.e. click of the button. Double clicking on the component opens up the source window and places the cursor on the point where code is to be added. Note that certain code is pre generated and cannot be changed. In the Source window add the single code line[`System.exit(0)`] as shown in Figure



Code to exit from an application

When we click the Source button, the application's Java source code in the Editor is displayed with sections of code that are automatically generated by the NetBeans Builder indicated by gray/blue areas, called Guarded Blocks. Guarded blocks are protected areas that are not editable in Source view. Note that we can only edit code appearing in the white areas of the Editor when in Source view.

Executing a File

Now that the code for the first application is ready let us test our first application. To execute the application simply select **Run**  **Run File** or press **Shift+F6**

On executing the first example, the window shown in Figure will appear. Click on the button and observe the result



Simple Button Application

As soon as we click on the button, the application ends and we return back to the Netbeans design window. So we learnt that the one line of code `system.exit(0)` causes the application to terminate successfully

Quick Recap - Steps for developing a Simple application

- Step 1 : Create a new Project
- Step 2 : Add a JFrame form
- Step 3 : Add the desired component from the Palette window using drag and drop feature
- Step 4 : Associate code with the component by double clicking the component.
- Step 5 : Add the source code.
- Step 6 : Test the form by pressing Shift+F6.

2.6 BASIC COMPONENT HANDLING METHODS

Any program that uses GUI (Graphical User Interface) such as Java application written for windows, is event driven. Event describes the change of state of any object. *For example*, pressing a button, entering a character in textbox.

Components of Event Handling : Event handling has three main components

1. **Events :** An event is a change of state of an object.
2. **Events Source :** It is an object that generates an event
3. **Listener :** A listener is an object that listens to the event. A listener gets notified when an event occurs.

Handling of Events

A source generates an event and sends it to one or more listeners registered with the source. Once event is received by the listener, they process event and return. Events are supported by a number of Java packages like java.util, java.awt and java.awt.event.

Basic Component Handling Methods

To handle different swing components, there are several methods by which the various properties of components can be handled and modified.

Some of the methods are as follows

1. **setText()**

This method is used to set the text of a component (button, label, textfield, textarea, checkbox etc) at run time.

Syntax : `Component_name.setText("text");`

Here, "text" is the display text to be shown on a particular component.

For example, `jButton1.setText ("Enter");`

In the above example, jButton1 is the component_name, "Enter" is the text displayed on the button and setText is the method name.

2. **getText()**

This method is used to fetch out or retrieve the text contained by the specified component (button, label, textfield, textarea, checkbox etc..) at run time.

Syntax : `Component_name.getText();`

For example, `String str = jTextField1.getText();`

In the above statement 'str' is a string type variable which holds the value of the text field named as jTextField1 using getText() method. By default this method returns a string value.

3. isSelected()

This method is used to get the current state of the component. It returns the value of the state in the form of true or false.

Syntax

```
Component_name.isSelected( );
```

For example,

```
Boolean b = jCheckBox1. isSelected( );
```

In the above statement 'b' is a Boolean type variable which holds the current state of the component named as jCheckBox1 using isSelected() method.

4. setSelected()

This method is used to set the current state of the component as selected or unselected.

```
Syntax :Component_name.setSelected();
```

For example, jCheckBox1.setSelected (true);

In the above statement, the checkbox name as jCheckBox1 is set to checked.

The following table summarizes the data members (properties) and methods of all classes learnt during the course till now.

Class	Swing Control	Properties	Methods
JButton	jButton1	<ul style="list-style-type: none"> • Background • Enabled • Font • Foreground • Text • Label 	<ul style="list-style-type: none"> • getText() • setText()
JLabel	jLabel1	<ul style="list-style-type: none"> • Background • Enabled • Font • Foreground • Text • ToolTipText 	<ul style="list-style-type: none"> • setText() • getText() • setEnabled() • setToolTipText()

Class	Swing Control	Properties	Methods
JTextField	jTextField1	<ul style="list-style-type: none"> • Background • Font • Foreground • Text • Editable • Enabled • ToolTipText 	<ul style="list-style-type: none"> • setText() • getText() • setEditable() • setEnabled() • setToolTipText()
JRadioButton	jRadioButton1	<ul style="list-style-type: none"> • Background • ButtonGroup • Enabled • Font • Foreground • Label • Selected • Text 	<ul style="list-style-type: none"> • getText() • setText() • isSelected() • setSelected()
JCheckBox	jCheckBox1	<ul style="list-style-type: none"> • ButtonGroup • Font • Foreground • Label • Selected • Text 	<ul style="list-style-type: none"> • getText() • setText() • isSelected() • setSelected()
JComboBox	jComboBox1	<ul style="list-style-type: none"> • Background • ButtonGroup • Editable • Enabled 	<ul style="list-style-type: none"> • getSelectedItem() • getSelectedIndex() • setModel()
JList	jList	<ul style="list-style-type: none"> • Background • Enabled • Font • Foreground • Model • SelectedIndex • SelectedItem • SelectionMode • Text 	<ul style="list-style-type: none"> • getSelectedValue()
Button Group	JButton Group	<ul style="list-style-type: none"> • Add 	
JOptionPane			<ul style="list-style-type: none"> • ShowMessageDialog()
DefaultTable Model			<ul style="list-style-type: none"> • getRowCount() • removeRow() • addRow()

EXERCISE**Multiple Choice Questions:**

1. The set of instructions in a programming language is called _____.
 - (i) Data
 - (ii) Information
 - (iii) Process
 - (iv) Program
2. Java is originally known as _____.
 - (i) C
 - (ii) D
 - (iii) OAK
 - (iv) NETBEAN
3. Java Programming was designed by _____.
 - (i) Sun microsystem
 - (ii) Microsoft
 - (iii) TCS
 - (iv) wipro
4. Who initiated the Java language ?
 - (i) James Gosling
 - (ii) Bjarne
 - (iii) Larry Page
 - (iv) DCP
5. Java Source Code is compiled into _____.
 - (i) .obj
 - (ii) .exe
 - (iii) Source Code
 - (iv) Bytecode
6. Which of the tool is used to compile java code ?
 - (i) jar
 - (ii) javac
 - (iii) Java
 - (iv) javadoc
7. _____ is a program development environment for writing Java applets and applications.
 - (i) JDK
 - (ii) RAD
 - (iii) JRE
 - (iv) IDE
8. Java compiler is named as _____.
 - (i) java
 - (ii) Javac
 - (iii) JRE
 - (iv) jar

9. _____ provides a way to interact with the different components of JAVA through visual graphical elements (Pictures, icons, symbols, etc.).
- (i) IDE
 - (ii) NetBeans IDE
 - (iii) JAR
 - (iv) Interface
10. Which of the following statements about the Java language is true?
- (i) Java supports only Procedural Oriented Programming approach
 - (ii) Both Procedural and Object Oriented Programming are supported in Java
 - (iii) Java supports only Object Oriented Programming approach
 - (iv) None of the Above

Very Short Answer Type Questions

1. **Expand the the following abbreviations:-**
JRE, JVM, RAD, JDK, IDE, GUI, JFC, AWT
2. (a) Which window is used to design a form?
(b) Which window contains the Swing Controls components?
(c) What is the most suitable component to accept multiline text?
(d) What will be the output of the following command?
`Learning.concat("Java")`
(e) What will be the output of the following command?
`"Learning".concat("Java")`
(f) Name any two commonly used method of ListBox.
(g) What does `getpassword()` on a password field return ?
(h) What is event driven programming?
(i) Name the loop that never ends.
(j) Is Java case sensitive? What is meant by case sensitive?

3. Short Answer Type Questions

- (a) What is GUI programming?
- (b) Which window contains the Swing Controls components.
- (c) What is JVM ?
- (d) What is byte code?
- (e) How is Swing related to GUI programming?
- (f) What is container component?
- (g) Name some top level containers.
- (h) What is an event? What is event handler?
- (i) Name some properties of JFrame object.
- (j) Name some properties of JPanel object.
- (k) Write the difference between swing and AWT.
- (l) Which window of the IDE interface displays a hierarchical tree structure of all the components used in the current form?

4. Short Answer Type Questions

- (a) Briefly state the evolution of java?
- (b) How does java compile its program?
- (c) What is the need of JVM ?
- (d) Discuss briefly about JFC and its constituent APIs.
- (e) What are various categories of Swing controls?
- (f) Write steps to create a GUI application in java.
- (g) What are top-level containers? What is their significance?
- (h) Class can be used to define user-defined datatype. How?
- (i) What is dialog box ?define its type.

5. Long Answer Type Questions

- (a) Explain the characteristics of JAVA.
- (b) Discuss briefly about different layout managers offered by Swing API.
- (c) Explain the following terms:
 1. IDE
 2. Inspector Window
 3. Form
- (d) Explain the usage of the following methods with the help of an example:
 1. setText()
 2. toString()
 3. concat()
- (e) Explain the different tools of JDK.
- (f) Explain the NetBeans IDE.
- (g) Explain any six Component of swing control.

**Note :**

To install 'NetBeansIDE 8.1' in your PC, download the necessary software from www.oracle.com/technetwork/java/javase/downloads

and follow the link

`jdk-8u91-nb-8-1-windows-x64` for 64-bits systems

`jdk-8u91-nb-8-1-windows-i586` for 32-bits systems

CHAPTER - 5
PROGRAMMING FUNDAMENTALS

2. B PROGRAMMING FUNDAMENTALS

Introduction

Every programming language has some basic fundamentals which are very helpful for a user to write elementary programs. Some of the fundamental elements are Java Character Set, Java Tokens etc., which are helpful to write codes for Graphical User Interface application modules.

Character Set

A character set is a set of textual and graphic symbols, each of which is mapped to a set of non-negative integers. The first character set used in computing was US-ASCII. It is limited in that it can represent only American English. US-ASCII contains uppercase and lowercase latin alphabets, numerals, punctuations, a set of control codes, and a few miscellaneous symbols. But, Java uses the UNICODE Character Set.

Unicode defines a standardized, universal character set, used for representing characters and symbols as integers. Unlike ASCII, which uses 7 bits for each character, Unicode uses 16 bits, which means that it can represent more than 65,000 unique characters.

Java Tokens

Tokens are the Java program's elements which are identified by the compiler. A token is the smallest element of a program that is meaningful to the compiler. Tokens supported in Java includes; *keywords, identifiers, literals, punctuators, operators, etc.*

1. Keywords

Keywords are those reserved words that convey a special meaning to the compiler. There are 50 reserved keywords in Java programming language. These keywords have pre-defined functions that are usually highlighted with different colours in most of the IDEs.

Some Java keywords are as follows

abstract	assert	boolean	break	byte
case	catch	char	class	const
continue	default	do	double	else
enum	extends	final	finally	float
for	goto	if	implements	import
instanceof	int	interface	long	native
new	package	private	protected	public
return	short	static	strictfp	super
switch	synchronized	this	throw	throws
transient	try	void	volatile	while

2. Identifiers

An identifier is a name of fundamental building blocks of a program such as class, object, interface, method, variable etc.

Some of the rules to define a Java identifier are

1. Identifiers can contain alphabets,digits, underscore or dollar sign character.
2. They must not begin with a digit.
3. They can be of any length and contains upper-case as well as lower-case letters.
4. They cannot be a keyword, boolean literals or null character.

Identifiers should be meaningful, short enough to be quickly and easily typed and long enough to be descriptive and easily readable.

Following are some valid identifiers

Valid Identifier	Explanation
Student	Capital letter with small letters are allowed.
student	Small letters are allowed.
student1	Digits and underscore are allowed with alphabets
INT	Keywords are allowed but we have to change case of any letter or complete word.
_NUM	Underscore at the first position is allowed.
Sum_of_numbers	We can concatenate multiple words with underscore.
void	Keyword name cannot be given to identifier.
num'1	Special characters are not allowed.
x y	Spaces are not allowed.
2x	Digits are allowed but not as first character

Rules for Naming Identifiers

1. Name of instance variables and public methods should start with lowercase letter. *For example, age, total, percentage.*
2. When multiple words are used in a name, the second and subsequent words should start with an uppercase letter.
For example, collegeTeam, totalMarks.
3. Private and local variables use only lowercase letters together with underscores.
For example, class_exam.
4. All uppercase letters and underscores between words are used for constant values.
For example, S_MARKS, SALARY_INCR.
5. All classes and interfaces start with a leading uppercase letter.
For example, HelloJava MetroCity

3. Literals

Literals are those data items whose value does not change during the program execution. They are also known as **constants**.

Java supports different types of literals which are

- (a) Integer literal
- (b) Character literal
- (c) Floating-point literal
- (d) Boolean literal
- (e) String literal
- (f) Null literal.

(a) Integer Literals

These are the primary literals used in Java. They are of three types-

1. decimal (base 10),
 2. hexadecimal (base 16) and
 3. octal (base 8).
- (i) **Decimal Integer Literals**- Whose digits consists of the numbers 0 to 9.
 - (ii) **Hexadecimal Integer Literals**-Whose digits consists of the numbers 0 to 9 and letters A to F.
 - (iii) **Octal Integer Literals**- Whose digits consists of the numbers 0 to 7 only.

Normally, integer literals are stored in the 'int' type, which is signed 32-bit value but if you are working with very large number, it will be stored in 'long' -type with attached l or L at the end of the number. Long type is a signed 64-bit value.

Some rules for integer literals are given below

1. It must have at least one digit and can't use a decimal digit.
2. It must have a positive or negative sign, if the number does appear without any sign, it is assumed to be a positive number.
3. Hexadecimal literals appear with a leading 0x (zero, x). Octal literals appear with a leading 0 (zero) in front of its digits. While decimal literals appears as ordinary numbers with no special notation.

For example, an decimal literal for the number 10 is represented as 10 in decimal, 0xA in hexadecimal and 012 in octal.

(b) Character Literals

These literals represent a single unicode character and appear within a pair of single quotation marks. Like, 'a', '\$' and '1'.

There are some character literals which are not readily printable through a keyboard such as backspace, tabs, etc. These type of characters are represented by using escape sequences (\).

The table below shows the codes that can represent these special characters.

Escape Character	Meaning
\n	New line
\t	Tab
\b	Backspace
\r	Carriage return
\f	Form feed
\\	Backslash
\'	Single quotation
\"	Double quotation
\ud	Unicode character
\d	Octal character
\xd	Hexadecimal character

where, the letter d such as in octal, hexa etc., represents a number.

(c) Floating-Point Literals

Floating-point numbers are like real numbers in mathematics. *For example,* 4.13179, -0.0001. Java has two kinds of floating-point number: **float** and **double**. The default type when you write a floating point literal is double. Float is of 32 bits, whereas double is of 64 bits.

A floating-point literal can be denoted as a **decimal point**, a **fraction part**, an **exponent** (represented by E or e) followed by an optionally signed integer. We also add a suffix to the floating point literal as D, d, F or f (D or d for double and F or f for float).

The following floating-point literals represent double precision floating-point values

☞	6.5E+32 (or 6.5E32)	Double-precision floating-point literal
☞	7D	Double-precision floating-point literal.
☞	.01f	Floating-point literal.

The following table represents the requirement of each part of the floating point literals:-

Part	Is It Required?	Examples
Whole number	Not, if fractional part is present	0,5,100,1005
Decimal Point	Not, if exponent is present; must be there if there is a fractional part.	e 23, 6.5E32, 6.5
Fractional	Cannot be present if there is no decimal point; must be there if there is no whole number part.	0, 3, 1415927, 0.5, 1.6, 7.5
Exponent	Only if there is no decimal point.	2 43, E-17, E5, e+304, e-2
Type suffix	No. The number is assumed to be double precision in the absence of a type suffix.	7D, .01f

(d) Boolean Literals

There are two Boolean literals true and false. True represents a true value and false represents a false value. Unlike C, we can't assume that the value of 1 is equivalent to true and value of 0 is equivalent to false. We have to use the values true and false to represent a Boolean value. Like,

```
Boolean select = true;
```

Literals true or false should not be represented by the quotation marks around it. Java compiler will take it as a string of characters, if it is represented in quotation marks.

(e) String Literals

It is a sequence of characters between a pair of double quotes.

For example, “Hello Java” “result = 0.01”

(f) Null Literal

It is a special Java literal used to represent null value. Null literal in a source code is specified as ‘null’. Generally, null literal is assigned to object reference variables.

For example, A=null ;

In the above example, A is an object. Now, null is assigned to A and this will reduce the number of references to object A.

As, the object is no longer referenced so it will be available for the garbage collection i.e., the garbage collector will destroy it and free memory will be allocated to any other object.

4. Punctuators (Separators)

A punctuator is a type of token that has syntactic and semantic meaning to the compiler, but the exact meaning depends upon the context where we use it.

Punctuators are used for grouping and separating the numeric and non-numeric data. Some of the Punctuators used in Java are

Name	Use
())Parameter in method definition, contains statements for condition etc.
{}	Used to define a code for method and classes.
[]	Used for array declaration.
;	Used to show the separate statement
,	Used to show the separation in identifier in variable declaration
.	Used to show the separate package name from sub packages and classes. Separate variable and method from reference variable.

For example; int A, B = 10; Here , (comma) is a punctuation.

5. Operators

Operators are those tokens that perform a specific task/computation when applied on variable or some o objects in an expression (known as **operands**).

Operators are used in programs to operate on data and variables. Java carries a wide range of operators

They are

1. Arithmetic operators
2. Logical operators
3. Relational operators
4. Assignment operators
5. Conditional operators
6. Increment and Decrement operators
7. Bitwise operators.

1. Arithmetic Operators

These operators are used in mathematical expressions. These operators can operate on built-in data types of Java.

Following table lists the arithmetic operators Consider two variables A and B. Let the value of A is 5 and value of B is 10.

Operator	Name	Description	Example
+	Addition	It adds the values of either sided operands	$A + B = 15$
-	Subtraction	Subtracts the value of right handed operand from left hand operand's value.	$A - B = -5$
*	Multiplication	Multiplies the value of either sided operands.	$A * B = 50$
/	Division	Divides left hand operand by right hand operand.	$B / A = 2$
%	Modulus	Divides left hand operand by right hand operand and returns remainder.	$B \% A = 0$

2. Relational operators

Relational operators compare two operands to one another. The relational operators are `=`, `!=`, `<`, `>`, `<=` and `>=`. When used in an expression, they all return a boolean value which states the result of comparison. Relational operators are sometimes called comparison operators and expressions that contain relational operators are called relational expressions.

Following table shows all the relational operators supported by Java language. Assuming variable **A** holds 10 and variable **B** holds 15 then

Operator	Name	Description	Example
<code>=</code>	(Equality)	Checks if the values of two operands are equal or not, if yes then condition becomes true.	<code>(A = B)</code> is not true
<code>!=</code>	(Not equal to)	Checks if the values of two operands are equal or not, if values are not equal then condition becomes true.	<code>(A! = B)</code> is true
<code>></code>	(Greater than)	Checks if the value of left operand is greater than the value of right operand, if yes then condition becomes true.	<code>(A > B)</code> is not true.
<code><</code>	(Less than)	Checks if the value of left operand is less than the value of right operand, if yes then condition becomes true.	<code>(A < B)</code> is true.
<code>>=</code>	(Greater than or equal to)	Checks if the value of left operand is greater than or equal to the value of right operand, if yes then condition becomes true.	<code>(A >= B)</code> is not true.
<code><=</code>	(Less than or equal to)	Checks if the value of left operand is less than or equal to the value of right operand, if yes then condition becomes true.	<code>(A <= B)</code> is true.

3. Logical Operators

These operators are used to make a decision on two conditions. Logical operators are typically used with boolean values and what they return is also a boolean value. Logical operators are mainly used to control program flow. Usually, you will find them as part of an if, a while, or some other control statements.

Following tables shows all the logical operators supported by Java language. (assuming variable A holds 5, B holds 7, and C holds 9, then)

Operator	Name	Description	Example
&&	Logical AND	Returns TRUE if both the conditions are true; FALSE otherwise	((A<B)&&(C<B)) returns FALSE
	Logical OR	Returns TRUE if any of the two conditions is true	((A<B) C<B)) returns TRUE
!	Logical NOT	Works on single operand and reverses its state	!((A<B)&&(C<B)) returns TRUE

4.

Conditional Operator

Conditional operator is used when one expression and two statements are given, it is a ternary operator. The syntax of this operator is like,

Syntax

```
exp1 ? expression2 : expression3;
```

Where, exp 1 is an expression and expression1 and expression2 are statements.

Firstly, exp 1 is tested and if it is true then expression2 is evaluated and becomes the value of the conditional expression, otherwise, expression3 is evaluated and it becomes the value of the conditional expression.

For example,

```
a=3; b=4; c=(a<b)? a:b;
```

```
c=(3<4)? 3:4; c=3
```

In above example, the compiler will firstly evaluate the condition (a < b) i.e., (3 < 4), which is true. So, it will return 3.

Nested Conditional Operator The conditional operator can be nested.

For example, let you have three numbers a, b, c. If you want to find smallest number among these numbers, then the conditional operator will be used like.


```
int a, b, c, min;
    min=((a<b)?((a<c)?a:c):((b<c)?b:c));
    Let a = 5, b = 3, c = 4;
    min = ((5 < 3)? ((5 < 4)? 5 : 4) : ((3 < 4)? 3 : 4));
```

The compiler will evaluate the first condition (5 < 3), it is false. So, the control will pass to the conditional operator ((3 < 4)? 3 : 4) and it will be evaluated. It is true. So it will return the value 3.

5. Increment and Decrement Operator

The increment operator is used to increase some value by 1. The decrement operator is used to decrease some value by 1. Both are unary operators. Usually these operators are used in, for and while loops.

Increment operator ++

Decrement operator — —

The following table shows the exact working of increment and decrement operators

Expression		Example	Result
A++	Process	A = 2, B; B = A++;	A=3 B=2
	Add 1 to variable after use	int A = 2, B; B = ++A;	A=3 B=3
++A	Add 1 to the variable before use	int A = 2, B; B = A — —;	A = 1 B=2
A — —	Subtract 1 from a variable	int A = 2, B; B = — — A;	A = 1 B=1

after use

6. — —A

Subtract 1 from a variable
before use

Bitwise Operators

A bitwise operator works on bits and perform bit-by-bit operation. These operators manipulate the values of data at bit level. These operators are generally used for testing and shifting of bits.

For example, if $A = 60$ and $B = 13$. Now in binary format they will be as follows

$A = 0011\ 1100$

$B = 0000\ 1101$

A table is given to understand bitwise operators more precisely

Operator	Name	Description	Example
&	bitwise AND	binary AND operator copies a bit to the result, if it exists in both operands	$(A \& B)$ gives 12, which is 0000 1100
	bitwise OR	binary OR operator copies a bit, if it exists in either operands	$(A B)$ gives 61, which is 0011 1101
^	bitwise XOR	binary XOR operator copies the bit, if it is set in one operand but not both	$(A \wedge B)$ gives 49, which is 0011 0001
~	bitwise Complement	binary 1's complement operator is unary, flips the bit	$(\sim A)$ gives -61, which is 1100 0011
<<	Left Shift	binary Left Shift operator. The left operand valuee gets moved left by the number of bits specified by the right operand.	$A \ll 2$ gives 240, which is 1111 0000
>>	Right Shift	binary Right Shift operator. The left operand valuee gets moved right by the number of bits specified by the right operand.	$A \gg 2$ gives 15, which is 1111
>>>	Right Shift	Shift Right Zero Fill operator. The left operand valuee gets moved right by the number of bits specified by the right operand and shifted values gets filled up with zeros	$A \ggg 2$ gives 15, which is 00001111

7. Assignment Operators:

These operators are also known as Shorthand Operators. These operators are used to assign the value of an expression to a variable. The following table lists the assignment operators

Operator	Name	Description	Example
=	Assignment operator	assigns values from right side operands to left side operand	$C = A + B$; assign value of $A+B$ to C
+=	Add and Assignment operator	adds right operand to the left and assigns the result to the left operand	$C += A$ is equivalent to $C = C + A$

<code>-=</code>	Subtract and Assignment operator	subtracts right operand from the left and assigns the result to the left operand	$C - = A$ is equivalent to $C = C - A$
<code>*=</code>	Multiply and Assignment operator	multiplies right operand with the left and assigns the result to the left operand	$C * = A$ is equivalent to $C = C * A$
<code>/=</code>	Divide and Assignment operator	divides left operand with the right and assigns the result to the left operand	$C / = A$ is equivalent to $C = C / A$
<code>%=</code>	Modulus and Assignment operator	takes modulus using two operands and assigns the result to the left operand	$C \% = A$ is equivalent to $C = C \% A$

Unary, binary and ternary operators in Java

Unary Operator These are the operators which work on single operands.

