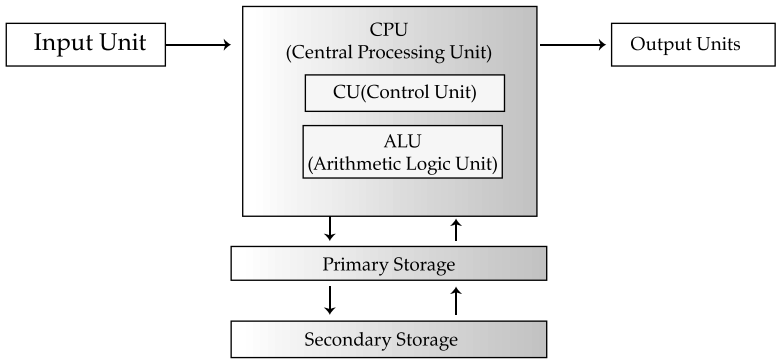
UNIT-I
Computer Systems and Organisations

CHAPTER-1
BASIC COMPUTER ORGANISATION

TOPIC-1
Computer System

Revision Notes

- A computer is an electronic device that can perform a variety of operations in accordance with a set of instructions called program. In other words, we can say that computer is an electronic device which manipulates the data according to the list of instructions.



- **Strengths of Computer :**
  - Much faster as compared to human beings.
  - Can store a large amount of information in very small space.
  - Immune to tiredness.
  - Computers can perform repetitive jobs efficiently.
- **Weaknesses of Computer :**
  - Computers cannot take decisions on their own.
  - They need to be told each and every step.
  - Need special languages.
- A Computer works on the IPO principle ie.

  Input -> Process -> Output.
- Based on this these, use some components to perform input, some components to do processing and some components provide output
- **Input Unit :** It takes the input i.e. instructions from the user and converts it in the binary form so that it can be understood by the computer. Input unit is formed by the input devices connected with the computer, e.g : keyboard, mouse, etc.
  - **Central Processing Unit (CPU) :** It is known as the brain of the computer. It has three components :
  - **Control Unit (CU) :** It controls and guides interpretation, flow and manipulation of data and information. It sends control signals until the required operations are properly done by ALU and memory unit (storage unit).
- **Arithmetic Logic Unit (ALU) :** It performs all the required arithmetic and logical operations to perform the task.
- **Memory Unit (Storage Unit) :** It comprises of primary and secondary memory unit.

- **Hardware :** All the physical components of the computer system are known as hardware. e.g.: keyboard, mouse, monitor, printer, etc.
- **Software :** Software represents the set of programs that govern the operations of a computer system and make the hardware run smoothly. Software can be classified in three types :
  - **System Software :** These software are mandatory for all computer systems to work. For example : Operating systems like BOSS, Windows and language processors like assembler etc.
  - **Application Software :** These software are made to perform the specific task. For example : WordPad, MS-Word etc.
- * Utility Software : Software that assists the computers to perform functions like data backup, virus removal, scanning, defragmenting, etc. For example: Antivirus, disk defragmenter, etc. .