For example, `!`, `-`, `++`, `—`, `()`, (cast) operator, Unary `+` and Unary`(—`

Unary, binary and ternary operators in Java

Unary Operator These are the operators which work on single operands.

For example, `!`, `-`, `++`, `—`, `()`, (cast) operator, Unary `+` and Unary`(—` are some examples of unary operators.

Binary Operators These are the mostly used operators. These operators work on two operands. Binary operators include arithmetic operators (`+`, `*`, `/`, `%` etc.).

Ternary Operator Operator that works on three operands is known as ternary operator.

Conditional operator (`? :`) is the example of ternary operator.

Precedence of Java Operators:

When in an expression more than one operator are present then to decide which operator will be applied first and which one will be the next, we use precedence and associativity. Precedence is the order in which operators are applied. Some operators have higher precedence while some others have lower precedence.

Precedence order.

When two operators share an operand the operator with the higher precedence goes first. For example, $1 + 2 * 3$ is treated as $1 + (2 * 3)$, whereas $1 * 2 + 3$ is treated as $(1 * 2) + 3$ since multiplication has a higher precedence than addition.

Associativity.

When an expression has two operators with the same precedence, the expression is evaluated according to its associativity.

For example $x = y = z = 17$ is treated as $x = (y = (z = 17))$, leaving all three variables with the value 17, since the $=$ operator has right-to-left associativity (and an assignment statement evaluates to the value on the right hand side). On the other hand, $72 / 2 / 3$ is treated as $(72 / 2) / 3$ since the $/$ operator has left-to-right associativity.

A table of operators is given below according to their precedence from highest to lowest

Precedence	Operator	Description	Associativity
1	() [] .	Parentheses Array subscript Access object method	Left to Right
2	++ -- + - ! ~	Postfix Prefix increment Postfix Prefix decrement Unary plus Unary minus Logical NOT Bitwise NOT	Right to left
3	(type)	Unary type cast	Right to left
4	*	Multiplication	Left to right
5	/	Division	Left to right
	%	Modulus	
	+ - +	Addition Subtraction String Concatenation	
6	<< >>	Bitwise left shift Bitwise right shift with sign extension	Left to right

	>>>	Bitwise right shift with zero extension	
7	< <= > >= instance of	Relational less than Relational less than or equal Relational greater than Relational greater than or equal Type comparison (objects only)	Left to right
8	== !=	Relational is equal to Relational is not equal to	Left to right
7	&	Bitwise AND	Left to right
8	^	Bitwise exclusive OR	Left to right
9		Bitwise inclusive OR	Left to right
10	&&	Logical AND	Left to right
11		Logical OR	Left to right
12	?:	Ternary conditional	Right to left
13	= += -= *= /=	Assignment Addition assignment Subtraction assignment Multiplication assignment Division assignment Modulus assignment	Right to left

DATA TYPES

Data type describes the type of data that a variable can store along with the set of operations performed on it. Java supports various kinds of data types such as, integer, character, float etc.

Java facilitates to handle different types of data types as given below:

1. Primitive / Built-in

- (i) Numeric
- (ii) Fractional
- (iii) Character
- (iv) Boolean

2. Referenced

- (i) User Defined
- (ii) Derived
 - (a) Classes
 - (b) Arrays
 - (c) Interfaces

PRIMITIVE DATA TYPES

These are the basic/core data types that are not derived from any other types. These are predefined and are also known as built-in data types.

Built-in data types are further classified into four categories as mentioned above.

(i) Numeric Data Type

Those primitive data types which are used to store only integer values are known as numeric data types. Numeric data types are further divided into four parts.

Byte: It is the smallest numeric data type. It is a 8-bit signed two's complement integers that has a range from -128 to +127. The byte data type is useful for saving memory in large arrays. It is declared by the use of byte keyword. Byte is four times smaller than an integer.

For example,

```
byte y;
```

Here, y is a variable of byte data type.

Short: It is the least used numeric data type. It is a 16-bit signed two's complement integers that has a range from -32768 to +32767. It is declared by the use of short keyword. Short is two times smaller than an integer.

For example,

```
short y;
```

Here, y is a variable of short data type.

Int : It is the most commonly used data type. It is a 32-bit signed integers that has a range from

-2147483648 to +2147483647.

It is declared by the use of **int** keyword.

For example, `int y;`

Here, y is a variable of int data type.

Long: It is a 64-bit signed integers that has a range from -2^{63} to $2^{63} - 1$ ($-9.223372e18$ to $+9.223372e18$.) This data type is used where an integer type variable is not large enough to hold the desired value. It is declared by the use of long keyword.

For example, `long x;`

Here, x is a variable of long data type.

The storage size, range and default values of different numeric data types are listed below in the table

Sl.No.	Data Type	Size	Range	Default value
1	Byte	1 byte	+ 127 to -128	0
2	Short	2 byte	+ 32767 to -32768	0
3	Int	4 byte	+ 2147483648 to 2147483647	0
4	Long	8 byte	-263 to 263-1	0

(ii) Fractional Data Type

Those primitive data types used to store floating point numbers are known as fractional data types. Floating point numbers are also known as real numbers. There are two types of fractional data types

Float: It holds the floating point numbers. It specifies a single precision value that uses 32 bits of storage. It is declared by the use of float keyword.

For example, `float y;`

Here, y is a variable of float type.

Double: It is an extended version of float data type. It uses 64 bits to storage. All complex math functions, such as `sin()`, `sqrt()` return double values. It is declared by the use of double keyword. For iterative calculations, double is the best choice.

For example, `double x;`

Here, x is a variable of double data type.

Sl. No.	Data Type	Size	Range	Number of decimal places
1	Float	4 byte	+2147483647 to -2147483648	8
2	Double	8 byte	approximately \pm 1.79769313486231570E+308	16

(iii) Character Data Type

The data type used to store characters known as character data type. Java uses unicode to represent characters. The size of character data type is 16-bits and ranges from 0 to 65535.

It is declared by the use of char keyword. Character data type does not contain negative characters

For example, `char name;`

Here, name is a variable of char data type.

(iv) Boolean Data Type

Boolean data type is used to represent logical values. It can have only one of the two possible values, i.e., true or false. In Java boolean data type reserved 8-bits storage space but uses only 1-bit. It is declared by the use of boolean data type.

For example, `boolean x = true;`

Here x is a variable of boolean data type.

REFERENCED DATA TYPES

These are the non-primitive data types which are based on built-in data types. These data types are constructed by using primitive data types, as per user need. There are two types of referenced data types.

1. Derived data types

Derived data types are those whose variables allow us to store multiple values of same type. But they never allow storing multiple values of different types. These are the data type whose variable can hold more than one value of similar type. In general derived data type can be achieving using array.

For example, `int a[] = {10,20,30}; // valid`

`int b[] = {100, 'A', "ABC"}; // invalid`

Derived data types store the memory address of an object. Class, Interface and Array are the examples of derived data types.

2. User-Defined data types

User-Defined data types are those which are developed by programmers by making use of appropriate features of the language. User defined data types related variables allows us to store multiple values either of same type or different type or both. This is a data type whose variable can hold more than one value of dissimilar type, in java it is achieved using class concept. In C language, user defined data types can be developed by using struct, union, enum etc. In java programming user defined data type can be developed by using the features of classes and interfaces.

For example, `Student s = new Student ();`

Note: In Java, both derived and user defined data types are combined and named as reference data type.

STRING in Java

Strings are the sequence of characters. In Java, strings are taken as objects not data-type. Java provides Strings class to create and manipulate Strings. This class represents character strings. **Declaring and creating strings** In Java, strings are class objects and implemented using two classes, String and StringBuffer.

Strings can be declared and created as follows

```
String stringname,  
stringName = newString("string");
```

For example, a string "Information Technology" is declared below

```
String str;  
str = newString("Information Technology");
```

Whenever the compiler encounters a string in the code, it will create a String object whith its value "Information Technology".

VARIABLE

It is the basic unit of storage in a Java program or we can say that, variable is a container object that stores a meaningful. Variable is an identifier which

holds data or another one. Variable is an identifier whose value can be changed at the execution time of program. Variable is an identifier which can be used to identify input data in a program.

In a Java program, to store different types, of values such as, name of a shop and quantities of items (goods) available in the shop during the run time, we require some sort of storage area called variables. Each variable has a specific type, which determines the size and layout of the variable's memory, the range of values that can be stored within that memory and the set of operations that can be applied to the variable.

Declaring a variable:

In Java, all of the variables must be declared before they used in a Java program. A variable is declared in a program by providing the data type of a variable and the identifier name/variable name that identifies the variables. The basic structure of a variable declaration is as follows

Data_ type variable_name;

For example:

- (1) To declare a character variable.

Syntax: char ch;

- (2) To declare an integer variable.

Syntax: int x;

- (3) To declare a floating-point variable.

Syntax: float a;

Variable Naming

Naming convention defines a set of rules and regulations to decide the name of the variable you want to use in a Java program.

Some of the rules for naming a variable are

1. Variable names are case sensitive.
2. No spacing is allowed within variable names.
3. No special symbols can be used in variable names such as !, 0, #, %, A, * etc.
4. A variable name does not start with a numeric value or the underscore () symbol.

5. Variable names must not be a keyword or reserved word.
6. Variable names can only contain characters, numbers and the underscore (_) symbol.
7. Every variable name should start with an alphabet.

Examples of some invalid variable names;

- 123 Æ Number is not allowed.
- (area) Æ Parenthesis is not allowed in a variable.
- 6th Æ First character cannot start in digit.
- %abc Æ Special symbols are not allowed.

Assigning Values to Variables

A value is assigned to a variable after it has been declared but before it is used in an expression. Assigning a value to a variable is known as initialization of variables. Initialization is important because, uninitialized data has been a common source of bugs. To initialize a variable we use an assignment operator (=).

The General form of Initializations is

```
Datatype variable_name =constant;
```

Example:

```
float salary = 54000;  
char salute = 'Namaste';  
int n = 54;
```

Note that, the *char values* are within single quotation marks, but not the *numeric values*! More than one variable can also be declared and intialized at the same time such as

```
int a=3, b =8, c=12;
```

CONSTANT in Java

There are several values which never get changed. For example, a day will always have 24 hours, the value of PI (up to three decimal places) will always be 3.141. These are fixed values and always remain constant. In context of programming, it is convenient to represent these values in same way (declare them as constant). These variables are known as constant.

In Java, variables are declared constant by the use of final keyword. So, that the value of that variable get fixed in the program. Final is a reserved keyword and tells the compiler that the value will remain unchanged.

For example, `int hours = 24;`

Here, we know this value will remain unchanged as a day always has 24 hours so the final keyword can be used.

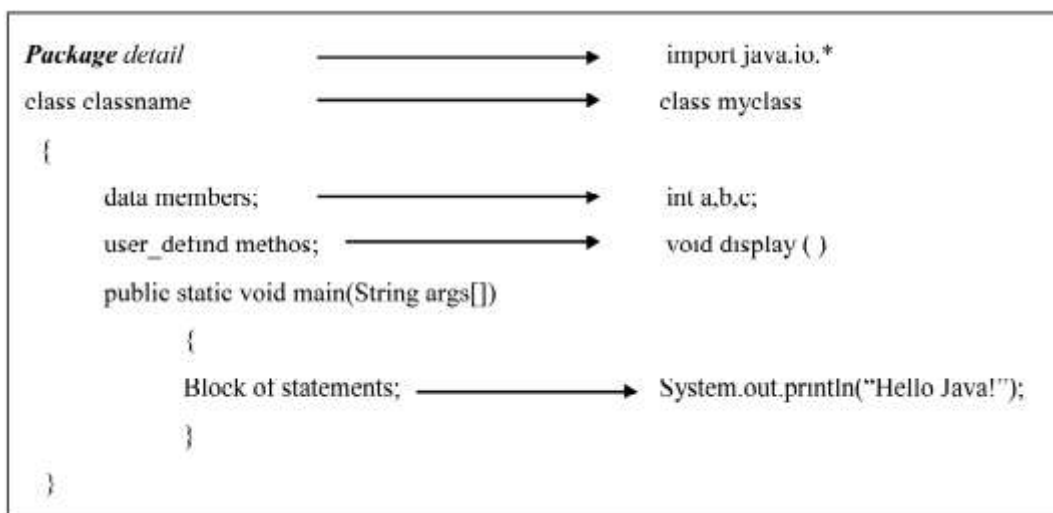
`final int hours = 24;`

Advantages of using Constants

1. By making variables final, the values of variables can never be changed accidentally.
2. You have to declare them only once in a program.
3. If you want to change their value, you have to change it only at one place (at the time of declaration).

Structure of Java Program

Structure of a java program is the standard format released by Language developer to the Industry programmer. Sun Micro System has prescribed the following structure for the java programmers for developing java application. ?



- ☞ A package is a collection of classes, interfaces and sub-packages. A sub package contains collection of classes, interfaces and sub-sub packages etc. `java.lang.*`; package is imported by default and this package is known as default package.
- ☞ Class is keyword used for developing user defined data type and every java program must start with a concept of class.
- ☞ “ClassName” represent a java valid variable name treated as a name of the class each and every class name in java is treated as user-defined data type.
- ☞ Data member represents either instance or static they will be selected based on the name of the class.
- ☞ User-defined methods represents either instance or static they are meant for performing the operations either once or each and every time.
- ☞ Each and every java program starts execution from the `main()` method. And hence `main()` method is known as program driver.
- ☞ Since `main()` method of java is not returning any value and hence its return type must be void.
- ☞ Since `main()` method of java executes only once throughout the java program execution and hence its nature must be static.
- ☞ Since `main()` method must be accessed by every java programmer and hence whose access specifier must be public.
- ☞ Each and every `main()` method of java must take array of objects of String.
- ☞ Block of statements represents set of executable statements which are in term calling user-defined methods are containing business-logic.
- ☞ The file naming convention in the java programming is that which-ever class is containing `main()` method, that class name must be given as a file name with an extension `.java`.

Main () Method

main () method is starting execution block of a java program or any java program start their execution from main method. If any class contain `main()` method known as main class.

Syntax:

```
public static void main(String args[])
{
.....;
.....;
}
```

Public

public is a keyword in a java language whenever if it is preceded by main() method the scope is available anywhere in the java environment that means main() method can be executed from anywhere. main() method must be accessed by every java programmer and hence whose access specifier must be public.

Static

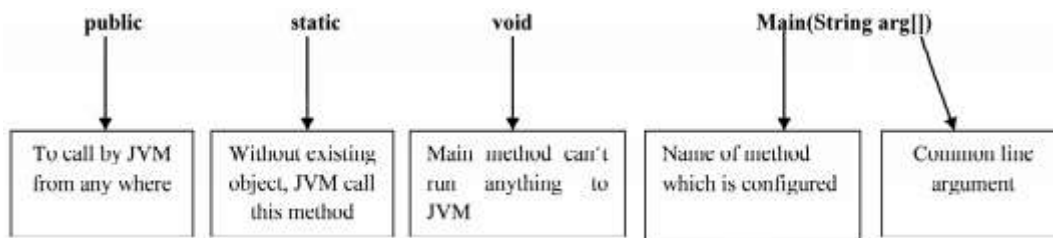
static is a keyword in java if it is preceded by any class properties for that memory is allocated only once in the program. Static method are executed only once in the program. main() method of java executes only once throughout the java program execution and hence it declare must be static.

Void

void is a special datatype also known as no return type, whenever it is preceded by main() method that will be never return any value to the operating system. main() method of java is not returning any value and hence its return type must be void.

String args[]

String args[] is a string array used to hold command line arguments in the form of String values.



INTEGER OBJECT METHOD

Integer is a wrapper class that holds an int variable. The Integer class includes both instance methods and class methods.

parseInt()

parseInt() method of Integer class is used to convert string to integer. It is a static method. This returns an integer (decimal only).

Syntax :

```
return_type data_type method_name (datatype variable)
static int parseInt (string S)
```

For example,

```
public class example {
    public static void main(String args[]) {
        int x =Integer.parseInt("123");
        System.out.println(x);
    }
}
```

Output

```
123
```

DOUBLE OBJECT METHOD

Double is a wrapper class that holds a double variable. The Double class includes both the instance methods and class methods.

parseDouble() method

parseDouble() method of Double class is used to return a double type value for the value represented by a specified string. It converts a string value to a double type value.

Syntax : static double parseDouble(String s)

For example,

```
double a = Double.parseDouble("123");
public class example {
```

```
public static void main(String args[]) {  
    double x= Double.parseDouble("123");  
    System.out.println(x);  
    }  
}
```

Output

123

parseFloat () method:

parseFloat () method of float class is used to convert string to floating-point type value.

Syntax : static double parse Float (String s)

For example,

```
float x = Float.parseFloat("5");
```

Sample program to show Integer and Double object methods

```
class example{  
public static void main(String arg[]){  
    int x;  
    float y;  
    double z;  
    x = Integer.parseInt(arg[10]);  
    y = Float.parseFloat(arg[20]);  
    z = Double.parseDouble(arg[30]);  
    System.out.println("Integer Number =" +x);  
    System.out.println("Float Number =" +y);  
    System.out.println("Float Number =" +z);  
    }  
}
```


CONTROL STRUCTURE

A control structure determines an order in which the statements used in a program are executed. A Java program contains classes and these classes have some methods which contains statements that are executed by the compiler. A control structure might cause a statement to be executed once, many times, or not at all. The following control statements are

1. Decision Structure
2. Looping Structure

Decision Structure

Decision making structures have one or more conditions to be evaluated or tested by the program, along with a statement or statements that are to be executed if the condition is determined to be true, and optionally, other statements to be executed if the condition is determined to be false. Decision structures are also known as conditional statements or selection statements. It allows a user to choose any specified condition/statement from a slot of statements.

Java programming language provides following types of decision making statements.

1. **If** structure
2. **If..... else** structure
3. **Switch** structure

If structure

It is the most basic form of selection statement which executes a single block of statements if the specified condition evaluates to true, otherwise the given set of actions/ statements are ignored.

Syntax:

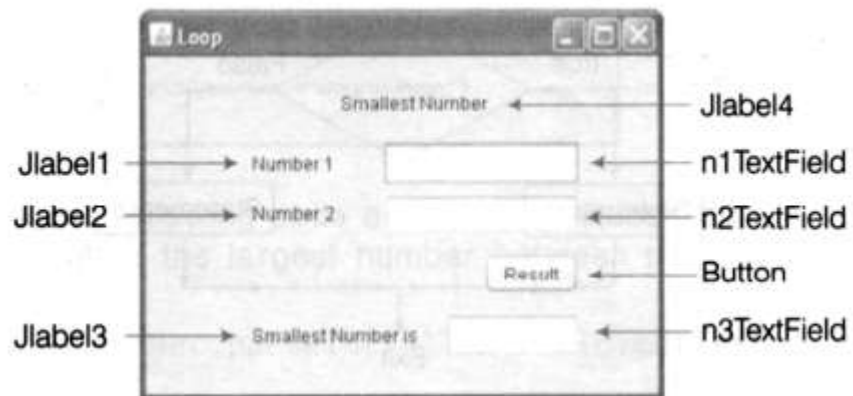
The syntax of an *if statement* is:

```
If (expression)
{
    //Statements will execute if the expression is true
}
```

If the expression evaluates to true then the block of code inside the if statement will be executed otherwise skip the statement.

Example:

Write a code for a GUI application using NetBeans IDE to find the smallest number between two numbers (take input from user).

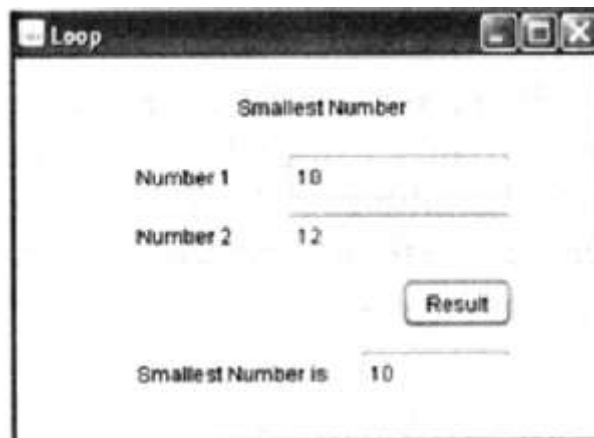


Type the following code and click on Resultbutton to calculate the smallest number.

```
int a= Integer.parseInt(n1TextField.getText());
int b =Integer.parseInt(n2TextField.getText());
int smallest= 0;
if(a < b)
{
    smallest= a;
```

```
n3TextField.setText (" "+ smallest);  
}
```

Output of the above code.



If-else statement

It is the type of selection statement in which a single statement or a group of statements enclosed within the if condition is executed when if is true. Otherwise the else block is executed.

Syntax

```
If(expression){  
    //Executes when the expression is true  
}else{  
    //Executes when the expression is false  
}
```

If the expression evaluates to true, then the if block of code will be executed, otherwise else block of code will be executed.

Example

Write a code for a GUI application using NetBeans IDE to find the less number of students between two different classes of a college.

Code

```
int a=Integer.parseInt(n1TextField. getText()):
int b=Integer.parseInt(n2TextField. getText ()):
int less=0:
if(a < b)
I
less=a:
n3textfield.setText(" "+ less);
}
else
{
less=b:
n3Textfi el d. setText (" " + less):
}
```

Switch Structure

A **switch** statement allows a variable to be tested for equality against a list of values. Each value is called a case, and the variable being switched on is checked for each case.

Syntax:

```
switch(expression){
case value1 :
    //Statements;
break; //optional
case value2 :
    //Statements;
```

```
break; //optional
.....
.....
.....
case value N :
    //Statements N;
break; //optional

    //You can have any number of case statements.
default : //Optional
    //Statements;
}
```

The following rules apply to a **switch** statement:

- ☞ The variable used in a switch statement can only be integers, convertible integers (byte, short, char), strings and enums
- ☞ You can have any number of case statements within a switch. Each case is followed by the value to be compared to and a colon.
- ☞ The value for a case must be the same data type as the variable in the switch and it must be a constant or a literal.
- ☞ When the variable being switched on is equal to a case, the statements following that case will execute until a *break* statement is reached.
- ☞ When a *break* statement is reached, the switch terminates, and the flow of control jumps to the next line following the switch statement.
- ☞ Not every case needs to contain a break. If no break appears, the flow of control will *fall through* to subsequent cases until a break is reached.
- ☞ A *switch* statement can have an optional default case, which must appear at the end of the switch. The default case can be used for performing a task when none of the cases is true. No break is needed in the default case.

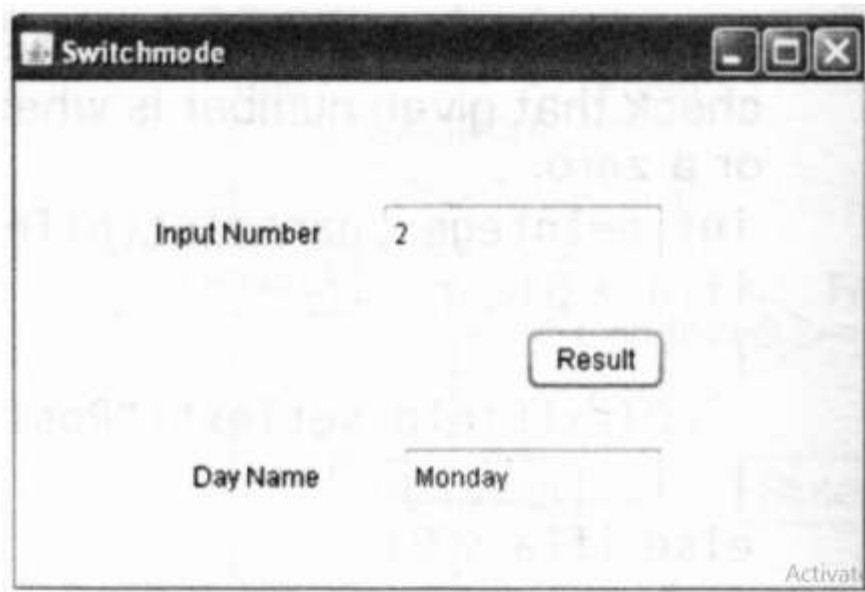
Example

Write a code for a GUI application using NetBeans IDE to display the name of the day corresponding to an input number.

Type the following code in the Result Button

```
int d=Integer.parseInt(n1TextField.getText());
switch(d)
{
case 1:
    n2TextField.setText(" Sunday ");
break;
case 2:
    n2TextField.setText(" Monday");
break;
case 3:
    n2TextField.setText("Tuesday");
break;
case 4:
    n2TextField.setText("Wednesday");
break;
case 5:
    n2TextField.setText("Thursday");
break;
case 6:
    n2TextField.setText("Friday");
break;
case 7:
    n2TextField.setText("Saturday")
break;
default:
    n2TextField.setText("Wrong choice")
break;
}
```

Output of the above code is



Looping Structure

There may be a situation when you need to execute a block of code several number of times. In general, statements are executed sequentially: The first statement in a function is executed first, followed by the second, and so on.

Programming languages provide various control structures that allow for more complicated execution paths.

A **loop** statement allows us to execute a statement or group of statements multiple times and following is the general form of a loop statement in most of the programming languages:

Java programming language provides the following types of loop to handle looping requirements.

1. Do.....While loop
2. While loop
3. For loop

Do.....While 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:

```
do
{
    //Statements
} while (Boolean_expression);
```

The expression appears at the end of the loop, so the statements in the loop execute once before the Boolean is tested.

If the expression is true, the control jumps back up to do statement, and the statements in the loop execute again. This process repeats until the Boolean expression is false.

Example

```
public class natural{
public static void main(String args[]){
int x = 0;
do{
System.out.print("value of x : " + x );
x++;
System.out.print("\n");
}while( x <10 );
}
}
```

The output of the above code is :

```
value of x : 0
value of x : 1
value of x :2
value of x : 3
value of x : 4
value of x : 5
value of x : 6
value of x : 7
value of x : 8
value of x : 9
```


While loop

A **while** loop statement in java programming language repeatedly executes a target statement as long as a given condition is true.

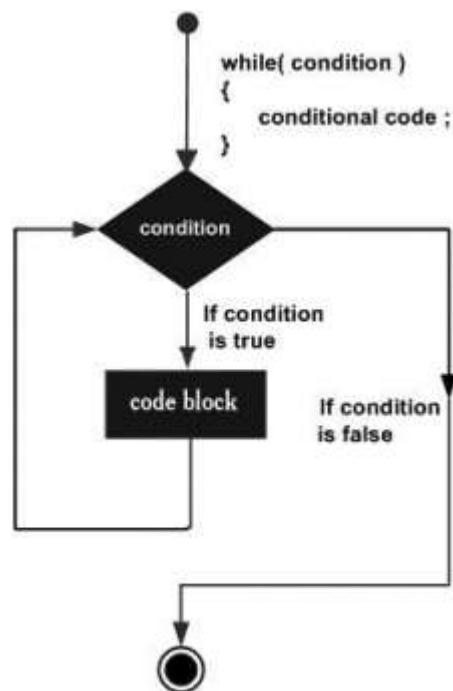
Syntax:

```
while(Boolean_expression)
{
    //Statements
}
```

Here, **statement(s)** may be a single statement or a block of statements. The **condition** may be any expression, and true is any non zero value.

When executing, if the *boolean_expression* result is true, then the actions inside the loop will be executed. This will continue as long as the expression result is true.

When the condition becomes false, program control passes to the line immediately following the loop



Example:

```
public class natural {
    public static void main(String args[]) {
        int x = 10;

        while( x < 20 ) {
            System.out.print("value of x : " + x );
            x++;
            System.out.print("\n");
        }
    }
}
```

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
```

For loop

A **for** loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times. A for loop is useful when you know how many times a task is to be repeated.

Syntax:

```
for(initialization; Boolean_expression; update)
{
    //Statements
}
```

Here is the flow of control in a for loop:

- Ø The **initialization** step is executed first, and only once. This step allows you to declare and initialize any loop control variables. and this step ends with a semi colon (;)
- Ø Next, the **Boolean expression** is evaluated. If it is true, the body of the loop is executed. If it is false, the body of the loop will not be executed and control jumps to the next statement past the for loop.
- Ø After the **body** of the for loop gets executed, the control jumps back up to the update statement. This statement allows you to update any loop control variables. This statement can be left blank with a semicolon at the end.
- Ø The Boolean expression is now evaluated again. If it is true, the loop executes and the process repeats (body of loop, then update step, then Boolean expression). After the Boolean expression is false, the for loop terminates.

Example:

```
public class example {
    public static void main(String args[]) {
        for(int x = 100; x <110; x = x+1) {
            System.out.print("value of x : " + x );
            System.out.print("\n");
        }
    }
}
```

This would produce the following result:

value of x : 100

value of x : 101

value of x : 102

value of x : 103

value of x : 104

value of x : 105

value of x : 106

value of x : 107

value of x : 108

value of x : 109

EXERCISE**1. Multiple Choice Question**

- (a) JVM is a/an _____
- (i) compiler (ii) an interpreter
(iii) assembler (iv) Debugger
- (b) Is Java called as platform independent language?
- (i) Yes (ii) No
- (c) Which of the following is smallest integer data type ?
- (i) int (ii) short
(iii) byte (iv) long
- (d) Which of the following is not a primitive data type ?
- (i) int (ii) short
(iii) byte (iv) enum
- (e) What is the size of int data type in java?
- (i) 1 bytes (ii) 2 bytes
(iii) 4 bytes (iv) 8 bytes
- (f) Which is a valid float literal?
- (i) 1.23 (ii) 2
(iii) 1.23d (iv) 1.23f
- (g) Which of these values can a boolean variable contain?
- (i) True & False (ii) 0 & 1
(iii) Any integer value (iv) Both a & b
- (h) Which of these are selection statements in Java?
- (i) if (ii) for
(iii) continue (iv) all of these
- (i) Which of these is incorrect string literal?
- (i) Hello World (ii) Hello\n World
(iii) \ Hello World (iv) Hello
- (j) What is the default value of Boolean variable?
- (i) true (ii) false
(iii) null (iv) not defined

2. Very short Answer Type Question

- (b) What is variables?
- (c) What is meant by token ? Name the token available in java.
- (d) What is identifier ?
- (e) What is loop?

- (f) What is initialization expression?
- (g) What is the difference between while and do—while loop?
- (h) Write a conditional statement to check the value of character variable var as V or y.
- (i) Write a conditional statement to check the value of variable num lying in the range of numbers 1500 and 2200.
- (j) Write a conditional statement to check whether the sum of a and b is greater than double of b.
- (k) Write a conditional statement to check whether the sum of a and double of b is greater than product of a and b and the difference of a and b is less
- (a) What does the token 'keyword' refer to, in the context of Java? Give an example for keyword.
- (b) Explain with an example the *if-else-if* construct.
- (c) State the difference between = and ==.
- (d) Differentiate between *if* and *switch* statements.
- (e) What is a sequence construct ?
- (f) What are the control statements available in java ?
- (g) Write the general format (or syntax) of if-statement.
- (h) What will be the output of the following code segment:

```
String firstName = "Dhruba ";
String lastName = "Pradhan";
String fullName = firstName + lastName;
jTextField1.setText("Full Name: ");
jTextField2.setText (fullName);
```
- (i) What is the main difference between a while loop and a do while loop?
- (j) What will be the content of jTextField1 after executing the following code:

```
int Num = 6;
Num = Num + 1;
if ( Num > 5)
```

```

        jTextField1.setText(Integer.toString(Num));
    else
        jTextField1.setText(Integer.toString(Num+5));

```

- (k) What is the main difference between a combo box and a list box?
 (l) How many times will the following loop get executed?