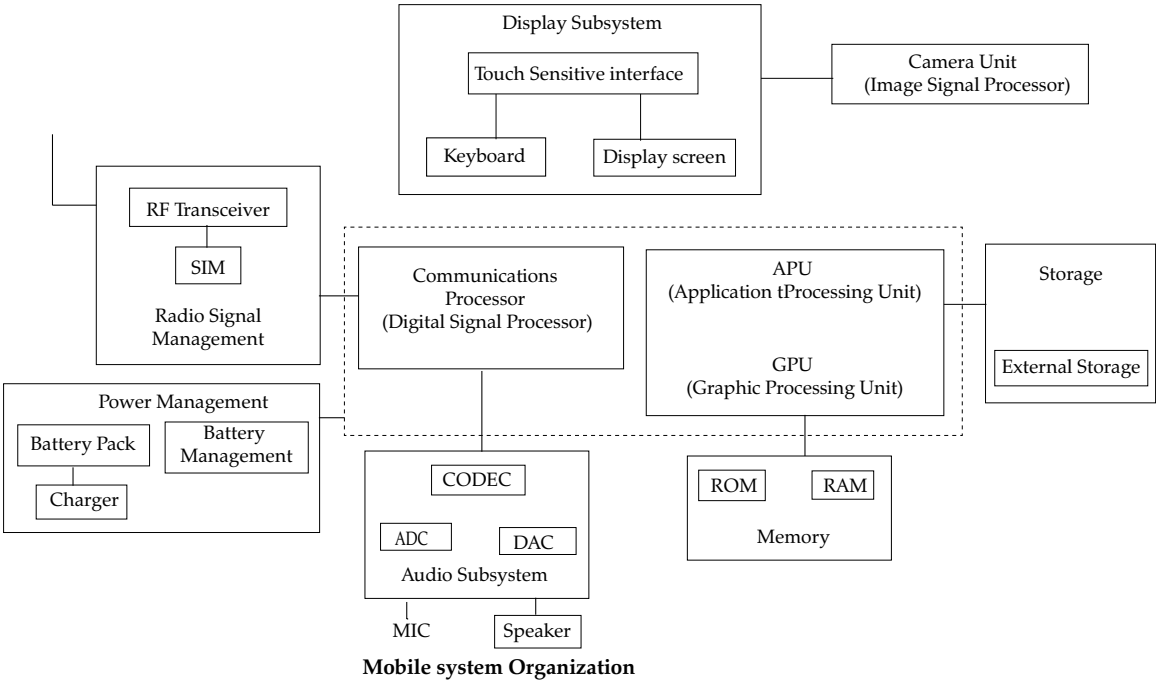
Know the Terms

- **Data :** Data is the collection of facts and figures.
- **Information :** Meaningful content extracted from data.
- **Program :** Set of instructions that direct the computer.
- **Commands :** Special codes or keywords that the user inputs to perform a task.
- **Input :** Process of entering data and instructions in the computer.
- **Storage :** Saving data and instructions to make them readily available for future use.
- **Process :** Performing operations (arithmetic or logical) on data to convert them in useful information.
- **Output :** The process of producing useful information or result for the user

[1 mark for each]

TOPIC-2
Mobile System

Revision Notes



Mobile system Organization

- As a mobile system runs on battery power, a mobile's CPU has a little power.
- Given below are the basic components of a **mobile system**.
- **Mobile Processor** – It receives commands, makes instant calculation, plays audio / video, stores information and sends signals throughout the device. It has following two sub units.

- **Communications Processing Unit** is responsible for phone calls management. Its digital signal processor helps it work with RF Transceiver and Audio subsystem.
- **Application Processing Unit** governs controlling all types of operations of a mobile system.