```

x = 5;
y = 36;
while ( x <= y){
    x+=6;
}

```

- (m) Write a java program to print the larger of two numbers entered from the keyboard.
 (n) Accept a number from keyboard and check whether it is a positive number, negative number or a zero. Print suitable messages.

3. Short Answer Type Question

- (a) What are comments/remarks ?How can you give comments in java program.
 (b) Distinguish the if-else construct from a switch-case construct.
 (c) What is the significance of a break statement in a switch-case construct ?
 (d) Write the outputs of the following segments of programs :

(i) `if (a>0 && a<20)`

`a+ +;`

`else a—;`

`System.out.print(a);`

`/*Assume a as 0 in the above program*/`

(ii) `if (a> b)a!=b)`

`C =+ +a + — —b ;`

`System.out.print (c+""+a+""+b);`

(1) Assume a=5 and b=2

(2) Assume a=0 and b=0

- (iii) `System.out.print ((x+y>x*2)?x++:y); /* Assume x=5 and y=3*/`
- (iv) `System.out.print ("Income tax=Rs.");`
`System.out.print (Income>100000)?`
`10/100.0*Income 0); •`
`/* Assume Income=225000*/`
- (e) Write a program to check a number to be odd or even. Print suitable messages for the number.
- (f) A voter is eligible to vote for the elections held in his city if his age is atleast 18 yrs. Define a class `EligibilityTest` that checks for the eligibility of a voter.
- (g) Write the code for the following application



- (h) Write the code for the following application :



- (i) Write the code for the following application :



- (j) Explain the use of for statement along with its syntax.
 (k) What are relational operators? Explain with the help of suitable examples.

4. **Long Answer Type Question [**

- (a) Define a GUI application to create a list box ClassName with the following values.

Class Name
XI-A
XI-B
XI-C
XI-D

Write a program to print the names of the class teacher according to the class selected based on the following information

Class Name	Class Teacher Name
XI-A	Ranjan Kumar Bal
XI-B	Dhruba Charan Pradhan
XI-C	N.C. Samal
XI-D	Ajay Kumar Mishra

- (b) Design a GUI application as shown below: On selecting the radio button and clicking the Set Alignment button the alignment of the text on the button gets changed to Left, Right or Centre.



- (c) An electricity board charges the bill depending on the number of units consumed as follows:

Units	Charge
First 100 units	40 P per unit
Next 200 units	60 P per unit
above 300 units	Re.1 per unit

Write a java program to print the net bill to be paid by a consumer. The net bill includes a rent of Rs.250.00 charged from a consumer.

- (d) Write a program that returns the commission in Rupees for a salesman: The commission is based on the following conditions:

Sales	Commission%
< Rs. 5000	5%
Rs.5000 to Rs. 10000	8%
Above Rs. 10000	10%

- (e) Write a java program to accept a student's name, his marks in three subjects and print his result sheet in the following format:

NAME:

ENGLISH MARKS:

I.T. MARKS:

MATHS MARKS:
 TOTAL MARKS =
 PER CENT =
 GRADE =

Assign grade to the student based on the following grade scale:

Percent < 50 D
 Percent \geq 50 but < 60 C
 Percent \geq 60 but < 80 B
 Percent \geq 80 A

- (f) Write class 'leap' to accept any year and check if it is a leap year or not.
- (g) Write a program to compute and display the roots of a quadratic equation of the form $ax^2 + bx + c$.
 If $b^2 - 4ac$ results a negative answer, print IMAGINARY ROOTS.
 If $b^2 - 4ac$ gives a positive result, print REAL AND DISTINCT ROOTS.
 If $b^2 - 4ac = 0$, print EQUAL ROOTS
- (h) Write a complete java program to accept a day number from the user and print the corresponding day name. Make use of switch-case construct.
- (i) Write a java class to accept a month number and print the corresponding month name. Make use of switch-case construct.
- (j) A cloth showroom has announced the following festival discounts on purchase of items:

Amount of purchase	Discount in %	
	Mill Cloth	Handloom items
Less than Rs1000	2%	5%
Rs 1000 to Rs 5000	20%	25%
Rs 5001 to Rs 10000	40%	50%
Above 10000	50%	60%

Write a java program to compute the net amount paid by the customer. Assume all required values to be inputted by the user. Make use of if and switch structures.

- (k) An employee is entitled to pay an income tax based on his gross annual income as follows:

Annual gross income	Annual deduction %Tax
Less than or equal to Rs. 100000	0%
Rs. 100001 to Rs. 500000	Rs.1000 + 10 % of income exceeding Rs.100000
500001 to 800000	Rs.5000 + 20 % of income exceeding Rs.500000
>800000	Rs.10000 + 30 % of income exceeding Rs.800000

Write a java program to compute and print the tax payable by a salaried person.

- (l) Write a javaprogram to calculate and print the sum of each of the following series:
- (i) Sum=2-4+6-8+.....+20
 - (ii) Sum =1+3+5+.....+99



CHAPTER - 6

PROGRAMMING GUIDELINES

2.C PROGRAMMING GUIDELINES

Introduction

In these guidelines, several constructs are plainly outlawed. That doesn't mean that programmers using them are evil or incompetent. It does mean that the constructs are not essential and can be expressed just as well or even better with other language constructs. If you already have programming experience, in Java or another language, you may be initially uncomfortable at giving up some fond habits. However, it is a sign of professionalism to set aside personal preferences in minor matters and to compromise for the benefit of your group.

These guidelines are necessarily somewhat dull. They also mention features that you may not yet have seen in class. Here are the most important highlights:

1. Tabs are set every three spaces.
2. Variable and method names are lowercase, with occasional upper Case characters in the middle.
3. Class names start with an Uppercase letter
4. Constant names are UPPERCASE, with an occasional UNDER_SCORE.
5. There are spaces after keywords and surrounding binary operators.
6. Braces must line up horizontally or vertically.
7. No magic numbers may be used.
8. Every method, except for main and overridden library methods, must have a comment.

9. At most 30 lines of code may be used per method.
10. No continue or break is allowed.
11. All nonfinal variables must be private.

Note to the instructor: Of course, many programmers and organizations have strong feelings about coding style. If this style guide is incompatible with your own preferences or with local custom, please feel free to modify it

GENERAL CONCEPTS

A computer system cannot perform any task on its own. Programming guidelines are also known as **programming methodology** which includes various development models (such as RAD, Waterfall model, Spiral model etc.) and a developing environment (*i.e.*, IDE packages).

Program

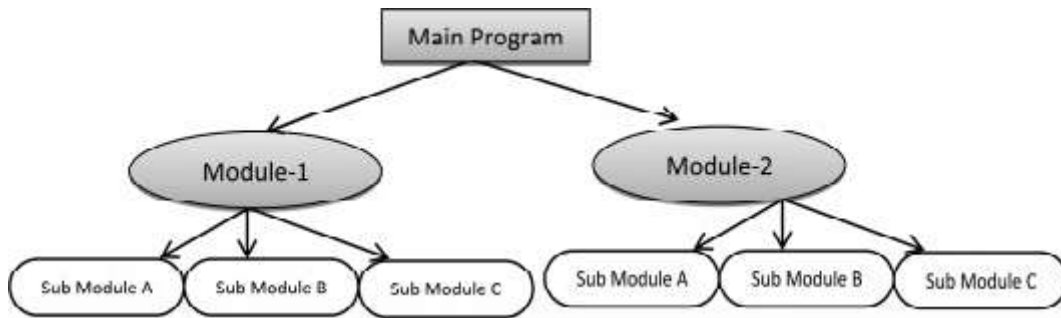
A program is a set of instructions provided to a computer system to perform a specific ask. Without programs a computer becomes useless. A simple program is divided into various parts as

1. **Instructions:** These are the code segments that contain some steps which are needed to be executed by the CPU (Central Processing Unit). An instruction is generally composed of two basic elements
 - (a) **Operator:** It defines the action carried out by the processor.
 - (b) **Operand:** Pieces of data on which the specified operation is performed. Operands are commonly known as **variables**.
2. **Constant:** It is an identifier with fixed values *i.e.* those values used within a program for comparing the data are known as constants.

MODULAR APPROACH

The modular approach to programming is a technique in which large program splits into a number of small parts commonly known as **modules**. Each module is designed to perform a specific task, such as to calculate the gross pay of an employee in a payroll system. A module may be further split into sub modules where each sub module is composed of self contained set of instructions.

For example:



Modular Approach

As shown in the above diagram, main program which is very large is divided into two sub modules named as module1 and module 2*i.e.*, a **big** problem is divided into two small problems. These two modules **can** be further divided into discrete sub modules which give an easy way to handle a problem.

Advantages of Modular Programming

1. It is easier to test and debug the individual modules as compare to a long program.
2. A modular program is easier to understand.
3. It is easier to modify a modular program by modification or replacement of individual modules.
4. Modules can be developed in a general way so these can be used with other programs.

Disadvantages of Modular Programming

1. Modular programs need more storage space and time for execution.
2. Integration of various modules into a single program may not be an easy task.
3. Testing and debugging of separate modules takes a lot of time.
4. Maintaining a link between different modules is a difficult job.

Different approaches are used in order to design a program consisting of modules

Top-Down Design

One programming approach that has proven to be most productive is called top-down decomposition. Top-down decomposition is the process of breaking the overall procedure or task into component parts (modules) and then subdivides each component module until the lowest level of detail has been reached. It is called top-down design or top-down decomposition since we start “at the top” with a general problem and design specific solutions to its sub problems.

In order to obtain an effective solution for the main problem, it is desirable that the sub problems (subprogram) should be independent to each other. Then each sub-problem can be solved and tested by itself.

The top level of the decomposition is the highest level of the overall plan. Unfortunately, there is no single formula that will decompose a complex problem into individual tasks. The strategy involves top-down reduction of the processing until a level is reached where each of the individual processes consists of one self-contained task, which is relatively easy to understand and can be programmed in a few instructions.

Bottom-Up Design

It begins at the lowest level modules or sub-programs and progresses upward to the design to the main program, or main module. To determine the order of execution, a structure chart is needed, and to complete the bottom up design, the development of drivers is needed.

Using bottom-up design strategy, we take an already existing computer program as a model for our new program. We try to utilize the existing facilities or design in a way, which gives out program a better performance.

RUNNING AND DEBUGGING PROGRAMS

Running a program is also known as **executing process** for finding whether the current coded program runs correctly or not and produces the output we want. It is an important step in program development by which a user ensures that the program code is correct and all their modules are working in a systematic manner.

Debugging is a process of finding and reducing the number of errors and defects, in a computer program. Debugger is a computer program that lets you run your program line by line and examine the values of the variables which are passed into functions.

It helps to figure out why a program is not running the way you expected *i.e.*, the art of pinpointing a bug and fixing it is known as debugging and when all bugs are cleared out then the program code is correct and error free.

A debugging process can be divided into three main parts

1. Identifying a bug

It is the first step of debugging in which a user needs to focus on the program code and identify whether the code is infected by a bug or not. If a bug is serve to cause the program to terminate abnormally, the existence of a bug becomes obvious.

If the bug is minor and only causes the wrong results, it becomes much more difficult to detect the existence of a bug.

2. Classifying a bug

This is an important step, which uses a bottom-up approach from the place where an error occur and analyze the code to classify the type of bug occurred in a program.

3. Fixing the bug

This is the straight forward step, after analyzing the original bug, in which you have to fix the bugs which cause series damage to your program and test the program after fixing it.

ERROR

An error is a flow, fault or failure in a computer program that causes it to produce an incorrect or unexpected result, or to behave in unintended ways. Most bugs arise from mistakes and errors made by people in either a program's source code or its design, or in frameworks and environment used by such programs.

Broadly, there are three type of errors

1. Compile-time Error

All the errors that are detected and displayed by the Java compiler are known as compile-time errors. Whenever the compiler displays an error, it will not be able to run. It is therefore necessary that we fix all the errors before we can successfully compile and run the program. There are two categories of compile-time errors : Syntax errors & Semantic errors.

(i) Syntax error

When a formal set of rules defined for writing a program in a particular language is not followed then error raised is known as syntax error. Syntax errors occur when syntax rules of any programming language are violated. NetBeans highlighted syntax errors in design state itself using the error indicator.

Some of the common examples of syntax errors are missing semicolon, parenthesis etc.

(ii) Semantic error

This type of compile time errors indicate an improper use of Java statement. These errors occur when statements are not meaningful. The word “semantics” relates to the meaning of words, sentences, or programs.

For example, use of an undeclared variable comes under semantic errors.

2. Run-time Error

If an application is syntactically correct then it is compiled and translated into machine language and ready to be executed. Run-time errors occur during the execution of a program. These errors will result the abnormal termination of program. Some of the common examples of run-time errors are number division by zero etc.

3. Logical Error

In case the program cannot give any error but still giving an incorrect output, it is due to logical errors.

These errors occur due to mistakes of the programmer. This is the most difficult task to find and debug a logical error.

For example, when a wrong formula is used to calculate any mathematical expression gives logical error.

PROBLEM SOLVING PROCEDURES**1. Understand the Problem**

The first step in software development is to specify the problem to be solved. This may be more demanding than it sounds. The specification stage is

also called **analysis**. The skills required have more to do with understanding a business or organization than using a programming language.

The more thorough the analysis and specification are, the less effort will be needed at a later stage to re-specify the task that the program is to perform. Professionals who specialize in this work are called **systems analysts**.

2. Identifying Minimum Number of Inputs required for Output

Key components of a program's specification are its input, its output, and the relationship between them. Software developers have long prepared sample input/output as part of program specification. Most of our programming exercises include sample I/O.

As software has become more complex, the specification may include entire screens, such as forms with which to get input or windows to display results.

3. Breaking Down of Solutions

Programming is very essential for any system to work, so the general concepts of programming are described below for better understanding of programming.

These are developed to provide computer solutions to user problems. So the software should be designed effectively so as to meet its objective of user satisfaction.

The software designing involves mainly two things

(i) Program Structures

It refers to how a program should be. The program structure is decided using top-down approach or any other popular approach. The main task is divided into clear logical subtasks thereby deciding about the subroutines, local and global variables, information hiding etc.

(ii) Program Representation

It refers only to the presentation style so that the program becomes more readable and presentable. A program which is easily understood can easily be checked for errors, if any, and easily modified.

It is seen that usually 60-75% of the time and 90% of the cost of the program is spent in program testing, debugging and modifying it. It is therefore essential to present the program in good style to minimize the wastage of time, efforts and cost.

GUIDELINES

GUI Programming uses a simplified approach to programming. In GUI Programming, most of the components are predefined in a generic way to be adapted and incorporated according to the needs or requirements of the application. It provides a very comfortable feel to the programmer to develop applications without getting into complicated and cumbersome logic and commands. Most of the GUI tools are developed to provide simplified approach to programming and provide enormous inbuilt functionality to the programmer. This chapter will help the readers to learn how to utilize GUI tools to develop programs for various applications in efficient and effective manner.

GUI Application Development Guidelines

Some good application development guidelines are:

1. Understand the need of the application before starting the development.
2. Find out all possible inputs, which are required to produce the desired result or results
3. Make sure that the user provides appropriate information with minimum efforts to avoid ambiguity in data. The same can be done by appropriately deciding on the various input components - maximize use of radio button, checkbox, combo box, and list. Wherever possible avoid use of text field and text area for accepting inputs from the user to reduce ambiguity.
4. Radio Button should be used wherever one of the option out of limited number of known set of options are required to be taken from the user. For example, for accepting gender (Male or female), marital status (Single or Married), for accepting membership type (Monthly, Annual or lifetime) etc.
5. Checkbox should be used wherever multiple options are required to be selected from a limited number of known set of options. For example, for accepting multiple hobbies (Swimming, singing, Dancing, Debating), for accepting food order in a restaurant (Pizza, Burger, Channaulcha, PaoBhaji, Chowmein) etc.
6. Combo box should be used wherever only one of the option from a large number of known set of options is required to be taken from

the user. For example, selecting state, selecting marital status, selecting schools etc.

7. List should be used wherever multiple options are required to be selected from a large number of known set of options. For example, selecting multiple food items from a menu containing five or more number of items.
8. It is advisable to use List and Combo box when there are too many options as they help save space and are less cumbersome to design as compared to radio button and checkbox.
9. Options in a list or a combo box may be displayed in alphabetical order so that it is easier for the user to locate the appropriate option or may be displayed according to the probability of choice. For example, to take input of name of a state the names should be displayed according to alphabetical order, to take input of employee designation the highest level should be put at last and the lowest level should be put at top. The explanation for this is since there are more employees at lower levels as compared to higher levels, the probability of choosing the lower level is more. In short the most probable answer should be at the top.
10. It is advisable to use appropriate labels for each input and output options to help the user to correctly interpret them.
11. While writing the code do not use variable names as A, B, C etc. Instead use meaningful names and follow naming conventions. All the variables and constants must be named according to the naming conventions. They make the code easier to read, understand and maintain. For example a variable storing total marks should be named as Total. Similarly a variable holding cost price may be named as CP.
12. While writing the code do not use variable names as A, B, C etc. Instead use meaningful names and follow naming conventions. All the variables and constants must be named according to the naming conventions. They make the code easier to read, understand and maintain. For example a variable storing total marks should be named as Total. Similarly a variable holding cost price may be named as CP.

13. The conditional construct `if..else` should be preferred when there are very few alternative execution paths to choose from and also when decisions are to be made based on ranges of values or conditions. For example, to show gender based title etc.
14. The `switch` construct should be used when there are too many alternative execution paths and decisions is based only on a single integer or enumerated value (countable value). For example, to show weekday based message etc.
15. For repeating code a known number of times, the `for` loop is the best choice and when the number of iterations is not preknown, use the `while` or the `do..while` loop. When you want to test a condition at the end of the loop to see whether some block should be repeated, the `do..while` statement should be preferred. For example to sum 10 numbers `for` loop is the best whereas to accept password, the `do..while` is the best.
16. Use appropriate comments. Comments are very essential for providing internal documentation of a program. Comments can be used to explain the various complicated steps of the program thereby making the program more understandable for the user. In Java single line comments begin with `//` and multiple lines of comments are enclosed between `/*` and `*/` (quotes are not to be included).
17. Insert blank lines and blank spaces where necessary to separate logical group of statements.
18. Proper indentation must be used to highlight nesting of constructs like `if`, `select` or loops.
19. Avoid using Free formatting styles. In Java we can type any number of statements in the same line separated by `;` (semi colon). This is called free formatting style but it makes program less readable and difficult to debug. So we should avoid it and instead Pretty printing should be encouraged.
20. The application must be reliable. It must be capable of handling situations like wrong input or no input. It should display proper error messages in case of such errors.
21. The application should be portable. It should be able to run on different platforms.

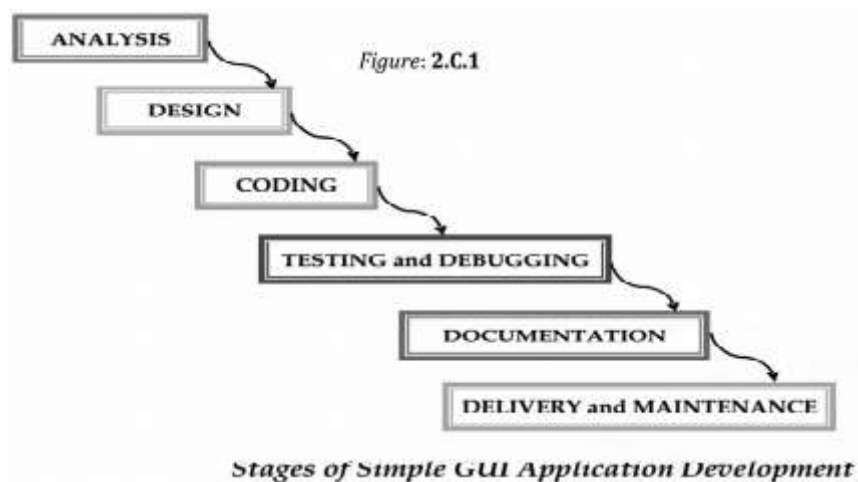
Stages of a Simple GUI Application Development

Various steps involved in the development of a new application are as follows:

1. **Analysis:** This phase involves the following steps:
 - (a) in-depth understanding of the problem
 - (b) deciding the requirements from the new system
 - (c) Jotting down possible inputs and outputs that are required for obtaining the desired solution.
2. **Design:** This phase involves planning of step-by-step procedures required to solve a given problem. At this stage a detailed design of the following components is to be completed:
 - (a) **Inputs:** involves defining the kind (data type) of data to enter into the application. In this stage we should also decide on the type of input components to minimize ambiguity and inconsistency.
 - (b) **Outputs:** decide on the possible data to be displayed from the application and also how, where and when it is to be displayed.
 - (c) **User Interface (Forms):** involves designing of the screen the user will see and use to enter data or display data. The placement of various input-output components on the form in an aesthetic and visually appealing manner is a major step in this phase.
 - (d) **Modular Components:** involves breaking of complex steps into simple ones to attain the target. Depending on the user interface this step will involve deciding on the functionality required from each component placed on the form to obtain the desired output.
 - (e) **Algorithms:** involves creating a simple solution in the form of steps called an algorithm and it helps in making the coding process easier.
3. **Coding:** This phase involves actual writing of programs using appropriate programming languages. A good programmer will make an optimum code, which is readable, easy to understand and maintain with appropriate error handling, comments and indentation.

4. **Testing and Debugging:** Virtually all applications have defects in them called 'bugs' and these need to be eliminated. Bugs can arise from errors in the logic of the program specification or errors in the programming code created by a programmer. Testing means the process of executing the application with possible set of input data in order to find probable errors. Debugging means correction of those errors in the application. In the testing and debugging stage, we should try out all possible inputs in order to make our application error free.
5. **Documentation:** Documentation means the instructions and information about the usage of the application. Providing the documentation makes it easier for the end user to understand the functionality of the application.
6. **Application Delivery and Maintenance:** The completed software is packaged with full documentation and delivered to the end users. When the end users use the software, bugs that were not found during testing may appear. The maintenance involves the rectification of previously undetected errors and changes that are to be made for the enhancement of the functionality of the application. An updated version of the software with the reported bugs corrected and enhancements is then sent as a replacement to the end user.

Figure: 2.C.1



EXERCISE**1. Multiple Choice Question**

- (a) _____ is the process of translating a task into a series of commands that a computer will use to perform that task.
- A. Project design B. Installation
C. Systems Analysis D. Coding
- (b) Translating the problem statement into a series of sequential steps describing what the application must do is known as:
- A. Coding. B. Debugging.
C. Creating the algorithm. D. Writing the documentation
- (c) Which of the following component is the best suited to accept the country of the user?
- A. List B. Combo box
C. Radio button D. Check box
- (d) Which type of loop is best suited to check whether the password input by the user is correct and display an error message?
- A. for B. do..while
C. while D. All of the above
- (e) Which of the following is not a good programming guideline?
- A. Adding lots of comments
B. Using prettyprinting
C. Using text fields to accept input of marital status
D. Designing visually appealing forms

2. Short Answer Type Question

- (a) Differentiate between compile time and run time errors.
- (b) Which error is harder to locate and why?
- (c) What is unit testing?
- (d) What is white box testing?
- (e) What is validation ?
- (f) Name the three types of debugging error.

3. Short Answer Type Question

- (a) Which methods are used to detect the errors in the program,?
- (b) Differentiate between white box and Black box Testing.
- (c) Differentiate between verification and Validation.
- (d) Define and explain problem solving methodology.

4. Long Answer Type Question

- (a) Write the steps involved in program development?
- (b) Describe the factors which are considered to decide the goodness of a program.
- (c) Explain errors and their different types in detail.
- (d) How is modular approach useful in programming?
- (e) Explain the following terms:
 - 1. Exception handling
 - 2. Syntax error
 - 3. Syntax



Unit

3

RELATIONAL DATABASE MANAGEMENT SYSTEM

CHAPTER - 7

**Database Management
System**

CHAPTER - 8

Introduction to MySQL

CHAPTER - 7**DATABASE MANAGEMENT SYSTEM****3.A DATABASE MANAGEMENT SYSTEM****Introduction**

From ancient times man has been keeping record of events in different forms and formats. Information in respect of finance, scientific research, social parameters and hosts of others has been maintained meticulously all over the world by skilled personalities for future and immediate uses. In our day-to-day life, we maintain diaries (note down the events of different categories & their timing in daily basis), a shopkeeper maintains its stock-register & account book, Academic Section of a college maintains the Long Roll Register & Tabulation Register (keeps the bio-data & mark secured in various examinations by the students) and so on. We record this information manually on papers, for future use. Advent of modern digital computers has made the task easier, because of its speed, accuracy and availability of large memory spaces.

In computers, data are stored in two different ways- *file base system* and *database system*. In file-base, permanent records are stored in various files. One has to write/use various application programs to add or extract records in appropriate files. As applications proliferate, number of files increase considerably. Increase in the number of users further escalates their count. This system fails to effectively check the duplication, accuracy and integrity of data; retrieval and security of information become less effective.

To minimize the limitations of a file-base record-keeping system, the database approach is conceived. Initially, simple and flat databases were in use; the information are arranged in rows and columns of a table. Look at the following tables.

Student Table

ROLL_NO	NAME_STUDENT	BIRTH_DT	ADDRESS	TEL_NO
IA15-001	Akash Acharya	01-Apr-1999	1251, Surya Vihar Unit-II, Bhubaneswar	9900099999
IA15-002	Bisakha Soren	23-Nov-2000	Meramundali, Dhenkanal	7134508764
IC06-002	Sasmita Mallick	16-Jun-1989	Mandia, Patkura	9439726666

Subject Table

ROLL_NO	COMP_1	COMP_2	ELECT_1	ELECT_2	ELECT_3	ELECT_3
IA15-001	English	MIL(Odia)	History	Economics	Pol Science	IT
IA15-002	English	MIL(Hindi)	Pol Science	Logic	Education	Statistics

Mark Table

ROLL_NO	CMARK_1	CMARK_2	EMARK_1	EMARK_2	EMARK_3	EMARK_4
IA15-001	63	70	81	65	02	80
IA15-002	54	85	63	75	69	23

Book Table

BOOK_ID	BOOK_NAME	PRICE	PUBLISHER_ID
ACT-001	A Textbook of Accountancy	250-00	PUB-003
IT-001	Information Technology_ Vol -I	270-00	PUB-001
PHY-001	Higher Secondary Physics_Part-1	360-00	PUB-002

Publisher Table

PUBLISHER_ID	PUB_NAME	PUB_ADDRESS	PUB_TEL_NO
PUB-001	Text Book Bureau of Odisha	123, SatyaNagar, Bhubaneswar-4	0674-232121
PUB-002	Rajahans Press	College Square, Cuttack-753003	0671-345329
PUB-003	ABC Publishing House	Modi Para, Sambalpur-765001	8760987640

Book Issue Table

ISSUE_DT	ROLL_NO	BOOK_ID	RETURN_DT
09-Sept-2015	IA15-006	IT-001	24-Sept-2015
23-Sept-2015	IC14-002	ACT-001	03-Opt-2015
01-Dec-2015	IS15-036	PHY-001	03-Jan-2016

In all practical purposes, databases are being maintained in various sectors, like: large/small Business houses, Academic & Research Institutions, Police Organizations, Local Civic Bodies & many other Public & Private Sector Units. Almost all e-commerce sites use databases to store product inventory and customer information.

Databases are composed of related tables, while tables are composed of rows and columns. The first row describes the type of information to be stored under different columns & subsequent rows hold the actual information. In the six tables, shown here, information have been recorded as per the specification of their Column-headings. Information stored in a **column** (generally called as „**field**“) represent a specific **attribute** of the entities; each row holds the information for one entity (called as „**record**“ or „**tuple**“). Each Information is called a „datum“ („**data**“ in plural).

As such a **field** is the smallest unit of a record in a database. ROLL_NO, BIRTH_DT, ADDRESS & PUB_ID, etc are some of the field names in the above tables. Fields are specified by:

Field Name - Name given to the Field.

Data Type: It specifies a type of data for each data items in a table.

Field Size: Amount of space reserved for storing data.

An **entity** is an object, which may have a logical or physical existence; e.g. place, person or an organization, etc. Each entity has certain properties or **attributes** associated with it and operations applicable to it. **Type** and **Instance** are two important terms related to an entity. An *entity type* refers to the objects which share common properties e.g. STUDENT, COURSE etc. An entity type is usually denoted in UPPER CASE letters. An entity instance is a specific person or organization etc. For example, “Akash” and “Text Book Bureau of Odisha” are the instances of STUDENT and PUBLISHER respectively. An entity instance is usually denoted in normal case letters.

While storing the information under different **fields**, care must be taken, that data is accurate and as per the real world values; otherwise one may enter „31-Feb-2016 under a „Date column!

Data-type:

Look at the Data tables shown earlier; each column holds data of a particular type. BIRTH_DT column in Student Table or ISSUE_DT in Issue Table stores only Date-values in a given date-format (dd-mmm-yyyy). Similarly, TELEPHONE_NO field in Students Table & CMARK-1 column in Mark Table hold only Number values. We shall now discuss different type of data which are in use to populate a database table.

In general there are four types of data;

- (i) *Text / Character*: All type of characters (letters, numbers, special characters) & character strings
- (ii) *Numeric*: Numbers which can be manipulated using mathematical operators
- (iii) *Date & Time*: Calendar dates which can be manipulated mathematically.
- (iv) *Logical*: True or False, Yes or No

A **data-type** in a programming language defines specific characteristics for data to be entered in a field. Usually, a limited number of such data types come built into a language package. The package specifies the range of values for a given data type, processing of their values by the computer, and how they will be stored.

In object-oriented programming (OOP), a programmer can create new data-types (user-defined data types) to meet application needs, called as „data abstraction . Such data-types are based upon the “built-in” data types of the package; such as number, integers and characters etc. Languages that leave little room for programmers to define their own data-types are said to be strongly-typed (A strongly-typed programming language is one in which each type of data (such as integer, character, hexadecimal, packed decimal, and so forth) is predefined as part of the programming language and all constants or variables defined for a given program must be described with one of the data types. Certain operations may be allowable only with certain data types. The language compiler enforces the data typing and uses compliance. An advantage of strong data

typing is that it imposes a rigorous set of rules on a programmer and thus guarantees a certain consistency of results. A disadvantage is that it prevents the programmer from inventing a data type not anticipated by the developers of the programming language and it limits “creativity” of its user) languages. An Open-source programming language like „Java defines eight primitive data types: **int, short, long, byte, float, double, char & boolean**. The first four types store integer values, while the next two are used to store floating-point values. „Char data type is used for Unicode encoding scheme & „Boolean is for true or false values. Each data type occupies certain amount of memory space as defined by the developer. Other database packages use the data-types listed in the following table:

Data-type	MS-Access	SQLServer	Oracle	MySQL
boolean	Yes/No	Bit	Byte	N/A
integer	Number (integer)	Int	Number	Int / Integer
float	Number (single)	Float/ Real	Number	Float
currency	Currency	Money	N/A	N/A
string (fixed)	N/A	Char	Char	Char
string (variable)	Text (<256)	Varchar	Varchar	Varchar
binary object	OLE Object	Binary/ Varbinary	Long/Raw	Blob/Text

In addition to the data-types described above, following types are also used frequently in different programming languages.

- (i) DECIMAL (p, s): Exact numerical with precision p and scale s . e.g. **decimal (5, 2)** is a number that has 3 digits before the decimal and 2 digits after the decimal
- (ii) REAL: Approximate numerical, mantissa precision 7
- (iii) DATE: Stores year, month, and day values
- (iv) TIME: Stores hour, minute, and second values
- (v) TIMESTAMP: Stores year, month, day, hour, minute, and second values
- (vi) ARRAY: A set-length and ordered collection of elements
- (vii) XML: Stores XML data

CHAR

1. Used to store character string value of **fixed length**.
2. The maximum no. of characters the data type can hold is **255 characters**.
3. It's **50% faster** than VARCHAR.
4. Uses **static memory allocation**.
5. For CHAR columns, truncation of excess trailing spaces from inserted values is performed silently regardless of the SQL mode.

VARCHAR

1. Used to store **variable length** alphanumeric data.
2. The maximum this data type can hold is up to **65,535 characters** shared for the row.
3. It's **slower** than CHAR.
4. Uses **dynamic memory allocation**.
5. For VARCHAR columns, trailing spaces in excess of the column length are truncated prior to insertion and a warning is generated, regardless of the SQL mode in use.
6. VARCHAR values are not padded when they are stored. Trailing spaces are retained when values are stored and retrieved, in conformance with standard SQL.

Relational Database:

A database is a collection of information organized to provide efficient updating, editing and retrieval. The collected information could be in any number of formats (electronic, printed, graphic, audio, statistical, combinations). There are physical (paper/print) and electronic databases. It could be as simple as an alphabetical arrangement of names in an Address book or as complex as a College database that provides information in a combination of formats. Most databases contain multiple tables, which may each include several different fields. For example, a company database may include tables for products, employees, and financial records. Each of these tables would have different fields that are relevant to the information stored in the table.

Let us consider a database for a College; it may contain tables for storing information about its Students, Teachers, Non-teaching staff members, Infrastructure maintenance, Library maintenance, financial accounts and so on. Most of the tables would have inter-dependencies, like: a student or a teacher transact (borrow/return books or journals) with the library, a student deposits fees and receives scholarships from the Account section, a teacher maintains the information of a group of students as a proctor and so on. Invariably, most databases contain such inter-related tables (tables are related to each other through common columns), and are called as **Relational Database**. We shall now discuss some useful aspects of a Relational Database (RDB).

RDBs establish a well-defined relationship between database tables. Tables communicate and share information, which facilitates searching (query), editing and reporting. Look at the tables described earlier; the **Student table** and the **Subject table** can be related through the common column „ROLL_NO . The **Book table** & the **Publisher table** can be linked via the field „PUBLISHER_ID. Further the data under the column „ROLL_NO uniquely represent a record in the **Student table**. Such *field(s)* (or *columns* or *attributes*) in a table are called as „Key in RDBs. In general, we define five types of keys in RDBs.

- (i) Primary key
- (ii) Candidate key
- (iii) Alternate key
- (iv) Foreign key
- (v) Composite key

To illustrate the idea of different keys in an RDB, let us create a table named „ADMISSION to input information relating to students Money Receipt-cum-Index Number (MRIN), ROLL_NO, NAME, STREAM, SEC, FEES and Date of Admission (DOA). Thus, there are seven fields of different data-types.

MRIN:	a CHAR type data of a fixed length (13)
ROLL_NO:	a CHAR type data of a fixed length (or may be an INTEGER data-type)
NAME:	a CHAR type data of a fixed maximum length (45)
STREAM:	a CHAR type data of a fixed maximum length (9)
SEC:	a CHAR type data of a fixed length (1)
FEES:	a MONEY/CURRENCY data-type
DOA:	a DATE data-type with format of „dd-mmm-yyyy

ADMISSION Table

MRIN	ROLL_NO	NAME	STREAM	SEC	FEES	DOA
19051101/0144	IA15-001	Akash Acharya	ARTS	A	630-00	23-Jun-2015
19051203/3240	IS15-001	Bhabagrahi Das	SCIENCE	A	810-00	23-Jun-2015
07181203/1503	IC15-001	U C Chandrika	COMMERCE	A	720-00	23-Jun-2015
07032102/0677	IA15-002	Bisakha Soren	ARTS	B	630-00	23-Jun-2015

When you applied for admission you got an Index Number (MRIN), which is not assigned to any other applicant. After admission, you were given a Roll Number. The Roll Number assigned to you was not assigned to any other student of your college; which means this is uniquely refers to you only!. When a person opens an account in a bank, he/she would be provided with a unique Account Number. Roll Number & Account Number are examples of Primary key.

A **Primary key** is a field in a table that is unique for each record. Every database table should have a column or a group of columns designated as the primary key. The purpose of a primary key is to uniquely identify each record in a table. Some other examples of Primary key are: BOOK_ID in Book table, PUB_ID in Publisher table, EMP_ID of an Employee Table, Item Code of an item in the Stock table, etc.

Look at the other fields like, MRIN & NAME; these two fields also identify a record uniquely. All such fields are called candidate keys.

A **Candidate key** is an attribute (or set of attributes) that uniquely identifies a row. A Primary Key is one of the candidate keys. A table may have more than one candidate keys but definitely has **one and only one** Primary key.

Only one of the candidate keys is selected as the primary key of a table. All other candidate keys are called **Alternate keys**. In the above example, if we use one of the candidate keys, say, ROLL_NO as the Primary Key, the other Candidate Keys MRIN and NAME would be the Alternate keys.

Foreign Key is that attribute/column/field, which are common in two database tables and is used to define relationship between them. A Foreign key ensures referential integrity. We can create more than one foreign key per table. A foreign key is generally a primary key from one table that appears as a field in another where the first table has a relationship to the second. For example look at the **Book table** & the **Publisher table**. The common field **Relational** „PUBLISHER_ID appears in both of them. This field is the Primary key in the

Publisher table; therefore it is said to be the **foreign key** in the **Book table**. The two tables are related through this field.

Some times we create a **key** on more than one column of a table; such a key is said to be a **Composite key**. To illustrate this concept, let s go through the following tables.

Table-1

STREAM	ROLL	NAME	SEC	FEES	DOA
ARTS	001	Akash Acharya	A	630-00	23-Jun-2015
SCIENCE	001	Bhabagrahi Das	A	810-00	23-Jun-2015
COMMERCE	001	U C Chandrika	A	720-00	23-Jun-2015
ARTS	002	Bisakha Soren	B	630-00	23-Jun-2015

In this table combination of „STREAM and „ROLL columns have to be combined to create a Primary **key**, because neither the ROLL nor the STREAM independently can represent a unique record in the table. Therefore, such a key, is said to be a **Composite key**. Not only the Primary key, but any other key like: Alternate keys or foreign keys present in a table, created out of more than column/attribute can be called as Composite keys.

RDBs organize data in Tables. Each table is known as a relation, which contains one or more data category columns. Each record (or row) contains a unique data instance defined for a corresponding column category. One or more data or attribute relate to one or many records to form functional dependencies. These are classified as follows:

- ☞ One to One: One table record relates to another record in another table.
- ☞ One to Many: One table record relates to many records in another table.
- ☞ Many to One: More than one table record relates to another table record.
- ☞ Many to Many: More than one table record relates to more than one record in another table.

Advantages of Relational Database:

- ☞ *Scalability:* New data can be added without modifying existing records.

- ▣ *Flexibility*: It can deal with multiple data requirement capabilities.
- ▣ *Data security*: It is capable of blocking critical data during data sharing.

Relational Database Management System (RDBMS)

An RDBMS is software that organizes and maintains the information in database tables. Another component, the Database Application is the program that lets us view, retrieve and update information stored in the tables. An RDBMS is used to

- (i) Store data in a structured manner
- (ii) Validate data for its consistency
- (iii) Add, modify, and delete data.
- (iv) Sort and manipulate data in the database
- (v) Retrieve data selectively through queries
- (vi) Generate flexible reports based on data both on screen(soft copy) and on paper (hard copy)

An RDBMS is based on the relational model specified by Edgar F. Codd, a British computer scientist with IBM during the year 1970. As per the paper published by him “A Relational Model of Data for Large Shared Data Banks”

1. Data must be stored and presented as relations, through key fields (primary/foreign keys).
2. To manipulate the data stored in tables, a system should provide relational operators - that enables the relationship to be tested between two entities.

Codd later published another paper that outlined the 12 rules that all databases must follow to qualify as relational. Many modern database systems do not follow all 12 rules, but these systems are considered relational because they conform to at least two of the 12 rules. RDBs use Structured Query Language (SQL), which is a standard user application that provides an easy programming interface for database interaction. Most modern commercial and open-source database systems are relational in nature. Some popular DBMS suits are,

- (i) Commercial DBMS packages
 - a. Oracle DB
 - b. IBM DB2

- c. Microsoft SQL Server
 - d. Microsoft Access
 - e. Sybase
- (ii) Open Source database suites
- a. MySQL
 - b. PostgreSQL
 - c. Ingres
 - d. Enterprise DB

Database Management Tools for Mobile devices:

In our day-to-day life, we have been using various electronic gadgets like, Mobile phones, Laptop computers, wireless printers and variety of smart devices (phone, wrist-watch, video camera, digital music players, etc). The retail shops are using Point of Sale (POS) machines to accept money in electronic forms, Bank ATMs disburse cash, and Public Utility Departments are not only providing utility bills like electricity & water dues on the spot, but also collecting the dues through smart devices. Here, we are using the word „smart“, to describe that the device can be connected over internet platforms. Phones, S More and more mobile applications need data to work, and databases have for quite a while been the most common way of storing and managing data. So, in a typical scenario, a mobile application uses a database that is hosted in the cloud, and connects remotely to it in order to access its data. This of course implies that, in order to be responsive, a mobile application needs an active and quite fast network connection.

Embeddable databases try to find their place in a world where mobile applications are smarter, and can take over more tasks in order to become either more independent, in a stand-alone scenario, or more active, in a distributed one. Apart from the five embeddable databases presented here, there are several more. SQL, document-oriented, key-value, object-oriented, graph... Choose the one that best suits the structure of your data (or the lack of it) and the needs of your application and use it.

The table gives an overview of the type of the data that each one of these databases can store, the license under which it is distributed, and the platforms it supports.

Database	Type of data stored	License	Supported platforms
BerkeleyDB	relational, objects, key-value pairs, documents	AGPL 3.0	Android, iOS
Couchbase Lite	documents	Apache 2.0	Android, iOS

Database	Type of data stored	License	Supported platforms
LevelDB	key-value pairs	New BSD	Android, iOS
SQLite	relational	Public Domain	Android, iOS, Windows Phone, Blackberry
UnQLite	key-value pairs, documents	BSD 2-Clause	Android, iOS, Windows Phone

Berkeley DB (Oracle Corporation): Berkeley DB is a family of open-source libraries that allow you to manage your data either in the traditional relational way (using SQL) or as key-value pairs (where both keys and values are byte arrays) or as Java objects or as XML documents. Regardless of the API you choose, Berkeley DB promises indexing, caching, transactions, single-writer/multiple-reader access, full-text search (for the SQL interface), automatic failure recovery, replication, compression and encryption. It also supports both in-memory and on-disk databases.

Another intriguing feature of Berkeley DB is that the SQL API it provides is fully compatible with that of SQLite. As a result, you can make your SQLite applications use Berkeley DB as the underlying storage engine without having to re-write them. All you need to do is link them to a suitable version of the Berkeley DB library. This combination of SQLite and Berkeley DB has been reported to perform better in case of highly concurrent, write-intensive applications.

Berkeley DB provides API bindings for several programming languages, including C++ and Java, and has been compiled, run and tested on several operating systems, including Android and iOS. Nevertheless, you might be put off by its AGPL license, in case you want to use it in your application, but you do not want to distribute your own source code in exchange.

Note also that, mainly because of its effort to provide features that are expected in traditional client/server databases, Berkeley DB is considered as a

relatively heavyweight solution. So, unless you absolutely want a feature that is only offered by Berkeley DB, then there are other more lightweight alternatives you can choose from.

Update: Even though Berkeley DB technically supports iOS, it can't be used in App Store apps due to license incompatibility with App Store terms & conditions.

Couchbase Lite: Couchbase Lite is a document-oriented database: data is stored as JSON documents. Each document may have one or more attachments, which are essentially uninterpreted binary data that is stored and loaded separately from the document itself. Couchbase Lite supports persistent indices (called views), and uses map-reduce to manage and query them.

Couchbase Lite, together with Couchbase Sync Gateway and Couchbase Server, are the three components of the NoSQL JSON mobile database solution offered by Couchbase called Couchbase Mobile. One of the key features of Couchbase Mobile is built-in synchronization (through Couchbase Sync Gateway) between local databases (Couchbase Lite) and databases in the cloud (Couchbase Server), which relieves the developer from the burden of writing their own sync code. In order to support this feature, Couchbase Lite comes with a conflict resolution mechanism that is quite similar to the one used by Git.

Couchbase Lite provides a Native API for both Android and iOS, as well as plug-ins for two widely used cross-platform tools for mobile development: PhoneGap and Xamarin.

LevelDB: LevelDB is an open-source library (written at Google) that implements a key-value store, where keys and values are byte arrays, and data is stored ordered by key (based on some, probably custom, comparison function). LevelDB supports atomic batch updates, forward and backward iteration over the contents of the store, snapshots (i.e. consistent read-only views of the entire store), caching, data integrity (using checksums), and automatic data compression (using the Snappy compression library). An important limitation imposed by the library is that at any moment at most one process can have access to a specific database.

“LevelDB has good performance across a wide variety of workloads” state its authors, and here are the results of a benchmark they put together to prove their claim by comparing LevelDB with SQLite and Kyoto Cabinet (another key-value store implementation).

LevelDB is written in C++. iOS developers can use it directly in their applications or through one of the several Objective-C wrappers that are available for it. Android developers can use LevelDB via JNI and NDK.

SQLite: SQLite is an open-source C library for managing relational databases that can be stored both on disk and in memory. It supports dynamic typing (types are assigned to values, rather than to columns), transactions, full-text search, single-writer/multiple-reader access, efficient range queries (useful when storing geographical data), and shared caching (across connections opened by a single thread to the same database).

SQLite stores each database as a single disk file in a cross-platform format. This implies that you can create an SQLite database on one machine and then use it on another machine with an entirely different architecture by simply copying the corresponding file.

The authors of SQLite believe that it is “the most widely deployed SQL database”. Bindings for SQLite are available in most mainstream programming languages. SQLite is also included in both iPhone and Android operating systems.

UnQLite: There are a lot of people that incorrectly believe that UnQLite is the NoSQL counterpart of SQLite, although the two products are completely independent. UnQLite is an open-source database that exposes both a key-value and a document store interface. It supports both in-memory and on-disk databases, transactions, multiple concurrent readers, and cursors for linear traversal. Like SQLite, UnQLite uses a cross-platform file format, and stores each database in a single file on disk.

No matter which store interface you use, UnQLite's storage engine works with key-value pairs. A distinctive feature of UnQLite is that you can change at run-time the storage engine that is used. At the moment, UnQLite comes with two built-in storage engines, one for on-disk and one for in-memory databases.

UnQLite is written in C, and can thus be used in both iOS and Android applications (in the latter, through JNI).

CHAPTER - 8
INTRODUCTION TO MySQL

3. B INTRODUCTION TO MYSQL (ANSI SQL 99 standard commands)

Introduction

Structured Query Language (SQL), which is a computer language for storing, manipulating and retrieving data stored in relational database management systems (RDBMS). It is the standard language for Relation Database System; almost all relational database management systems like, MS Access, Oracle, Sybase, Informix, postgres and SQL Server and MySQL use it as standard database language.

SQL was initially developed at IBM by **Donald D. Chamberlin, Donald C. Messerly, and Raymond F. Boyce** in the early 1970s. This version initially called SEQUEL (Structured English Query Language), was designed to manipulate and retrieve data stored in IBM's original RDBMS. In the late 1970s, Oracle corporation saw the potential of the concepts described by Codd, Chamberlin and Boyce and developed their own SQL-based RDBMS with aspirations of selling it to the various U.S. government agencies. In June 1979, Oracle Corporation introduced the first commercially available implementation of SQL, Oracle V2 for VAX computers. The American National Standards Institute (ANSI) adopted SQL as the standard language for RDBMSs in 1986. The International Standards Organization (ISO) has also adopted SQL as the standard language for RDBMSs.

Characteristics of SQL

Main characteristics of SQL are given below

1. SQL is an ANSI and ISO standard computer language for creating and manipulating databases.

2. SQL allows the user to create, update, delete, and retrieve data from a database
3. SQL is very simple and easy to learn
4. SQL is used specifically for relational databases.
5. SQL works with database programs like DB2, Oracle, MS Access, Sybase MS SQL Server, etc.

Advantages of SQL

1. **High Speed** SQL queries can be used to retrieve large amounts of records from a database quickly and efficiently.
2. **Well Defined Standards Exist** SQL databases use long-established standard, which is being adopted by ANSI and ISO.
3. Non-SQL databases do not adhere to any clear standard.
4. **No Coding Required** Using standard SQL, it is easier to manage database systems without having to write substantial amount of code.
5. **Emergence of ORDBMS** Previously SQL databases were synonymous with relational database with the emergence of object oriented DBMS, object storage capabilities are extended to relational databases.
6. **Controlling a Database** This includes system security and authorization which must be controlled and maintained

MySQL is the most popular Open Source RDBMS based on SQL (Structured Query Language). In MySQL information is stored in tables and runs virtually on all platforms including Linux, Unix and Windows. It is one of the fastest relational databases management system (RDBMS) in the market today. MySQL was developed by a Swedish company named MySQL AB.

Features of MySQL are:

- ☞ Free to download and use.
- ☞ Fast, reliable, and easy to use.
- ☞ Database system used on the web.

- ☞ Database system that runs on a server.
- ☞ Ideal for both small and large applications.
- ☞ Uses standard SQL.
- ☞ Gets compiled on a number of platforms.
- ☞ Supported by almost all operating platforms.
- ☞ Works with almost all programming languages (PHP, PERL, C, C++, Python, Ruby, .NET, JAVA, etc).
- ☞ Customizable. The open-source GPL (General Public License) license allows programmers to modify the MySQL software to fit their own specific environments.
- ☞ It is developed, distributed, and supported by Oracle Corporation

History of MySQL

MySQL was introduced by a software company MySQL AB, founded by **David Axmark, Allan Larsson** and **Michael Widenius** in the year 1994 in Sweden. It was first released on 23-May-1995, for personal usage based on ISAM (Indexed Sequential Access Method). Sun Microsystems acquired MySQL AB on 26-February-2008.

It supports the basic principles of database and data manipulation used to retrieve, insert and update stored data. Being an open source anyone can use and change the software for their needs. It supports the basic principles of database and data manipulation using SQL statements.

Components of MySQL

A database is a structured collection of vast amount of data. To add, access, and process data stored in a computer database, we need a database management system. MySQL can be used for variety of applications but it is mostly used for the web applications on the Internet. MySQL consists three components:- Server, Client and Client Library.

(i) Server (MySQL server)

MySQL server is an engine which provides access to databases, responsible for creating, managing database, executing and returning queries and maintaining security together with additional tools to manage multiple MySQL servers.

(ii) Clients

A client is a program that connects to the database server and issues queries in a pre-specified format. In MySQL, you can enter command-line queries to manage user permissions and utilities to import and export MySQL database.

(iii) Client Library

A client library is an Application Programming Interface (API) where a client can write their own programs using a programming language like C, C++, Java etc.

SQL Commands

SQL commands can be used not only for searching the database but also to perform various other functions like create tables, add data to tables, modify data, drop the table, set permissions for users and many more. *SQL commands can be classified into following categories*

1. Data Definition Language (DDL)
2. Data Manipulation Language (DML)
3. Transaction Control Language (TCL)
4. Data Control Language (DCL)

1. Data Definition Language (DDL)

DDL is used to define the structure of the tables and other objects in a database. In SQL, the Data Definition Language allows to create, alter and remove database objects. Basically, a data definition language is a computer language used to create and modify the structure of database objects in a database. These database objects include views, schemas, tables, indexes, etc. DDL consists of various commands are as follow:

1. **CREATE:** This command is used to create objects in the database.
2. **ALTER:** This command is used to alter the structure of the database table. This command can add up additional column, drop existing columns and even change the data type of columns involved in a database table

3. **DROP:** This command is used to remove objects from the database.
4. **TRUNCATE:** This command is used to remove all records from a table.
5. **RENAME :** This command is used to rename an object

2. Data Manipulation Language (DML)

DML provides various commands used to access and manipulate data in existing database. This manipulation involves inserting data into database tables, retrieving existing data, deleting data from existing tables and modifying existing data. The basic goal of DML is to provide efficient human interaction with the system. Various data manipulation language commands are as follows:

1. **SELECT:** This command is used to retrieve data from a Database and its objects.
2. **INSERT:** This command is used to insert data into a table.
3. **UPDATE:** This command is used to update existing data within a table.
4. **DELETE:** This command is used to delete the records from a table.
5. **LOCK TABLE:** This command is used to control concurrency.

3. Transaction Control Language (TCL):

TCL commands are used to manage transactions in database. These are also used to manage the changes made by DML statements. It allows statements to be grouped together into logical transactions. A transaction is a single unit of work. Each individual statement is a transaction. If a transaction is successful, all of the data modifications made during the transaction are committed and became a permanent part of the database. If a transaction encounters an error and must be cancelled or rolled back, then all of the data modifications are erased. To manage all these operations, transaction control language commands are used.

Various transaction control statements are as follows

1. **COMMIT:** This command is used to save the work done.
2. **SAVEPOINT:** This command is used to identify a point in a transaction to which you can later rollback.

3. **ROLLBACK:** This command is used to restore database to original since the last COMMIT.
4. **SET TRANSACTION:** This command is used to establish properties for the current transactions.

4. **Data Control Language: (DCL)**

DCL commands are used to assign security levels in database which involves multiple user setups. They are used to grant defined role and access privileges to the users. *There are two kinds of user in the schema:*

- ☞ **Users** They work with the data, but cannot change the structure of the schema. They write data manipulation language.
- ☞ **Admin** They can change the structure of the schema and control access to the schema objects. They write data definition language.

Basically, the DCL component of the SQL language is used to create privileges to allow users access to and manipulation of the database. *Two types of DCL commands are*

1. **GRANT:** This command is used to give user's access privileges to database.
2. **REVOKE:** This command is used to withdraw access privileges given with the GRANT command.

DATA TYPES

A data type a data storage format that can contain a specific type or range of values. Data can be stored in many different formats called 'data types'. Data types are required so that you know what "kind" of data your variable holds, or can hold.

Some mainly used data types in DBMS are as follows

1. Numeric data type
2. String data type
3. Date and time data type.

NUMERIC DATA_TYPE

DATA TYPE	SIZE (BYTE)	DESCRIPTION
TINYINT	1 byte	A small integer; range: from -128 to 127. If unsigned, the allowable range is from 0 to 255. Width of up to 4 digits.
SMALLINT	2 byte	A small integer ;range: from -32768 to 32767. If unsigned, the allowable range is from 0 to 65535. Width of up to 5 digits.
MEDIUMINT	3 byte	Range is from -8388608 to 8388607. If unsigned, the allowable range is from 0 to 16777215. Width of up to 9 digits.
INT	4 byte	Range is from -2147483648 to 2147483647. If unsigned, the allowable range is from 0 to 4294967295. Width of up to 11 digits.
BIGINT	8 byte	Range is from -9223372036854775808 to 9223372036854775807. If unsigned, the allowable range is from 0 to 18446744073709551615. Width of up to 20 digits.
FLOAT(p,s)	4 byte	A floating-point number that cannot be unsigned. Display length/precision(p) and the number of decimals/scale (s). Default precision

DATA TYPE	SIZE (BYTE)	DESCRIPTION
		is(10,2), where 2 is the number of decimals and 10 is the total number of digits (including decimals). Decimal precision can go to 24 places for a FLOAT.
DOUBLE(p,s)	8 byte	Same as FLOAT, with default precision(16, 4). Decimal precision can go to 53 places for a DOUBLE. REAL is a synonym for DOUBLE.
DECIMAL(p,s)	5 to 17 byte	An unpacked floating-point number that cannot be unsigned. In unpacked decimals, each decimal corresponds to one byte. Defining the display length (p) and the number of decimals (s) is required. NUMERIC is a synonym for DECIMAL.

STRING DATA TYPE

DATA TYPE	DESCRIPTION
CHAR(size)	A fixed-length string between 1 and 255 characters in length. Default size is 1.
VARCHAR(size)	A variable-length string between 1 and 255 characters in length; Defining its size is compulsory.
INYTEXT	Holds a string(maximum length of 255 characters)
TEXT	Holds a string(max length of 65,535 characters)
MEDIUMTEXT	String(max length of 16,777,215 characters)
LONGTEXT	String(max length of 4,294,967,295 characters)

DATE & TIME DATA TYPE	
DATE()	A date in YYYY-MM-DD format; For example, May 30th, 2016 would be stored as 12016-05-30.
TIME()	Stores the time in HH:MM:SS format.
DATETIME()	A date and time combination in YYYY-MM-DD HH:MM:SS format.
TIMESTAMP()	Format: (YYYYMMDDHHMMSS).
YEAR()	Stores a year in 2-digit or 4-digit format. If the length is specified as 2 (for example YEAR (2)), YEAR can be 1970 to 2069 (70 to 69). If the length is specified as 4, YEAR can be 1901 to 2155. The default length is 4.

Note : MySQL is not case-sensitive, so you can write SQL commands either in uppercase or lowercase but for UNIX. MySQL is case-sensitive.

Creating and using a Database:

CREATE DATABASE command is used to make a new empty database with the specified name and would not contain any table. Basic syntax for creating a database is as follows:

```
CREATE DATABASE <database_name>;
```

Example:

The code line to create a database called “College”:
mysql>CREATE DATABASE College;

Showing a Database:

To check the names of the existing databases on the server you need to use the SHOW command. SHOW statement displays a list of existing databases on the server.

Syntax: SHOW <expression>;

Example:

```
mysql>SHOW databases;
```

The list of databases shown below by the statement may be different to the other user’s machine.

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| college |
| dhe |
| mysql |
| performance_schema |
+-----+
5 rows in set (0.19 sec)
```

Dropping a Database:

Database can be removed by using DROP command. When a database is removed /deleted, all its contents (tables) get removed along with it!

Syntax : DROP DATABASE <database_name>;

Example, mysql>DROP DATABASE dhe;

The output window would be as follows:

```
mysql> DROP DATABASE dhe;
Query OK, 0 rows affected (0.62 sec)

mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| college |
| mysql |
| performance_schema |
+-----+
4 rows in set (0.16 sec)
```

Using a Database:

Creating database is not enough. In order to put Tables in to it, we need to open it for use. This can be achieved by using the command USE.

Syntax : USE <database_name>;

Example, mysql >USE college ;

Where, USE command makes the specified database as a current working database. At present the current database is 'college'. The output of the above command would be Database changed.

Creating a Table:

Database is a collection of tables. The main purpose for creating tables is to store information in an orderly fashion. Each database table consists of Rows and Columns; just like a graphical tables. The columns specify what data is going to be stored, while the rows contain the actual data. Tables are created using CREATE TABLE statement.

Syntax :

```
CREATE TABLE <TableName>
(
    column_name1 data_type1 (size),
    column_name2 data_type2 (size),
    -----
    -----
    column_nameN data_typeN (size)
) ;
```

Here,

Let us create a table 'STUDENT', with the fields (columns); RollNo, Name, Date of Birth and TelephoneNo, with proper datatypes and size.

```
mysql > CREATE TABLE STUDENT
        (Roll CHAR(8),
         Name VARCHAR(36),
         DOB DATE,
         Tel INT);
```

Rules for Creating Tables

1. The table and column names must start with a letter and can be followed by letters, numbers, or underscores.
2. Table or column names not to exceed a total of 30 characters in length and not use any SQL reserved keywords as names for tables or column names (such as 'select', 'create', 'insert', etc.).
3. It is important to make sure that you are using an open parenthesis before the beginning of a table definition and a closing parenthesis after the end of the last column definition.

4. Separate each column definition with a comma (,).
5. All SQL statements should end with a semi-colon (;).

Display the Table Structure:

DESCRIPTION or DESC command is used to verify the structure of a table that you have created. This command display the column names, available data items with their data types.

Syntax : DESCRIBE <table_name>;

or

DESC <table_name>;

For example, MySQL>DESC STUDENT;

```
mysql> USE COLLEGE;
Database changed
mysql> CREATE TABLE Student
-> (Roll CHAR(8),
-> Name VARCHAR(36),
-> DoB DATE,
-> Tel INT);
Query OK, 0 rows affected (0.50 sec)
```

```
mysql> DESC Student;
```

Field	Type	Null	Key	Default	Extra
Roll	char(8)	YES		NULL	
Name	varchar(36)	YES		NULL	
DoB	date	YES		NULL	
Tel	int(11)	YES		NULL	

```
4 rows in set (0.09 sec)
```

INSERT INTO Statement

The INSERT INTO statement is used to insert records(rows) in a table. It is possible to write the INSERT INTO statement in two ways.

1. The first form does not specify the column names where the data are to be inserted; only their values:

Syntax: INSERT INTO <table_name>
VALUES (value1, value2, value3...);


```
mysql> SELECT Roll,Name FROM Student;
+-----+-----+
| Roll      | Name                |
+-----+-----+
| IA16-001  | Abinash Mohapatra  |
| IC16-009  | Sumitra Nayak      |
| IS16-018  | Gayatri Mallick    |
+-----+-----+
3 rows in set (0.03 sec)
```

Example 2: Suppose you want to retrieve **all columns** from the Student table, write the SELECT statement as follows.

```
SELECT * FROM Student;
```

The output is given below;

```
mysql> SELECT * FROM Student;
+-----+-----+-----+-----+
| Roll      | Name                | DoB          | Tel      |
+-----+-----+-----+-----+
| IA16-001  | Abinash Mohapatra  | 1998-09-09  | 243817   |
| IC16-009  | Sumitra Nayak      | 1996-10-19  | 675432   |
| IS16-018  | Gayatri Mallick    | 2000-06-23  | 612390   |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

Before going to proceed further, let us create another table 'Employee' in this database 'College'. Following the steps described earlier, the table looks like;

```
mysql> SELECT* FROM Employee;
+-----+-----+-----+-----+
| EmpCode   | EmpName              | Pay          | Tel      |
+-----+-----+-----+-----+
| EDN001    | Ratnakar Khuntia    | 96300.00    | 9437023456 |
| EDN002    | Bijayanand Lenka    | 96000.00    | 9950876496 |
| EDN002    | Sudhansu Mohanty    | 86065.00    | 7778889999 |
| EDN004    | Rama Chandra Behera | 23500.00    | 9336780097 |
| EDN005    | Itishree Patnaik    | 12000.00    | 671342138  |
+-----+-----+-----+-----+
5 rows in set (0.02 sec)
```

Display list of Tables in a database:

To see the existing tables on the current database, we use SHOW command as follows:

Syntax: mysql>SHOW tables;

The window shows two tables; 'student' and 'employee' are existing in the 'college' database.