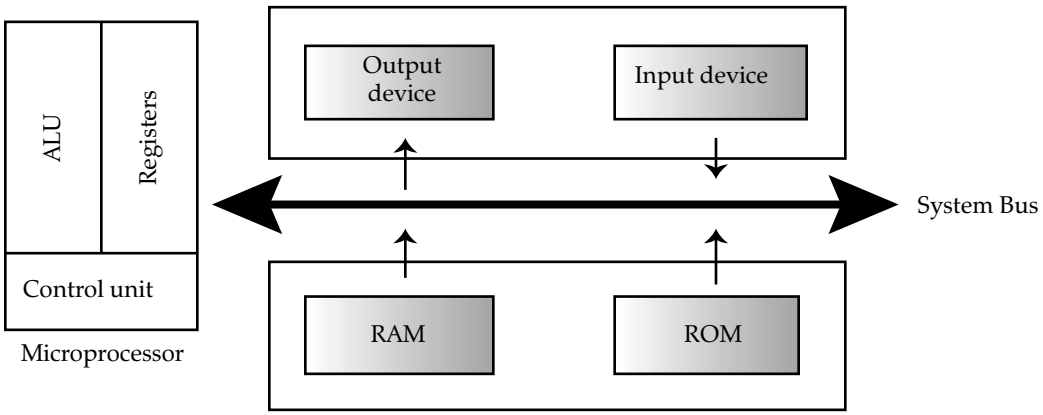
- Mobile system Memory is of two types
  - RAM (Random Access Memory) is the work memory of mobile system and does not store information once the device is turned off.
  - ROM (Read Only Memory) is part of internal storage. It is Flash memory or EEPROM. It has operating system and some preinstalled apps on it.
  - Display Subsystem provides display, touch sensitive interface and keyboards.
  - Camera subsystem has an integrated Image Signal Processor, ensures instant image capture, high-resolution support, image Stabilization and other image enhancements.
  - Radio Signal Management Unit connects SIM to the base stations through radio signals.
- Storage or external storage or expandable storage comes in the form of SD or micro SD cards.
- Power Management Subsystem provides power to a mobile system.
- A mobile phone is a handheld wireless device that allows users to make and receive calls and to send text messages.
- A mobile phone is also known as a cellular phone or a cell phone.
- Mobile phones belonging to GSM (Global system for Mobile Communications) network became capable of sending and receiving text messages.
- MMS- Multimedia Messaging Service is used to send and receive images.
- A mobile phone with advanced features similar to a computer is called a smartphone.
- The cellular network is composed of cell sites scattered throughout the world. Mobile phone operates on a cellular network.

TOPIC-3
Central Processing Unit (CPU)

Revision Notes

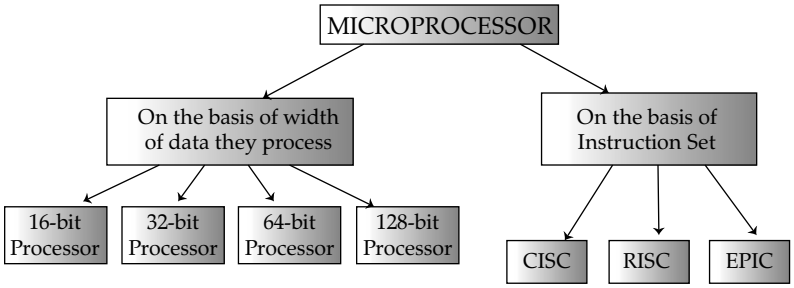
- A microprocessor, also called CPU, is a multipurpose, programmable logic device that reads binary instructions from a storage device called memory, accepts binary data as input and processes data according to those instructions and provides results as output.

Block diagram of microprocessor



- Components of Microprocessor :
  1. CPU **2.** Bus
- **CPU :** it is an acronym for Central Processing Unit. It controls all the operations of a computer. It has three components :
  (i) Arithmetic Logic Unit (ALU)
  (ii) Control Unit (CU)
  (iii) Registers
- **Bus :** It refers to the collection of wires through which data is transmitted from one part of computer to another. It is of 2 types : (i) Internal Bus, and (ii) External Bus.

- **Characteristics of microprocessor** :
  (i) Instruction Set
  (ii) Bandwidth
  (iii) Clock Speed
- **Classification of Microprocessor :**



Know the Terms

- **Arithmetic Logic Unit (ALU) :** It performs all the arithmetic and logical operations of the computer.
- **Control Unit :** It controls all the functions of the computer like input, output, storage and processes.
- **Registers :** It is a small amount of very fast memory that is built into the CPU. It is very expensive.
- **Internal Bus :** It is a bus which connects all the internal components of computer of CPU and main memory.
- **External Bus :** It connects the different external devices, peripherals, expansion slots, input/output ports and drive connections to the rest of the computer. External bus is also known as expansion bus.
- **Instruction Set :** The set of instructions that a microprocessor can execute.
- **Band width :** The number of bits processed in a single instruction.
- **Clock speed :** The clock speed determines how many instructions per second a processor can process. It is given in megahertz or gigahertz.
- **CISC :** It is acronym for Complex Instruction Set Computing. The CISC architecture contains a large set of computer instructions that range from very simple to very complex and specialised.
- **RISC :** It is acronym for Reduced Instruction Set Computing. It has relatively limited number of instructions. It is designed to perform relatively small number of operations so that it can operate at higher speed.
- **EPIC :** It is acronym for Explicitly Parallel Instruction Computing. It refers to architecture in which features are provided to facilitate compiler enhancements of Instruction Level Parallelism in all programs while keeping hardware complexity relatively low.