```
mysql> USE COLLEGE;
Database changed
mysql> SHOW TABLES;
+-----+
| Tables_in_college |
+-----+
| employee           |
| student            |
+-----+
2 rows in set (0.06 sec)
```

ALTER TABLE statement

The ALTER TABLE statement changes the structure of an existing table. It is possible to **add / drop a column, modify the datatype of a column, rename a column or a table**. For example.

To Add / Drop / Modify a column in a table, the syntax are;

```
mysql> ALTER TABLE <table_name> ADD <new col_name> datatype;
```

```
mysql> ALTER TABLE <table_name> DROP <existing col_name>;
```

```
mysql>ALTER TABLE<table_name>MODIFY<existing col_name> new datatype;
```

One can use these alterations separately, or combinely to a table. For example let us ADD a new column 'Mark' to the 'Student' table and DROP the 'Tel' column from 'Employee' table. The corresponding syntaxes would be,

```
mysql> ALTER TABLE Student ADD Mark DECIMAL(5,2);
```

```
mysql> ALTER TABLE Employee DROP Tel;
```

The table structures would look like;

```
mysql> DESC Student;
+-----+
| Field | Type           | Null | Key | Default | Extra |
+-----+
| Roll  | char(8)        | YES  |     | NULL    |       |
| Name  | varchar(36)    | YES  |     | NULL    |       |
| DoB   | date           | YES  |     | NULL    |       |
| Tel   | int(11)        | YES  |     | NULL    |       |
| Mark  | decimal(5,2)  | YES  |     | NULL    |       |
+-----+

mysql> desc employee;
+-----+
| Field | Type           | Null | Key | Default | Extra |
+-----+
| EmpCode | char(6)        | YES  |     | NULL    |       |
| EmpName | varchar(27)    | YES  |     | NULL    |       |
| Pay     | decimal(7,2)  | YES  |     | NULL    |       |
+-----+
```

```
mysql> use college;
Database changed
mysql> ALTER TABLE Student MODIFY Mark DECIMAL(4,1);
Query OK, 6 rows affected (3.21 sec)
Records: 6 Duplicates: 0 Warnings: 0

mysql> DESC Student;
```

Field	Type	Null	Key	Default	Extra
Roll	char(8)	YES		NULL	
Name	varchar(36)	YES		NULL	
DoB	date	YES		NULL	
Tel	int(11)	YES		NULL	
Mark	decimal(4,1)	YES		NULL	

```
5 rows in set (0.22 sec)
```

```
mysql> ALTER TABLE Student MODIFY Mark DECIMAL(4,1);
```

By using 'modify' clause, we are able to change the datatype of 'Mark' column from Decimal(5,2) to Decimal(4,1) in the 'Student' table. The new table structure is shown above.

We can change the name of a table by using RENAME TO clause in an ALTER TABLE statement. The syntax is,

```
mysql> ALTER TABLE <old table name> Rename to <New table name>;
```

For example, the statement

```
mysql> ALTER TABLE <Student> Rename to <Student1>;
```

would change the name of the table from 'Student' to 'Student1' in the 'College' database; all the structures and data remaining same.

WHERE clause:

The keyword 'WHERE' is used as a clause in a (SELECT or DELETE or UPDATE) statement to filter records on the basis of one or more criteria.

When a WHERE clause is present, the database program goes through the entire table one row at a time and examines each row to determine whether the criteria is satisfied or not. The row(s) that satisfy the criteria will be displayed in the output.

Using WHERE clause in a SELECT statement:

This type of *SELECT...FROM...WHERE* statement retrieves specific records from a database table. The syntax is,

```
SELECT <column1,column2,...> FROM <table_name>
```

```
WHERE <column> (= or > or any operator..) value or condition;
```

In order to learn the usefulness of different types of *SELECT....WHERE* statements, let us populate the 'Student' tables in the 'College' database.

```
mysql> SELECT *FROM STUDENT;
```

Roll	Name	DoB	Mark
IA16-001	ASHIMA MISHRA	1998-12-27	540.0
IC16-002	SUDHANSU SEKHAR SAHOO	1999-07-05	342.0
IS16-003	SARAH SMRUTI	1999-06-15	351.0
IA16-004	DUANI NAIK	1997-03-03	351.0
IA16-005	SARFRAZ AHMAD	1999-11-23	551.0
IS16-005	AGNAJITA PATNAIK	2000-01-03	554.0
IC16-005	CLEMENT XESS	1996-02-12	252.0
IA16-006	BISMAY PRADHAN	1997-05-15	452.0
IC16-006	PRIYANKA PATRA	1999-11-14	352.0
IS16-010	SAIMA KHAN QUADRI	1999-10-30	552.0

```
10 rows in set (0.00 sec)
```

```
mysql> select *from employee;
```

EmpCode	EmpName	Pay	Age
EDN001	Ratnakar Khuntia	96300.00	NULL
EDN002	Bijayanand Lenka	96000.00	NULL
EDN002	Sudhansu Mohanty	86065.00	NULL
EDN004	Rama Chandra Behera	23500.00	NULL
EDN005	Itishree Patnaik	12000.00	NULL
EDN006	Gunum Patnaik	23450.00	42
EDN007	Kiranmayee Sahu	43400.00	39
EDN008	Sahadev Debata	13400.00	231
EDN010	Daitary Sahoo	33500.00	50
EDN014	Sulagna Pandit	12500.00	36

```
10 rows in set (0.00 sec)
```

Suppose we want to retrieve RollNo, Name and Mark from the Student table, where the mark= 351; The corresponding statement would be,

```
my sql> SELECT Roll, Name, Mark FROM Student WHERE Mark = 351;
```

For retrieving EmployeeCode, Name from the Employee table, where the Age > 45,the statement would be,

```
my sql> SELECT EmpCode,EmpName FROM Employee WHERE Age > 45;
```

```
mysql> SELECT Roll, Name, Mark FROM Student WHERE Mark=351;
+-----+-----+-----+
| Roll      | Name          | Mark  |
+-----+-----+-----+
| IS16-003  | SARAH SMRUTI | 351.0 |
| IA16-004  | DUANI NAIK   | 351.0 |
+-----+-----+-----+
2 rows in set (0.16 sec)

mysql> SELECT EmpCode, EmpName FROM Employee WHERE Age>45;
+-----+-----+
| EmpCode | EmpName      |
+-----+-----+
| EDN008  | Sahadev Debata |
| EDN010  | Daitary Sahoo  |
+-----+-----+
2 rows in set (0.06 sec)
```

Using WHERE clause in an UPDATE statement:

UPDATE command is used with SET and WHERE clauses to modify the data in an existing table. The syntax of this statement is,

```
UPDATE <table_name> SET <col_name> = value WHERE <condition>;
```

Example: The following statement modifies the Age of the employee having EmpCode=EDN008 to 54 in Employee table.

```
mysql> UPDATE Employee SET Age = 54 WHERE EmpCode=EDN008;
```

```
mysql> UPDATE Employee SET Age=54 WHERE EmpCode='EDN008';
Query OK, 1 row affected (0.33 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT *FROM EMPLOYEE;
+-----+-----+-----+-----+
| EmpCode | EmpName          | Pay          | Age |
+-----+-----+-----+-----+
| EDN001  | Ratnakar Khuntia | 96300.00    | NULL |
| EDN002  | Bijayanand Lenka | 96000.00    | NULL |
| EDN002  | Sudhansu Mohanty | 86065.00    | NULL |
| EDN004  | Rama Chandra Behera | 23500.00    | NULL |
| EDN005  | Itishree Patnaik | 12000.00    | NULL |
| EDN006  | Gunum Patnaik    | 23450.00    | 42 |
| EDN007  | Kiranmayee Sahu  | 43400.00    | 39 |
| EDN008  | Sahadev Debata   | 13400.00    | 54 |
| EDN010  | Daitary Sahoo    | 33500.00    | 50 |
| EDN014  | Sulagna Pandit   | 12500.00    | 36 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

Using WHERE clause in an DELETE statement:

DELETE statement is used to delete rows from a table. DELETE removes the entire row, not the individual column values. Care must be taken while using this statement as accidentally important data may get deleted.

Syntax: DELETE FROM <tablename> WHERE <conditionn>;

Example:

Following statement removes the record from the 'Student' table where the Roll No is IA16-004.

```
mysql> DELETE FROM Student WHERE Roll = 'IA16-004';
```

```
mysql> DELETE FROM Student WHERE Roll='IA16-004';
Query OK, 1 row affected (0.16 sec)

mysql> select *from student;
+-----+-----+-----+-----+
| Roll      | Name                | DoB      | Mark  |
+-----+-----+-----+-----+
| IA16-001  | ASHIMA MISHRA      | 1998-12-27 | 540.0 |
| IC16-002  | SUDHANSU SEKHAR SAHOO | 1999-07-05 | 342.0 |
| IS16-003  | SARAH SMRUTI       | 1999-06-15 | 351.0 |
| IA16-005  | SARFRAZ AHMAD      | 1999-11-23 | 551.0 |
| IS16-005  | AGNAJITA PATNAIK  | 2000-01-03 | 554.0 |
| IC16-005  | CLEMENT XESS       | 1996-02-12 | 252.0 |
| IA16-006  | BISMAY PRADHAN     | 1997-05-15 | 452.0 |
| IC16-006  | PRIYANKA PATRA     | 1999-11-14 | 352.0 |
| IS16-010  | SAIMA KHAN QUADRI | 1999-10-30 | 552.0 |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

SELECT DISTINCT Statement:

As we have seen in the 'Student' table, the value 351 occur twice under the 'Mark' column. Likewise a table may contain many such duplicate values in different columns. In order to show the records, where only one instance of the duplicte values appear, we use DISTINCT key word along with the SELECT command. SELECT statement shows all the records;whereas a SELECT DISTINCT statement shows records with uniquw values. The syntax of this statement is,

```
mysql> SELECT DISTINCT <col_name> FROM <table_name>;
```

For example,

```
mysql> SELECT DISTINCT Mark FROM Student;
```

This code would show the following table

```
mysql> SELECT DISTINCT Mark FROM Student;
+-----+
| Mark |
+-----+
| 540.0 |
| 342.0 |
| 351.0 |
| 551.0 |
| 554.0 |
| 252.0 |
| 452.0 |
| 352.0 |
| 552.0 |
+-----+
9 rows in set (0.62 sec)
```

Data Integrity Through Constraints:

We enter data into a database table, by employing 'INSERT INTO' statements. During large data entry, some times erroneous data get typed in to the table, without any proper Error Alert!

If you notice, we have entered same EmpCodes against two of the employees in the 'Employee' table; and one record has the Age is 231. It is not difficult to find a Date column possess a value 31-Feb-2016! In order get around such inadvertent mistakes MySQL imposes some constraints, while creating a table in the database.

Constraints are actually the rules enforced on data columns in a table. These are used to limit the type of data that can go into a table and ensures accuracy and reliability of the data in a database. Basically, constraints maintain the data integrity.

If there is any violation between the constraint and the data action, the action is aborted by the constraint.

Constraints are of two types

(i) Column Constraints: It applies only to individual columns.

(ii) Table Constraints: It applies to groups of one or more columns.

Constraints can be specified when the table is created, by following the syntax given below

```
CREATE TABLE <table_name>
(
column_name1 data_type(size) constraint_name,
column_name2 data_type(size) constraint_name,
-----
-----
);
```

DATABASE INTEGRITY CONSTRAINTS

Some of the database integrity constraints are

1. **Unique Constraints:** It ensures that no two rows have the same value in the specified columns but null values are allowed.

For example,

```
CREATE TABLE Employee (EmpCode CHAR(6) UNIQUE);
```

2. **Default Constraints :** If a user does not enter a value for the column, the defined default value by this constraint is inserted in the field automatically.

For example,

```
CREATE TABLE Employee (Designation CHAR(7) DEFAULT 'Teacher');
```

3. **Primary Key Constraints :** It is similar to unique constraints except that this constraint can be applied to only one column and it does not allow null values.

For example,

```
CREATE TABLE employee (ecode integer PRIMARY KEY, ename char(20));
```

4. **Foreign Key Constraints:** It is a constraint that ensures that columns in one table match the primary key column of same or another table.

For example,

```
CREATE TABLE Items (Order_No integer PRIMARY KEY, Itemno integer REFERENCES Items (Order_No));
```

5. **Check Constraints:** It is used to limit the value range that can be placed in a column.

For example,

```
CREATE TABLE employee (ecode integer PRIMARY KEY, salary decimal CHECK (salary>5000));
```

When a check constraint involves more than one column from the same table, it is specified after all columns have been defined.

For example,

```
CREATE TABLE items (icode char(5), rol integer, quantity integer, CHECK rol<quantity));
```

6. **Not Null Constraint** This constraint ensures that all rows in the table contain a definite value for the column which is specified as not null, which means a null value is not allowed.

For example,

```
CREATE TABLE employee (ecode integer NOT NULL, ename char (20));
```

BASIC ELEMENTS OF MYSQL

The MySQL implementation of SQL has certain elements that play an important role in querying from a database. These elements are very helpful for the user even if he/she wants to write the most elementary SQL queries.

These basic elements are; Literals, Nulls and Comments

Literals

A literal is a constant value written into a query. A literal does not change its value during the execution of a query. Literals can be divided into multiple groups depending upon the type of data it is representing.

1. **Numeric Literal**

These are the sequences of digits which may be positive or negative. *For example, 33, 16, 2005 are all numeric literals. There are four types of numeric literals such as*

- (a) **Integer literals.** *For example, 17 is an integer literal.*

- (b) Floating point or real literals. *For example*, 17.0 and 17.5 are floating point literal.
 - (c) Octal literals.
 - (d) Hexadecimal literals.
2. Character Literal
- These are either a set of alphabets, a set of digits or a set of special symbols and all are enclosed within a pair of single quotation mark (' '). *For example*, 'Ranjan', 'Ajaya', '5', '1991' are all character literals.
3. Boolean Literal
- A Boolean literal can have a value of either 'true' or 'false'. It is used to represent a condition or state of a command.
4. String Literal
- It is a sequence of alphanumeric characters enclosed in double quotation marks (" ").

NULL or Null values

If a column in a table row has no value, then the column is said to be a null. NULL values are treated differently from other values. Nulls can appear in columns of any data type provided they are not restricted by NOT NULL or PRIMARY KEY integrity constraints. It is used as a placeholder for unknown or inapplicable values. NULL is not Zero or blank space or white space. A NULL value means no value has been entered for that column i.e. the value for that column is not known.

To test for NULL values, the IS NULL and IS NOT NULL operators are used.

- (i) IS NULL Operator returns true if column value is NULL.
- (ii) IS NOT NULL Operator returns true if column value is not NULL.

Comments

It is the text that is not executed and is used only for documentation purpose. Comments within SQL statement do not affect the statement execution. Comments are used to clarify the purpose or effect of particular SQL statements. You can also use comment symbols during program development to disable individual statements without deleting them from your source code.

The following table shows the SQL comment indicators that you can enter in your code

Comment Symbol	Description
Double Hyphen (followed by a space) "- "	The double hyphen precedes a comment within a single line. To comment more than one line, put double hyphen symbols at the beginning of each comment line. In MySQL, the "- " comment style requires the second hash to be followed by at least one white space or control character (such as a space, tab, newline and so on).
slash and asterisk "/ * */	C-language style slash and asterisk (<i>/* */</i>) paired delimiters enclose the comment. The <i>/</i> precedes the comment, and the <i>*/</i> follows it. These can delimit single-line or multiple-line comments, but comments cannot be nested.
Hash "#	Begin the comment with <i>#</i> . Proceed with the text of the comment. This text cannot be extended to a new line. End the comment with a line break.

Column Aliass:

So far we saw that the column headings appearing due to the execution of a `SELECT` statement is same as that of its definition in the database table structure. For example, we have defined 'DoB' as the column_name to enter Date of Birth of a student in the 'Student' table. Can we show its heading as 'Birth_Date' without affecting the table structure! Yes, the facility is there in MySQL; called as Column alias. The Column alias is used to temporarily rename a table's column for the purpose of a particular query. This renaming is a temporary change and the actual column name does not change in the database.

Syntax: `SELECT <column_name> AS <alias_name> FROM <table_name>;`

For example, We can show the DoB column of 'Student' table as 'Birth_Date' by using the following `SELECT` statement.

```
mysql> SELECT DOB AS Birth_Date FROM Student;
```

Operators in MySQL

An operator is a symbol that tells the compiler or interpreter to perform specific mathematical, relational or logical operation and produce final result. MySQL supports different types of operators; some of them are described below:

```
mysql> SELECT DoB AS Birth_Date FROM Student;
+-----+
| Birth_Date |
+-----+
| 1998-12-27 |
| 1999-07-05 |
| 1999-06-15 |
| 1999-11-23 |
| 2000-01-03 |
| 1996-02-12 |
| 1997-05-15 |
| 1999-11-14 |
| 1999-10-30 |
+-----+
9 rows in set (0.69 sec)
```

Arithmetic Operators

These operators are used to perform mathematical calculations, such as addition, subtraction, multiplication, division and remainder.

OPERATOR	DESCRIPTION
Addition (+)	Add the two arguments together
Subtraction (-)	Subtract the second argument from the first argument
Multiplication (*)	Multiplies the two arguments
Division(/)	Divide the first argument by the second argument
Modulo (% , mod)	Divide the first argument from the second argument and provides the remainder of that operation

Arithmetic operators can be implemented through simple SELECT statement without any table. This acts like a function. The syntax is,

```
SELECT<Expression1> <arithmetic operator> <Expression2>
```

```
[FROM <table_name>] [WHERE <Condition>];
```

Example-1 To display Employee Code, Employee Name and 20% of Pay for each employee as HRA, we may use the following code line.

```
mysql>SELECT EmpCode, EmpName, Pay*0.20 FROM Employee;
```

```
mysql> SELECT EmpCode, EmpName, Pay*0.20 FROM Employee;
```

EmpCode	EmpName	Pay*0.20
EDN001	Ratnakar Khuntia	19260.0000
EDN002	Bijayanand Lenka	19200.0000
EDN002	Sudhansu Mohanty	17213.0000
EDN004	Rama Chandra Behera	4700.0000
EDN005	Itishree Patnaik	2400.0000
EDN006	Gunum Patnaik	4690.0000
EDN007	Kiranmayee Sahu	8680.0000
EDN008	Sahadev Debata	2680.0000
EDN010	Daitary Sahoo	6700.0000
EDN014	Sulagna Pandit	2500.0000

```
10 rows in set (0.56 sec)
```

Example-2. Addition and Subtraction operators

```
mysql> SELECT 3 + 4 - 5;
```

```
+-----+
```

```
| 3 + 4 - 5 |
```

```
+-----+
```

```
|          2 |
```

```
+-----+
```

Example-3. Multiplication and Division operators

```
mysql> SELECT 3*3/9;
```

```
+-----+
```

```
| 3*3/9 |
```

```
+-----+
| 1.0000 |
+-----+
```

Example-4. Difference between the division and integer division operators

```
mysql> SELECT 9/2, 9 DIV 2;
+-----+-----+
| 9/2  | 9 DIV 2 |
+-----+-----+
| 4.5000 |    4 |
+-----+-----+
```

Example-5. % or mod operator is called the modulo operator

```
mysql> SELECT 11 % 3;
+-----+
| 11 % 3 |
+-----+
|    2 |
+-----+
mysql> SELECT 11 mod 3;
+-----+
| 11 mod 3 |
+-----+
|    2 |
+-----+
```

Comparison Operators

These operators are used to test or compare the value of two operands, *i.e.*, between two variables or between a variable and a constant. If the condition is false, then the result is zero (**0**) and if the condition is true, then the result is non-zero. These operators are also called **relational operators**.

OPERATOR	DESCRIPTION
=	Equal to
>	Greater than
<	Less than
>=	Greater than equal to
<=	Less than equal to
<> or !=	Not equal to

Syntax: `SELECT <column name> FROM <table name> WHERE <expression>
<comparison operator> <expression>;`

Example 1: To display Employee Name and Pay for those employees whose salary greater than or equal to 30000 following code may be used.

```
mysql>SELECT EmpName, Pay FROM Employee WHERE Pay > = 30000;
```

This query produces the following output:

```
mysql> SELECT EmpName, Pay FROM Employee WHERE Pay >= 30000;
+-----+-----+
| EmpName      | Pay      |
+-----+-----+
| Ratnakar Khuntia | 96300.00 |
| Dijayanand Lenka | 96000.00 |
| Sudhansu Mohanty | 86065.00 |
| Kiranmayee Sahu | 43400.00 |
| Daitary Sahoo   | 33500.00 |
+-----+-----+
5 rows in set (0.11 sec)
```

Logical Operators

The logical operators compare two conditions at a time to determine whether a row can be selected for the output. Logical operators are also called Boolean operators, because these operators return a boolean data type value as TRUE, or FALSE. When retrieving data using a SELECT statement, you can use these operators in the WHERE clause, which allows you to combine more than one condition. Some of the Boolean/Logical operators used in MySQL are as follows

OPERATOR	DESCRIPTION
AND	Logical AND compares two expressions and return true, when both expressions are true
OR	Logical OR compares two expressions and return true, when atleast one of the expressions is true
NOT	Logical NOT takes a single expression as an argument and changes its value from false to true or from true to false. You can use an exclamation point (!) in place of this operator

Syntax:

```
SELECT <column names> FROM <table name> WHERE <expressions>
<boolean operator> <expressions>;
```

Example-1:

To display EmployeeCode, Employee Name and Pay for those employees whose Age is less than 45 and Pay is less than 20000.

```
mysql> SELECT EmpCode, EmpName, Pay FROM Employee
        WHERE Age<45 AND Pay >20000;
```

Above query produces the following output:

```
mysql> SELECT EmpCode, EmpName, Pay FROM Employee
-> WHERE Age < 45 AND Pay > 20000;
+-----+-----+-----+
| EmpCode | EmpName      | Pay      |
+-----+-----+-----+
| EDN006  | Gunum Patnaik | 23450.00 |
| EDN007  | Kirannayee Sahu | 43400.00 |
+-----+-----+-----+
2 rows in set (0.69 sec)
```

Example-2: Try out the following queries:

To display EmployeeCode and Name for those employees whose Pay is not greater than 20000 the statement can be

```
mysql>SELECT EmpCode, EmpName FROM Employee WHERE (NOT Pay >20000);
```

Or

```
mysql> SELECT EmpCode, EmpName FROM Employee WHERE !(Pay >20000);
```

Between Operators

The BETWEEN operator is used to show a range of values of the specified column, including the lower and the upper values.

Syntax

```
SELECT <column name> FROM <table name> WHERE
      <column name> BETWEEN <value1> AND <value2>;
```

Example

To display EmployeeCode, Employee Name and Pay for those employees whose Pay lies between 36000 to 90000, we have used the following code;

```
mysql> SELECT EmpCode, EmpName, Pay FROM Employee
      WHERE Pay BETWEEN 30000 AND 96000;
```

The query produces the following output:

```
mysql> SELECT EmpCode, EmpName, Pay FROM Employee
-> WHERE Pay BETWEEN 36000 AND 96000;
+-----+-----+-----+
| EmpCode | EmpName          | Pay      |
+-----+-----+-----+
| EDN002  | Bijayanand Lenka | 96000.00 |
| EDN002  | Sudhansu Mohanty | 86065.00 |
| EDN007  | Kiranmayee Sahu  | 43400.00 |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

IN operator

This operator checks a value within a List (set of values separated by commas) and retrieve the matching rows in the output. The IN operator returns 1 when the search value is present within the set otherwise returns 0. The IN operator also allows to specify multiple values.

Syntax

```
SELECT <col_name> FROM <table_name> WHERE
      <col_name> IN <(value1, value2...)>;
```

Example

To display EmployeeCode, Employee Name and Pay for those employees whose Employee Code are ENN002 & EDN008. We have used the following code;

```
mysql> SELECT EmpCode, EmpName, Pay FROM Employee
        WHERE EmpCode IN ('EDN002', 'EDN008');
```

Above query produces the following output:

```
mysql> SELECT EmpCode, EmpName, Pay FROM Employee
        -> WHERE EmpCode IN ('EDN002', 'EDN008');
```

EmpCode	EmpName	Pay
EDN002	Bijayanand Lenka	96000.00
EDN002	Sudhansu Mohanty	86065.00
EDN008	Sahadev Debata	13400.00

```
3 rows in set (0.00 sec)
```

LIKE Operator

This operator is used to search data, which does not match exactly(as in case of using +, < or > operators) with the criteria. Searching the names starting with 'R' or PinCodes starting with '751' for example, retrieves records from database tables. This is called as pattern-matching. It is useful when we do not know the exact values.

The LIKE operator is used along with the wildcard characters to fetch records, where the data matches a specified pattern.

The wildcard characters used in MySQL are,

- (i) percent sign (%) used to match any **sequence of characters**.
- (ii) underscore (_) used to match a **single character**.

The symbols can also be used combinations.

Syntax

```
SELECT <column_name> FROM <table_name> WHERE
        <column_name> LIKE <pattern>;
```

In the following table we have elaborated the specific use of these wildcard characters, both individually and in combination.

Pattern	Description
'100%'	matches with a string that start with 100
'%100%'	matches with a string that have 100 at any position
'A__AB%'	matches with a string that starts with A and after two characters end with AB.
'__'	matches with a string that atleast 2 characters
'_2%3'	find any values that have a 2 in second position and end with a 3
'2__3'	find any values in a four-digit number that start with 2 and end with 3

Example,

To display RollNo, Name and Mark of the students whose name starts with 'S' in 'Student' table, we have used the following code;

```
mysql> SELECT Roll, Name, Mark FROM Student
        WHERE Name LIKE 'S%';
```

Above query produces the following output

```
mysql> SELECT Roll, Name, Mark FROM Student
-> WHERE Name LIKE 'S%';
+-----+-----+-----+
| Roll   | Name                | Mark  |
+-----+-----+-----+
| IC16-002 | SUDHANSU SEKHAR SAHOO | 342.0 |
| IS16-003 | SARAH SMRUTI         | 351.0 |
| IA16-005 | SARFRAZ AHMAD        | 551.0 |
| IS16-010 | SAIMA KHAN QUADRI   | 552.0 |
+-----+-----+-----+
4 rows in set (0.05 sec)
```

Operator Precedence

MySQL processes expressions according to specific operator precedence. When an expression in an SQL statement is processed, it is evaluated according to the order in which elements are included in the statement.

Precedence is the order in which the operators are evaluated in an expression.

When evaluating an expression containing multiple operators, they are evaluated in order of higher to lower precedence. Following table shows the list of operators in descending order(top to bottom) of precedence. Operators with equal precedence are evaluated from left to right within an expression. Parenthesis can be used to change the precedence of an operator.

ORDER	OPERATOR	DESCRIPTION
1	!	logical Negation
2	-	unary Minus
3	^	bitwise exclusive OR operation
4	*, /, %	Multiplication, Division, Modulo
5	-, +	Subtraction, Addition
6	=, <=, >, >=, != LIKE, IN	all comparison operator
7	BETWEEN	
8	AND, &&	logical AND comparison
9	OR,	logical OR comparison
10	=	Assignment operator

Sorting of records by ORDER BY clause:

When we display the contents of a table, by using SELECT statement, the rows appear in the order of their insertion. However, we can show the records in ascending or descending values of a single column or multiple columns by using ORDER BY clause in the SELECT statement.

The syntax is as follows;

```
SELECT <column name>, [<column name>...]
[WHERE <Condition list>] ORDER BY <column name>;
```

Example:

```
Mysql> SELECT * FROM Student ORDER BY Name;
```

This query would display all the columns of 'Student' table, sorted in ascending order of Names.

By default this statement sorts in ascending order of the values in the specified column. To sort in descending order, one has to use 'DESC' key word after the column name.

```
Mysql> SELECT * FROM Student ORDER BY Name DESC;
```

We can use WHERE clause in the above statements to restrict the records. In the following example, the records would be displayed from the 'Employee' table, in ascending order of Employees' Names, whose Pay is more than 36000.

```
Mysql> SELECT * FROM Employee WHERE Pay > 36000 ORDER BY EmpName;
```

The output of this query is shown below.

```
mysql> SELECT * FROM Employee
-> WHERE Pay > 36000 ORDER BY
-> EmpName;
```

EmpCode	EmpName	Pay	Age
EDN002	Bijayanand Lenka	96000.00	NULL
EDN007	Kiranmayee Sahu	43400.00	39
EDN001	Ratnakar Khuntia	96300.00	NULL
EDN002	Sudhansu Mohanty	86065.00	NULL

```
4 rows in set (0.78 sec)
```

The ORDER BY clause can also be employed on Column alias. We can order the records of a table on the basis of column alias (a temporary name for a column). Following example would illustrate this technique. The column 'Pay' is renamed as 'Salary' and then sorting is done over it. (remember, sorting would be in ascending order, by default)

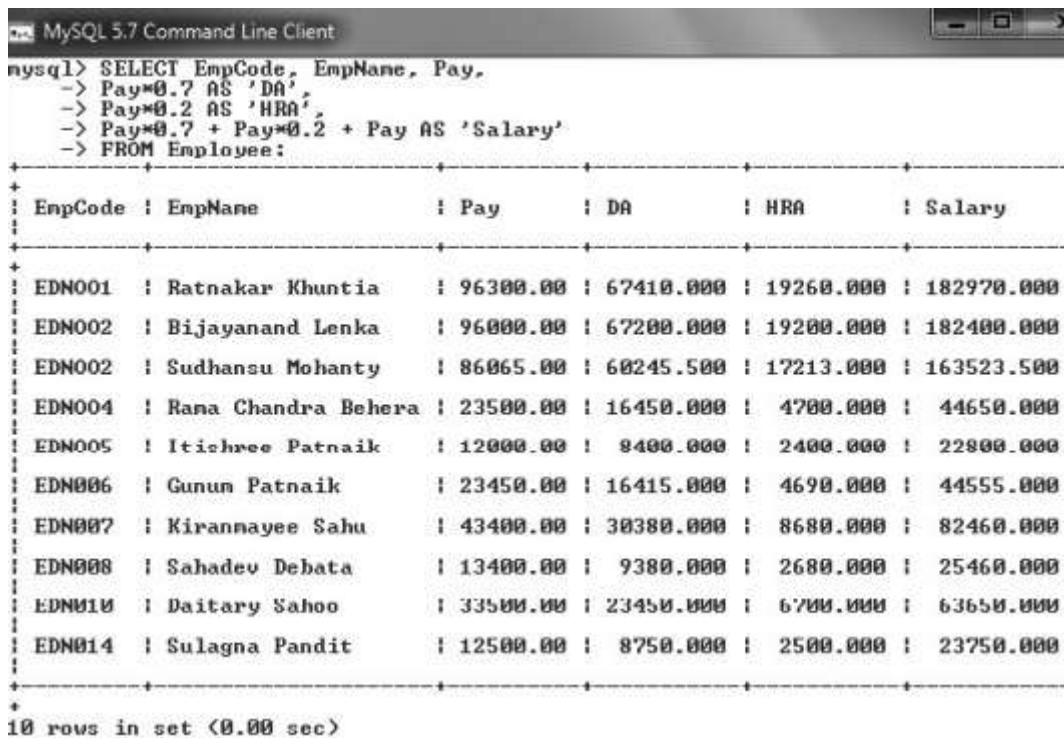
```
Mysql> SELECT EmpName, Pay AS 'Salary'
FROM Employee ORDER BY 'Salary';
```

Showing calculated Columns(fields) using Column alias:

As discussed earlier, we can output calculated fields by using SELECT statement. Here let us use column alias to rename those columns and show them in the output. Carefully go through the following example to learn about this technique.

```
SELECT EmpCode, EmpName, Pay,
       Pay*0.7 AS 'DA', Pay*0.2 AS 'HRA',
       Pay+Pay*0.7+Pay*0.2 AS 'Salary' FROM Employee ;
```

The output of this query is shown in the following table.



```
mysql> SELECT EmpCode, EmpName, Pay,
-> Pay*0.7 AS 'DA',
-> Pay*0.2 AS 'HRA',
-> Pay*0.7 + Pay*0.2 + Pay AS 'Salary'
-> FROM Employee;
```

EmpCode	EmpName	Pay	DA	HRA	Salary
EDN001	Ratnakar Khuntia	96300.000	67410.000	19260.000	182970.000
EDN002	Bijayanand Lenka	96000.000	67200.000	19200.000	182400.000
EDN002	Sudhansu Mohanty	86065.000	60245.500	17213.000	163523.500
EDN004	Rana Chandra Behera	23500.000	16450.000	4700.000	44650.000
EDN005	Itishree Patnaik	12000.000	8400.000	2400.000	22800.000
EDN006	Gunum Patnaik	23450.000	16415.000	4690.000	44555.000
EDN007	Kirannayee Sahu	43400.000	30380.000	8680.000	82460.000
EDN008	Sahadev Debata	13400.000	9380.000	2680.000	25460.000
EDN010	Daitary Sahoo	33500.000	23450.000	6700.000	63650.000
EDN014	Sulagna Pandit	12500.000	8750.000	2500.000	23750.000

10 rows in set (0.00 sec)

EXERCISE**Multiple Choice Questions:**

1. SQL stands for;
(a) Structured Query Language (b) Statistical Query Language
(c) Standard Query Language (d) Structured Query Library
2. SQL was initially developed by;
(a) IBM Inc. (b) Microsoft Inc.
(c) Oracle Inc. (d) Hewlett Packard
3. Which of the following is a DDL command?
(a) CREATE (b) SELECT
(c) COMMIT (d) GRANT
4. Which of the following is a DML command?
(a) ALTER (b) INSERT
(c) SAVEPOINT (d) REVOKE
5. Which of the following is a DCL command?
(a) DROP (b) DELETE
(c) UPDATE (d) ROLLBACK
6. Which of the following is a TCL command?
(a) CREATE (b) SELECT
(c) COMMIT (d) GRANT
7. DELETE command
(a) removes a Database
(b) removes a Table
(c) removes records
(d) all of these
8. IN operator
(a) checks a value in a list (b) matches a substring
(c) corrects a data (d) inserts a column
9. Which statement is used to insert new data in a table?
(a) ADD RECORD (b) INSERT RECORD
(c) INSERT INTO (d) INSERT ROW

10. Which SQL keyword(s) is/are used to sort the rows in the output?
 - (a) SORTED ORDER
 - (b) SORT
 - (c) SORT BY
 - (d) ORDER BY
11. Which statement is appropriate to change the Name “Duani” to “Dunia” in the “Name” column in the ‘Student’ table?
 - (a) UPDATE Student SET Name='Duani' WHERE Name='Dunia' ;
 - (b) MODIFY Student SET Name='Duani' INTO Name='Dunia' ;
 - (c) UPDATE Student SET Name='Duani' INTO Name='Dunia' ;
 - (d) UPDATE Student SET Name='Duani' WHERE Name='Dunia' ;
12. How can you delete the rows with marks below 33 in the ‘Student’ Table?
 - (a) DELETE FROM Student WHERE marks <=351;
 - (b) DELETE marks < 351 FROM Student ;
 - (c) DELETE ROW marks <351FROM Student;
 - (d) DELETE FROM Student WHERE marks <351;

Short Answer Type Questions :

1. What are the differences between DELETE and DROP commands of SQL?
2. Write the MySQL code to
 - (i) Create a database on the MySQL server.
 - (ii) Show table structure.
 - (iii) To delete a database.
 - (iv) To delete a table.
 - (v) Show all data in a table.
 - (vi) Show unique records in a table.
 - (vii) Add a new column in a table.
 - (viii) Change a column name.
 - (ix) Inserting a record in a table.
 - (x) Selecting parts of records:
 - (xi) To list the employee whose names starts an “d” (not “D”).
 - (xii) To list the employee names ending with ‘s’.
 - (xiii) To list the employee code , name and salary in descending order of salary;
 - (xiv) To increase everybody’s salary by 10%.
 - (xv) To change the department of EmpCode of an employee.

3. Differentiate between DDL and DML.
4. Compare CHAR and VARCHAR data types.
5. What is NULL? What happens when you perform arithmetic calculations on NULL values?

Long Answer Type Questions :

1. Discuss the different categories of commands of SQL.
2. What is data type? Explain the numeric Data type in My SQL .
3. Consider a database LOANS with the following table:

Table -Loan Account						
Acct. No	Cust_Name	Loan Amount	Int Rate	Instal-ments	Start_Date	Interest
1	Shreyas Pradhan	300000	12.00	36	19-07-09	
2	S.P. Sharma	500000	10.00	45	22-03-08	
3	A A Dumpy	300000	NULL	36	08-03-07	
4	Sumati Sahoo	800000	12.50	60	06-12-08	
5	Rakesh Bhoi	200000	12.50	36	03-01-10	
6	Trupti Biswal	700000	NULL	60	05-06-08	
7	Element D Souza	500000	10.00	48	05-03-08	

Answer the following questions :

1. Create the table 'LoanAccount' in a new database 'LOAN' and insert tuples in it.
2. Display the details of all the loans
3. Display the AcctNo, Cust_Name, and Loan Amount of all the loans.
4. Display the details of all the loans with less than 40 installments.
5. What value is assigned to a Null field if its value is not assigned in the INSERT INTO statement?



Note :

The 'MySQL' environment can be installed on the PC by downloading the necessary software from

<http://dev.mysql.com/downloads>

and following the link 'Community Server'

Unit

4

IT APPLICATIONS

CHAPTER - 9

e-Governance

CHAPTER - 10

e-Learning

4. IT APPLICATION

Information and Communication Technology (ICT) has revolutionized the socio-economic fabric of the mankind to-day. Be it the financial transactions to speculation, delivery of social welfare benefits to the citizens, social networking and information sharing, like many unthinkable attributes have been catalyzed by this ICT. Scientific research, Entertainments, etc cannot be thought of to-day without IT.

Many applications can be developed to simplify the activities like; playing games, buying movie tickets, making train/air/hotel reservations etc. in on-line mode. These IT applications are broadly categorized as e-Gaming, e-Business, e-Governance, e-Management, e-Learning etc. e-Gaming involves gaming applications on computers or internet, and e-Business involves applications dealing with buying and selling of products and services. E-Governance involves applications which are used for administration and management of an organization. E-Management involves applications dealing with day to day management of organizations. E-Learning involves applications which are developed to help learning of concepts and their applications.

We shall discuss briefly about e-Governance & e-Learning in the next section along with their benefits to the society, their challenges and scopes.

4.A E-GOVERNANCE

E-Governance or Electronic Governance means using Information and Communications Technology (ICT) to transform functioning of the Government. It differs from E-Government as Governance is wider than Government. E-governance refers to governance of a country, or the governance of an institution

or it may be governance of a household by a housewife. However, we shall highlight the first one here for the beginners.

The term 'Governance' is an activity of governing/controlling a country by its Government, controlling of an organization or a company by its CEO or Board of Directors or controlling of a household by the head of the house. Accordingly E-governance may also involve governing of a country, organization, company or a household, however with the help of Information and Communication Technology (ICT). But when we talk of E-Governance in the popular parlance we only refer to the governing of a Country/State using ICT. E-governance therefore means the application of ICT to transform the efficiency, effectiveness, transparency and accountability of exchange of information and transaction:

1. between Governments,
2. between Government agencies,
3. between Government and Citizens
4. between Government and businesses

E-governance also aims to empower people through giving them access to information.

Scope of E-Governance

Governance is all about flow of information between the Government and Citizens (G2C), Government and Businesses (G2B) and Government and Government (G2G).

A. Government to Citizen

Government to Citizen Relationship is the most basic aspect of E-Governance. In modern times, Government deals with many aspects of the life of a citizen. The relation of a citizen with the Government starts with the birth and ends with the death of the citizen. A person transacts with the Government at every corner of his life; be it the birth registration, marriage registration, divorce or death registration. The G2C relation also includes the services provided by the Government to the Citizens, like; Telecommunication, Transportation, Post, Medical facilities, Electricity, Education and also some of the democratic services relating to the citizenship such as Certification, Registration, Licensing, Taxation, Passports, ID Cards etc. Therefore E-Governance in G2C relationship involves facilitation of the services flowing from Government towards Citizens

with the use of Information and Communications Technology (ICT). Following terms may be useful in coming days to categorize different services provided by the Governments to its citizens.

1. E-Citizenship - E-Citizenship includes the implementation of ICT for facilitation of Government Services relating to citizenship of an individual. It involves online transactions relating to issue and renewal of documents like Ration Cards, Passports, Voter Identity Cards, Driving Licenses, and above all the UID (Aadhaar) Cards. It requires the Government to create a virtual identity of every citizen so as to enable them to access the Government services online. For the same, Government has to create a correct, updated and sustainable Citizen Database (like National Population Register) which is a huge task.

2. E-Registration - E-Registration covers online registration of various contracts, an individual enters into during his life-time. Many of these contracts and transactions require registration for giving it legality and enforceability. Such registration may also be made ICT enabled. E-registration will help to reduce a significant amount of paperwork in this process.

3. E-Transportation This service also includes the use of ICT by the Government relating to Transport by Road, Rail, Water or Air. This involves online –

1. booking and cancellation of tickets,
2. status of vehicles, railways, boats and flights,
3. issue and renewal of Driving Licenses,
4. registration and renewal of vehicles,
5. transfer of vehicles,
6. payment of the fees of licenses,
7. payment of fees and taxes for vehicle registration,

4. E-Health – Under this, all hospitals may be brought over one platform by using ICT. A patient database may be created. A local pharmacy database may also be created. These databases can be suitably utilized to maximize benefits to all the stake holders. Telemedicine is an e-Health initiative of the Government at present.

5. E-Education - E-Education covers the implementation of ICT in imparting educational course curriculum in class rooms and in distant mode as

well. Use of internet has reduced the communication time required in Distance education; Internet may also help in conducting online classes.

6. E-Help - E-Help refers to facilitation of disaster and crisis management using ICT. It includes the use of technologies like internet, SMS, etc. for the purpose of reducing the response time of the Government agencies to the disasters. NGOs help Government in providing help in situations of disasters. Online information relating to disasters, warnings and calls for help can help the Government and the NGOs coordinate their work and facilitate and speed up the rescue work.

7. E-Taxation - E-Taxation facilitates the taxing process by implementing ICT. Online tax due alerts and online payment of taxes is helping both the government agencies and the tax payers to transact faster.

C. Government to Government

G2G relationship includes the relationships between Central and State Government and also the relationship between two or more Government departments. Carrying out administration, policing and legal processes on-line have been of great help to all concerned.

1. e-Administration – This process includes the implementation of ICT in the functioning of the Government, internally and externally. Implementation of ICT has reduced the communication time between the Government Departments and Governments. It has substantially reduced the paperwork, brought in efficiency and transparency in the administration of Government Departments.

2. e-Policing - The concept of e-Police is little different from Cyber-Police. Cyber Police require technology experts to curb the electronic/cyber crimes. E-Police refers to the use of ICT for the purpose of facilitating the work of the Police department in investigation and administration. The concept of e-Police includes databases of Police Officers, their performances, Criminal databases – wanted as well as in custody, the trends in crimes and much more. ICT can help reduce the response time of the Police department and also reduce cost by reducing paperwork.

3. e-Courts - The concept of e-Court includes the ICT enablement of the judicial process. Technology has been helping in carrying out distant hearing through video conferencing, online summons and warrants and online publication of Judgments and Decrees.

D. Government to Business

1. e-Licensing - Companies have to acquire various licenses from the Government, similarly the companies have to acquire various registrations. ICT enablement of the licensing and registration can reduce time and cost.

2. e-Tendering – This process includes the facilities of online tendering and procurement. It alerts about any new opportunities of business with the Government and also online submission of tenders and online allotment of work. It reduces time and cost involved in the physical tendering system and checks the mal-practice prevailing in the system. E-Procurement process developed by our own National Informatics Centre has achieved a resounding success.

Objectives of E-Governance

Following are the objectives/aims of E-Governance:

1. To build an informed society – An informed society is an empowered society. Only informed people can make a Government responsible. So providing access to all, every piece of information of the Government and of public importance is one of the basic objective of E-Governance.

2. To increase Government and Citizen Interaction - In the physical world, the Government and Citizens hardly interact. The amount of feedback from and to the citizens is very negligible. E-Governance aims at build a feedback framework, to get feedback from the people and to make the Government aware of people's problems.

3. To encourage Citizen participation - True democracy requires participation of each individual citizen. Increased population has led to representative democracy, which is not democracy in the true sense. E-governance aims to restore democracy to its true meaning by improving citizen participation in the Governing process, by improving the feedback, access to information and overall participation of the citizens in the decision making.

4. To bring Transparency in the Governing process - E-governance carries an objective to make the Governing process transparent by making all the Government data and information available to the people for access. It is to make people know the decisions, and policies of the Government.

5. To make the Government accountable - Government is responsible and answerable for every act decision taken by it. E-Governance aims and will help make the Government more accountable than now by bringing transparency's and making the citizens more informed.

6. To reduce the cost of Governance - E-Governance also aims to reduce cost of governance by cutting down on expenditure on physical delivery of information and services. It aims to do this by cutting down on stationary, which amounts to the most of the government's expenditure. It also does away with the physical communication thereby reducing the time required for communication while reducing cost.

7. To reduce the Reaction time of the Government – Normally due to red-tapism and other reasons, the Government takes long to reply to people's queries and problems. E-Governance aims to reduce the reaction time of the Government to the people's queries and problems, because 's problems are basically Government's problems as Government is for the people.

Advantages of E-Governance

Following are the advantages of E-Governance

1. Speed – Technology makes communication speedier. Internet, Phones, Cell Phones have reduced the time taken in normal communication.

2. Cost Reduction – Most of the Government expenditure is appropriated towards the cost of stationary. Paper-based communication needs lots of stationary, printers, computers, etc. which calls for continuous heavy expenditure. Internet and Phones makes communication cheaper saving valuable money for the Government.

3. Transparency – Use of ICT makes governing profess transparent. All the information of the Government would be made available on the internet. The citizens can see the information whenever they want to see. But this is only possible when every piece of information of the Government is uploaded on the internet and is available for the public to peruse. Current governing process leaves many ways to conceal the information from all the people. ICT helps make the information available online eliminating all the possibilities of concealing of information.

4. Accountability – Once the governing process is made transparent the Government is automatically made accountable. Accountability is answerability of the Government to the people. It is the answerability for the deeds of the Government. An accountable Government is a responsible Government.

Disadvantages of E-Governance

On having many advantages there are some disadvantages too, which are as follows:-

1. Lack of equality in public access to the internet.
2. Reliability of information on the web.
3. Hidden agenda of government groups that can bias public opinions.
4. Vulnerability to cyber attacks and the disturbance to the status quo.
5. Hyper-surveillance lead to lack of privacy.
6. Large amount of money is spent on the development and implementation of e-governance.
7. Inaccessible to the remote areas, have low literacy levels and exist on poverty line income.

Challenges for e-Governance

Almost unknown a decade and half ago, e-government is a prominent concept today in both popular and academic discussions on governance reforms. At the same time, initial expectations about e-government's transformative potential have undergone a critical evaluation. Studies have repeatedly shown that the diffusion of e-government has been slower than expected and the governments are not making full use of the available technology for reforming governance. Wherever adopted, the impact of e-government has been only incremental rather than transformative. A number of challenges have limited the reach and impact of e-government. Several social, economic and political barriers constrain the scope of transformation and restrict the ability of policy makers to make effective use of new technology.

The two most commonly cited constraints are **digital divide** and the political nature of public sector reforms. Universal access to the internet is still far away in many countries. In some countries significant proportion of population cannot afford to access the technology even when it is available. The UN e-government survey notes that as a proportion of monthly income, internet access in the United States is 250 times cheaper than in Nepal and 50 times cheaper than in Sri Lanka. In the United States, 50 per cent of the people use the internet compared to a global average of 6.7 per cent. In Indian sub-continent, it is a mere 0.4 per cent. Studies have shown that even in developed countries online transactional systems have achieved modest and, in many cases, extremely low usage levels. Governance reforms literature shows that bringing openness to public sector agencies and improving bureaucratic accountability often meet with stiff resistance as they require substantial changes in the existing

power structures. These political dynamics partly explain why e-government projects which have governance reforms as their principal target do not impact the system beyond achieving politically neutral goals such as improved access and increased efficiency. E-governance will continue to advance but its pace, spread and impact will be determined by the extent of technology diffusion and the pressure of public demand for reforms.

In India many e-governance projects are citizen-centric, but they could make less than the desired impact due to their limited features. The isolated and less interactive systems revealed major gaps that were thwarting the successful adoption of e-governance along the entire spectrum of governance. They clearly pointed towards the need for a more comprehensive planning and implementation. Adequate infrastructure required to be put in place and interoperability issues to be addressed to establish a more connected government.