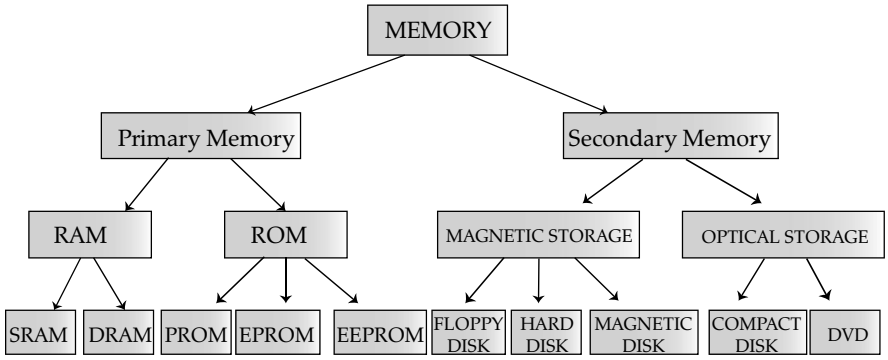
TOPIC-4
Memory Units & Concepts

Revision Notes

- **Order of Magnitude of Data :**

Metric system value (used to describe data transfer rate) (in Bytes)	**Notation**	**Actual value (used to describe data storage) in Bytes**
10^3	Kilo bytes (KB)	1024
10^6	Mega byte (MB)	$(1024)^2$
10^9	Giga byte (GB)	$(1024)^3$
101^2	Tera byte (TB)	$(1024)^4$
101^5	Peta byte (PB)	$(1024)^5$

- **Classification of Computer Memory:**



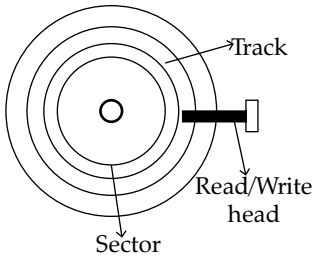
- A port is a connection point of interface between a computer and internal or external devices. Some common types of ports are - Serial Port, USB Port, Bluetooth, PS2 Port, Infrared Port, Firewire Port.
- **Primary Memory :** It is the internal storage which is used by our computer system to store data and instructions while processing. It is temporary memory and is volatile in nature. It is directly accessed by CPU. It is of two types viz. RAM and ROM.
  1. **RAM :** It is acronym for Random Access Memory . It is a type of Primary memory and is also known as read and write memory as it can be used for both the purposes. It is further subdivided into: .
     - **(i) SRAM :** It is acronym for Static Random Access Memory It is a type of memory that requires a constant power supply in order to hold the data and information.
     - **(ii) DRAM :** It is acronym for Dynamic Random Access Memory. The term dynamic indicates that the memory must be constantly needed to be refreshed otherwise it loses its contents. It is the most common kind of RAM used in personal computers.
  2. **ROM :** It is acronym for Read Only Memory. It provides non-volatile storage of data. We can access any memory location of ROM by supplying its address. It is further subdivided into following :
     - **(i) PROM :** It is acronym for Programmable Read Only Memory. It is a computer memory chip which can be programmed once it is created. But once it is programmed, then it cannot be changed or it is not re-programmable.
     - **(ii) EPROM :** It is acronym for Erasable Programmable Read Only Memory. It is a computer memory chip on which the written information can be changed as per the requirement. Its contents can be erased by using Ultraviolet light rays of a specific frequency for a specific period of time.
     - **(iii) EEPROM :** It is acronym for Electrically Erasable Programmable Read Only Memory. It is a PROM which can be erased and reprogrammed using an electric charge.
- **Secondary Memory :** It is also known as secondary storage Unlike primary memory, it is used by our computer system to store a large amount of data. It is non-volatile in nature and is not directly accessed by the CPU.
- **Cache Memory :** It is a smaller, faster memory used by a CPU to reduce average time to access data from main memory.
- **Magnetic Storage :** It is a magnetic medium which can be used to store large volume of data. It uses magnetic bead to read and write data. It gets the data to and from a magnetisable medium like plastic tape coated with fine particles of metal. It can store large volume of data but this data is slowly accessed.
- **Optical Storage :** Optical storage medium holds data in digital form. It uses a laser to read and write data It ranges up to 8GB.
- **Compact Disc (CD) :** It is an optical media that is used for electronically recording, storing and playing back audio, video, text and other information in digital form.
- **Digital Video Disc (DVD) :** It is also known as Super Density Disc(SDD) or Digital Versatile Disc(DVD). In appearance, they are somewhat similar to CDs but they have comparatively more storage capacities than that of CD.
- **Pen drive** is a storage device that includes flash memory with an integrated Universal Serial Bus (USB) interface. These are typically removable and rewritable and physically much smaller then an optical disc.
- **Blu-ray Disc (BD)** is a digital optical disc data storage format designed to supersede the DVD format. It is capable of storing high definition video resolution (1080p). It is of same size as DVDs and CDs.
- **Serial Port :** It is also known as COM Ports (communication ports).These are used to connect devices such as mouse, modem, etc. In this type of ports, only 1 bit information can travel at a time.
- **Parallel Port :** It is a type of port in which data are sent and received simultaneously over several parallel channels. It can transfer upto 8 bits of data at a time. It is used lo connect peripheral devices such as printer.
- **USB Port :** It is acronym for Universal Serial Bus Port. It comes under serial port and is used for short distance digital data communication. It allows data transfer between devices with little electric power.