E-Governance initiatives by the Government of India:

E-governance initiatives in India took a broader dimension in the mid 1990s for wider IT applications with emphasis on citizen-centric services. The major ICT initiatives of the Government includes some major projects such as Computerization of different operations of Indian Railways, land record computerization, etc. which focused mainly on the development of information systems. Later on, many states started ambitious individual e-governance projects aimed at providing electronic services to citizens.

The national level e-Governance programme called National e-Governance Plan was initiated in 2006. There were 31 Mission Mode Projects under National e-Governance Plan covering a wide range of domains, viz. agriculture, land records, health, education, passports, police, courts, municipalities, commercial taxes, treasuries etc. 24 Mission Mode Projects have been implemented and started delivering either full or partial range of envisaged services. Considering the shortcomings in National e-Governance Plan that included lack of integration amongst Government applications and databases, low degree of government process reengineering, scope for leveraging emerging technologies like mobile, cloud...etc, Government of India has approved the e-Kranti programme recently with the vision of “Transforming e-Governance for Transforming Governance”. All new and on-going e_ Governance projects as well as the existing projects, which are being revamped, should now follow the key principles of e-Kranti such as;

- (i) Transformation and not Translation
- (ii) Integrated Services and not Individual Services
- (iii) Government Process Reengineering (GPR)
- (iv) ICT Infrastructure on Demand
- (v) Cloud by default
- (vi) Fast Tracking Approvals
- (vii) Mandating Standards and Protocols
- (viii) National GIS (Geo-Spatial Information System)
- (ix) Security and Electronic Data Preservation

Idea of establishing Common Service Centres (CSC) to serve the public at large is one of the revolutionary initiatives by the Government of India. The Department of Information Technology, Government of India proposes to facilitate the establishment of a network of more than 100,000 internet enabled Information and Communication Technology (ICT) access points termed Common Service Centres (CSC). The CSCs are meant to provide high quality and cost effective video, voice and data content, in the areas of e-Government, Education, Health, Tele-medicine, Entertainment as well as possible Government and private services.

The goal of the CSC Project is to empower the rural community and catalyse social change through modern technologies. With a large and heterogeneous geographical area, the private sector, if sufficiently encouraged, can play an active role in development and implementation of the CSC Project in supplementing the Governments' efforts to realise its vision for the project.

The CSC envisages a bottom-up model for delivery of content, services, information and knowledge, that can allow like-minded public and private enterprises - through a collaborative framework - to integrate their goals of profit as well as social objectives, into a sustainable business model for achieving rapid socio-economic change in rural India. It is the community participation and collective action, not ICT alone, which will lead to a behaviour change for a sustainable socio-economic change and long-term rural prosperity.

E-Governance initiatives by the Government of Odisha:

Government of Odisha intends to increase the efficiency across the Administrative Structure; thereby improving the quality of services to the citizens through a system of Information and Communication Technologies (ICT)

involving the least manual interface. It has been well recognized that Electronic governance (e-Governance) is as an enabler to good governance in the State of Odisha. Apart from individual departments, Odisha Computer Application Centre (OCAC), the Designated Technical Directorate of Information Technology Department, Government of Odisha, has evolved through years as a center of excellence in Training, IT solutions and e-Governance. It has contributed significantly to the steady growth of IT in the state. Its web address is <http://www.ocac.in/>. Under the technical support of OCAC other portals are catering to various needs of the citizens, students, business establishments, etc of the state. Few of them are outlined here.

OSWAN - The objective of OSWAN is to create a state wide area network that will improve government efficiency through connectivity. The Odisha State Wide Area Network (OSWAN) would serve as the backbone network for data, video and voice communications throughout the State which includes 1 State Head Quarter (SHQ), 30 District Head Quarters (DHQ), 284 Block Head Quarters (BHQ) and 61 Horizontal Offices. Its web address is <http://www.oswan.gov.in/>

E-dhaRani: The department of Revenue and Disaster Management appreciates the importance of e-Governance and has undertaken e-dhaRani, a comprehensive project of computerization of all Registration offices across the state. A significant capacity in terms of physical and technology infrastructure has therefore been built to implement e-dhaRani in a competent manner, with a holistic perspective and with speed. Its web address is [http://www.igrodisha.gov.in/\(e-Registration\)](http://www.igrodisha.gov.in/(e-Registration)). A similar attempt has been made to computerize the land records of the state through another portal; <http://bhulekh.ori.nic.in/>

E-Shishu: This project is designed to achieve the three basic goals of Sarva Shikshya Abhiyan /DPEP (District Primary Education Programme). They are:

- (i) *Access*: By tracking the out of school children with their age and reason of being Out of school & bring them back into the mainstream education.
- (ii) *Retention*: By tracking the In-school children and providing necessary inputs so that they continue to remain in the school.
- (iii) *Quality of Education*: By tracking the achievement level of children and taking corrective measures for it's enhancement

The web address of this project is <http://www.opepa.in/website/e-sishu.aspx>

OdishaOnline: This portal provides the facilities of on-line payment of Electricity Bill, Water Bill, and Holding Tax; on-line booking of OSRTC Bus-tickets, Applying for Birth & Death Certificates in the state of Odisha. At present it is operational only in select cities of the state. The address of this portal is: <https://www.odishaonline.gov.in/>.

Students Academic Management System (SAMS): This portal provides the facilities of on-line application for admission in to Higher Secondary (+2 level) as well as Degree (+3 level) classes in different Colleges and Higher Secondary Schools under the Council of Higher Secondary Education, Odisha and different Universities of the state. In a single application form (On-line), an applicant can apply for 06 or more options (colleges or streams taken together), sitting at the comfort of one's home or at any place having internet facility. Each institution under SAMS is provided with an individual and authorized (password protected) **e-Space**; through which all the academic details of the students are meticulously maintained. The address of this portal is: <sams.dheodisha.gov.in/>.



Figure 4.1 : Web portal of SAMS intranet

E-Governance web sites

The website of the Ministry of Information Technology (MIT), Govt. of India lists briefly the E-Governance Initiatives undertaken by the various Ministries/Departments and States Governments. For details one can log onto: <http://www.mit.gov.in/eg/ms.asp>. Apart from offering e-governance solutions and services, the Centre showcases applications by various IT majors like C-DAC, CMC, IQ-Virtuals, Microsoft, NIC and Oracle. The Centre aims to offer services such as technical consultation, proof of concept and thematic presentations, apart from creating awareness amongst decision makers at the Central and State level and helping them in defining and implementing process and policy changes for effective e-Governance. Some of the Central and State Government websites are outlined below.

National Informatics Centre:

NIC is a premier IT organization in India which provides state-of-the-art solutions for the IT needs of the Government at all levels. The NIC home page gives access to a comprehensive directory (<http://goidirectory.nic.in/>) providing links to web sites of Government of India Ministries, Departments, States, Organizations, Union Territories. For further details log on to: <http://www.nic.in/>

Parliament of India Website of the Indian Parliament carries information on the proceedings of both the houses of the parliament, the Constitution of India, various budgets, resume of work, parliamentary debates, secretariats, committees and members of the houses, and links to other central and state government websites. For further details log on to: <http://alfa.nic.in/>

Department of Administrative Reforms and Public Grievances This is a section of the website of the Department of Administrative Reforms and Public Grievances under the Ministry of Personnel, Public Grievances and Pensions of the Indian Government. It has well categorized links to E-Governance initiatives from around the world and a review of the Indian experience with E-Governance. For further details log on to: <http://persmin.nic.in/arp/egov1.htm>

Government of Odisha web portal: The official website of the Government of Odisha provides links to individual Government Departments (Home, Revenue, Finance, Higher & Mass Education, Tourism, Forest, etc.) District Collectorate and Urban Local Bodies in the state. The web-address is <http://www.orissa.gov.in/portal/>.

Government of Kerala web portal: This site provides access to government services by providing online application forms for services from various departments like the electricity board, revenue department, corporations, public distribution system. Besides facilitating online application for grievance redressal, the site carries information of government machinery, contact details of elected representatives and a discussion forum. It also provides links to various govt., NGOs and educational institutions. For further details log on to: <http://www.keralagov.com/>

Rajasthan Government: A section of the official website of the Government of Rajasthan devoted to IT and Electronic Governance issues. For further details log on to: <http://www.rajgovt.org>

Andhra Pradesh Government: Official website of the Andhra Pradesh Government, which describes the state's IT initiatives towards Electronic Governance. For further details log on to: <http://www.andhrapradesh.com/>

Government of West Bengal: Official site of the West Bengal Govt. features the e-governance initiatives to be undertaken by the state govt. Initially for better citizen interface the following will be emphasized for computerization, viz., vehicle registration, land records, birth and death registrations, employment exchanges, payment of excise duty, sales tax and local tax, electronic bill payment of water and electricity, computerization of health records, relevant education through internet, work of Police and Police Stations, Criminal Justice etc. For further details log on to: <http://www.westbengal.gov.in/>

Government of Madhya Pradesh: The portal features state IT policies and initiatives by the Madhya Pradesh government. For further details log on to: <http://www.mpgovt.nic.in/>

Government of Maharashtra: The site features IT initiatives proposed by the Maharashtra government. These include Computerization of Government departments with very high public contact (e.g. sales tax, revenue, health, education, irrigation, PWD, etc.), Computerization of Government departments with very high public contact (e.g. sales tax, revenue, health, education, irrigation, PWD, etc.) For further details log on to: <http://www.maharashtra.gov.in/>

EXERCISE**Short answer type Questions :**

1. UID Card refers to _____ Card.
2. E-Procurement software is developed by OCAC. *True/False*
3. Digital divide is a
 - (a) Disadvantage
 - (b) Challenge
 - (c) Objective
 - (d) Advantage
4. Odishaonline is a
 - (a) Child Tracking Portal
 - (b) Disaster Management Portal
 - (c) Land Records Portal
 - (d) On-line Payment Portal
5. Expand the abbreviations: C-DAC, ICT, GIS, G2C , SAMS, OCAC, NIC, DPEP, SSA

Long answer type Questions :

1. What is e _ Governance? Discuss the scope of e _ Governance.
2. Discuss the advantages and disadvantages of e _ Governance.
3. Briefly describe few e_ initiatives of Government of Odisha.
4. Outline the address of some of the Central and State Government Portals.
5. What are the principles of e _ Kranti?



CHAPTER - 10

E-LEARNING

4.B E-LEARNING

E-Learning is a process, which allows you to learn anywhere and usually at any time, as long as you have a properly configured computer. E-learning can be CD-ROM-based, Network-based, Intranet-based or Internet-based. It can include text, video, audio, animation and virtual environments. It can be a very rich learning experience that can even surpass the level of training you might experience in a crowded classroom. It's self-paced, hands-on learning.

The quality of the electronic-based training, as in every form of training, is in its content and its delivery. E-learning can suffer from many of the same pitfalls as classroom training, such as boring slides, monotonous speech, and little opportunity for interaction. The beauty of e-learning, however, is that new software allows the creation of very effective learning environments that can engulf you in the material.

E-learning falls into four categories, from the very basic to the very advanced. The categories are:

1. **Knowledge databases:** While not necessarily seen as actual training, these databases are the most basic form of e-learning. You've probably seen knowledge databases on software sites offering indexed explanations and guidance for software questions, along with step-by-step instructions for performing specific tasks. These are usually moderately interactive, meaning that you can either type in a key word or phrase to search the database, or make a selection from an alphabetical list.
2. **Online support:** Online support is also a form of e-learning and functions in a similar manner to knowledge databases. Online support comes in the form of forums, chat rooms, online bulletin boards, e-mail, or live instant-messaging support. Slightly more interactive than knowledge databases, online support offers the opportunity for more specific questions and answers, as well as more immediate answers.

3. **Asynchronous training:** This is e-learning in the more traditional sense of the word. It involves self-paced learning, CD-ROM-based, Network-based, Intranet-based or Internet-based. It may include access to instructors through online bulletin boards, online discussion groups and e-mail. Or, it may be totally self-contained with links to reference materials in place of a live instructor.
4. **Synchronous training:** Synchronous training is done in real-time with a live instructor facilitating the training. Everyone logs in at a set time and can communicate directly with the instructor and with each other. You can raise your cyber hand and even view the cyber whiteboard. It lasts for a set amount of time - from a single session to several weeks, months or even years. This type of training usually takes place via Internet Web sites, audio- or video-conferencing, Internet telephony, or even two-way live broadcasts to students in a classroom.

Benefits of e-Learning:

E-learning has definite benefits over traditional classroom training. While the most obvious are the flexibility and the cost savings (not having to travel or spend excess time away from work), there are also others; for example:

- It's **less expensive** to produce - Using software packages to produce your own asynchronous training programs, e-training is virtually free once you reach the break-even point. Synchronous programs will have continued costs associated with the instructor managing the class, but will still be lower than traditional courses.
- It's **self-paced** - Most e-learning programs can be taken when needed. The "books" that you set up create a module-based design allowing the learner to go through smaller chunks of training that can be used and absorbed for a while before moving on.
- It **moves faster** - e-learning courses progress up to 50 percent faster than traditional courses. This is partly because the individualized approach allows learners to skip material they already know and understand and move onto the issues they need training on.
- It provides a **consistent message** - E-learning eliminates the problems associated with different instructors teaching slightly different material on the same subject.
- It can work from **any location and any time** - E-learners can go through training sessions from anywhere, usually at anytime. This Just-In-Time (JIT) benefit can make learning possible for people who never would have been able to work it into their schedules prior to the development of e-learning.

- It can be **updated easily and quickly** - Online e-learning sessions are especially easy to keep up-to-date because the updated materials are simply uploaded to a server. CD-ROM-based programs may be slightly more expensive to update and distribute, but still come out cheaper than reprinting manuals and retraining instructors.
- It can lead to **increased retention and a stronger grasp on the subject** - This is because of the many elements that are combined in e-learning to reinforce the message, such as video, audio, quizzes, interaction, etc. There is also the ability to revisit or replay sections of the training that might not have been clear the first time around.
- It can be **easily managed** for large groups of students. It allows corporate training directors, HR managers and others to keep track of the course offerings, schedule or assign training for employees and track their progress and results. Teachers can review a student's scores and identify any areas that need additional training.

E learning web-sites and their salient features:

In the recent past, many websites have been developed with focus area being interactive teaching and learning. Few of them are listed below. All these websites helps students learning the latest advancement in the respective fields and also enhance their listening abilities, understanding abilities and makes them on par with the students studying in the top universities around the world.

National Programme on Technology Enhanced Learning (NPTEL):

This is a website maintained by the IIT's present in the country and also provides questionnaire to answer so that students can be benefited. NPTEL provides E-learning through online Web and Video courses in Engineering, Science and humanities streams. The mission of NPTEL is to enhance the quality of engineering education in the country by providing free online courseware. To know more, visit <http://nptel.iitm.ac.in/>

Virtual Learning Environment(VLE), Delhi University: The Institute of Life Long Learnung (ILL) of Delhi University has uploaded over 2000 e_Lessons (335 lessons on Political Science, 183 lessons on Hindi, 192 lessons on Botany, 151 lessons on Physics and 78 lessons on Zoology), which are freely available at the Virtual Learning Environment (VLE). To know more, visit <http://vle.du.ac.in/>

Khan Academy: In this website, one initially need to register (which is FREE) and then you can have complete access to all the courses available in the site. To know more, visit <https://www.khanacademy.org/>

Coursera: Coursera is an education platform that partners with top universities and organizations worldwide, to offer courses online for anyone to

take, for free. It covers a wide range of sectors. To know more, visit <https://www.coursera.org/>

MIT OpenCourseWare (OCW): MIT OpenCourseWare is a web-based publication of virtually all Massachusetts Institute of Technology (MIT) course content. OCW is open and available to the world and is a permanent MIT activity. It covers a wide range of domains including cross sectoral areas. To know more, visit <http://ocw.mit.edu/>

Open Culture: This is a site which provides information regarding online courses available in multiple subjects including Computer Sciences Engineering. To know more, visit <http://www.openculture.com/>

Educational Technology Services, Berkeley University: Educational Technology Services is Berkeley's central campus resource providing technology tools, training and support for faculty, staff and students in their work as teachers, learners, researchers and collaborators. To know more, visit <http://ets.berkeley.edu/>

Umass Boston OpenCourseWare: The UMass Boston Open Courseware provides various e-learning courses covering various subjects. To know more, visit <http://ocw.umb.edu/>

Open University: The Open University offers courses through distance learning at all levels. To know more, visit <http://www.open.edu/openlearn/science-maths-technology>

Utah State Open Course Ware: Utah State Open Courseware is a collection of educational material used in the formal campus courses of Utah State University. It seeks to provide people around the world with an opportunity to access high quality learning opportunities. To know more, visit <http://ocw.usu.edu/>

Cosmo Learning: Created with the goal to provide the world with a free online school, CosmoLearning (CL) is an educational website committed to improving the quality of homeschooling, teaching and student excellence. Designed for educators and self-learners, organized according to traditional curriculum standards, CosmoLearning provides courses, video lectures, documentaries, images, books and other multimedia in dozens of subjects. To know more, visit <http://www.cosmolearning.com/>

EdX: EdX offers interactive online classes and MOOCs from the world's best universities. Topics include biology, business, chemistry, computer science, economics, finance, electronics, engineering, food and nutrition, history, humanities, law, literature, maths, medicine, music, philosophy, physics, science, statistics and more. To know more, visit <https://www.edx.org/>

Dave Conservatoire: It is an entirely free online music school offering a self-proclaimed "world-class music education for everyone," and providing video lessons and practice tests.

Codecademy — Codecademy offers data science and software programming (mostly Web-related) courses for various ages groups, with an in-browser coding console for some offerings.

Stanford Engineering Everywhere — SEE/ Stanford Engineering Everywhere houses engineering (software and otherwise) classes that are free to students and educators, with materials that include course syllabi, lecture videos, homework, exams and more.

Big Data University — Big Data University covers Big Data analysis and data science via free and paid courses developed by teachers and professionals.

Better Explained — BetterExplained offers a big-picture-first approach to learning mathematics — often with visual explanations — whether for high school algebra or college-level calculus, statistics and other related topics.

HOW Design University — How Design University (How U) offers free and paid online lessons on graphic and interactive design, and has opportunities for those who would like to teach.

HTML Dog — HTML Dog is specifically focused on Web development tutorials for HTML, CSS and JavaScript coding skills.

Skill crush — Skillcrush offers professional web design and development courses aimed at one who is interested in the field, regardless of their background — with short, easy-to-consume modules and a 3-month Career Blueprints to help students focus on their career priorities.

Udacity — Udacity offers courses with paid certification and nanodegrees — with emphasis on skills desired by tech companies in Silicon Valley — mostly based on a monthly subscription, with access to course materials (print, videos) available for free.

Apple Developer Site — Apple Developer Center may be very specific in topics for lessons, but it's a free source of documentation and tutorials for software developers who want to develop apps for iOS Mobile, Mac OS X desktop, and Safari Web apps.

Google Code — As with Apple Developer Center, Google Code is topic-narrow but a good source of documentation and tutorials for Android app development.

Code.org — Code.org is the home of the “Hour of Code” campaign, which is aimed at teachers and educators as well as students of all ages (4-104) who want to teach or learn, respectively, computer programming and do not know where to start.

Mozilla Developer Network — MDN (Mozilla Developer Network) offers learning resources — including links to offsite guides — and tutorials for Web

development in HTML, CSS and JavaScript — whether you're a beginner or an expert, and even if you're not using Mozilla's Firefox Web browser.

Government of India Initiative on e-Learning: SWAYAM

Under 'National Mission on Education through Information and Communication Technology' (NMEICT) Scheme, Ministry of Human Resource Development (MHRD) has planned to leverage the potential of Information and Communication Technology (ICT) in providing high quality personalized and interactive knowledge modules over the internet/ intranet for all the learners in Higher Education Institutions in anytime any where mode. The thrust area is to include 'Massive Online Open Courses (MOOCs)' that shall be developed and leveraged for e-education.

Accordingly the MHRD embarked on a major and new initiative called 'Study Webs of Active Learning for Young Aspiring Minds' (SWAYAM), which will provide one integrated platform and portal for online courses, using ICT and covering all higher education subjects and skill sector courses to ensure that the every student in our country has access to the best quality higher education at an affordable cost.

In the first phase of SWAYAM, IIT-Mumbai, IIT- Chennai, IIT-Kanpur, IIT-Guwahati, University of Delhi, Jawaharlal Nehru University and IIM Bangalore, among others, will offer courses in management, social sciences, basic sciences, engineering education and energy.

SWAYAM would provide

1. High quality learning experience using multimedia on anytime, anywhere basis.
2. One-stop web location for interactive e-content for all courses from School to University level.
3. State of the art system that allows easy access, monitoring and certification.
4. Peer group interaction and discussion forum to clarify doubts.
5. Hybrid model that adds to the quality of classroom teaching

SARANSH : Saransh is an online platform for a comprehensive self-review and analysis of student's performance and progress. It is a digital interface portal offering a one-to-one interaction platform between teachers and the parents of their students. It works on a data-driven decision support system to help parents in evaluating the strengths and weaknesses of their children and make them better informed about the child's likes and dislikes in turn enabling them to take informed decisions about their future.

Saransh has been recently launched by the Ministry of Human Resource and Development as one of the many digital initiatives under '**Digital India**'

campaign, to promote Information and Communication Technologies in CBSE affiliated schools and bring transparency in the existing educational system in India. It also provides information regarding competitions and attendance that can now be directly shared with parents through this digital platform. It also makes available the **e-books** of all standards and subjects. The schools can use this tool for analyzing subject performance of all students and identify the fields where the child needs improvement. The portal is equipped with 3 preceding years' data of all schools. Hence, a comparative study, like year-on-year, can be done to find the well-performed areas of the student and fields that further needs to be improved.

Societal impact of e-Learning

Effective eLearning comes from using information communications technology (ICT) to broaden educational opportunity and help students develop the skills they need to thrive in the 21st century. E-Learning can deliver substantial positive effects as outlined below:

- (a) Students are more engaged and able to develop 21st century skills.
- (b) Teachers have a more positive attitude toward their work and are able to provide more personalized learning.
- (c) Family interaction and parental involvement may increase.
- (d) Communities benefit by bridging the digital divide.
- (e) Economically disadvantaged students and children with disabilities benefit particularly.
- (f) Economic progress can result from direct job creation in the technology industry as well as from developing a better educated workforce.

An effective eLearning can produce promising effects. Research seems to indicate that a more technology-rich environment delivers greater impacts. It has helped to transform innovation-driven entrepreneurship from the point of aspiration to the point of reality for the people of India. In addition to the direct economic benefits, ICT investment that enables teachers and students to develop strong competency in the teaching and using of 21st century skills can create more competitive workforce in the long run. Technological advances affect every aspect of the modern society. In an increasingly digitized world, individuals who develop the proficiencies necessary to effectively utilize these digital technologies will be at an increasing advantage in educational opportunities, workforce development, and participating fully in society. Evidence indicates that investments in e-Learning can deliver substantial positive effects. For stakeholders engaged in discussions in education reform, it is critical that a wide spectrum of research may be considered to help inform national policies as well as local level plans.

EXERCISE**Short answer type Questions:**

1. Real time training falls under Asynchronous / Synchronous training.
2. Dave Coservatoire is an onlinr portal offering courses on_____. *Math/ Music*
3. SWAYAM stands for
 - (a) Sequential Way of Amplitude Modulation
 - (b) Study Webs of Active Learning for Young Aspiring Minds
 - (c) State’s Wireless Application Yojana for Assisted Mechanism
 - (d) Students’ Web-portal for Academic Year Acquired Marks
4. SARANSH is a digital interactive portal offering interaction between
 - (a) Child and Parents
 - (b) Student and Teachers
 - (c) Student and Peers
 - (d) Parent and Teachers
5. Expand the abbreviations: MOOC, ICT, MDN, OCW , JIT, NPTEL, MHRD

Long answer type Questions:

1. What is e _ Learning? Discuss its four categories.
2. Discuss the societal impact of e _ Learning.
3. Briefly describe the initiatives of Government of India to promote e _ Learning .
4. Outline the address of some of the e _ Learning Portals.
5. What is the objective of SARANSH?