- **Bluetooth :** It is a wireless technology standard for exchanging data over short distances (using short-wavelength) from fixed and mobile devices.
- **PS/2 Port :** It is a type of port developed by IBM for connecting a mouse or a keyboard to the computer
- **Infrared Port :** It is also known as IR port. It is a meta port which words and receives, infrared signals from other infrared enabled devices.
- **Firewire Port :** It is a serial bus interface standard for high speed communication and real time data transfer.

TOPIC-5
HARD DISK

Revision Notes

- Hard disk memories store information on one or more circular platters or disks which are continually spinning.
- The rotating disks are coated with a magnetic material.
- These have space between them.
- Information is recorded as tiny magnetic spots.
- Heads to record or read data are mounted on access arms.
- Information is recorded in bands.
- Concentric circles on the magnetised surface of the magnetic disks are known as tracks.
- Tracks are divided into pie-shaped sections called sectors.



TOPIC-6
INPUT – OUTPUT (I/O) Devices

Revision Notes

- **Input devices**
- Input unit is used to accept data and instructions from the user.
- Keyboard is used to type in letters, digits and commands.
- Mouse is a pointing input device.
- Microphone is used to send input to computer.
- A scanner creates an electronic form of the printed image.
- A touch screen can detect where on the screen surface you are touching.
- A barcode reader is used to read barcodes.
- **Output devices**
- Output unit is responsible for producing the output in user readable form.
- Monitor displays information in a way similar to that shown on a television screen.
- The picture on a monitor in made up of thousands of tiny coloured dots called pixels.
- CRT contains an electron gun at the back of the glass tube.
- Liquid Crystal is a material used to create each pixel on the screen in LCD.
- TFT (Thin Film Transistor) is the device within each pixel that sets the charge.
- PDP (Plasma Display Panels) are flat panel display.
- Plasma technology utilises small cells containing electrically charged ionised gases.
- OLED Monitors are created from pushing an electronic current through organic materials, causing these materials to glow.

- In impact printers, there is mechanical contact between the print head and paper.
- Speakers receive the sound in form of electric current from the sound card.
- The Plotter is a graphic output device used to create drawings on paper.

TOPIC-7
BATTERY

Revision Notes

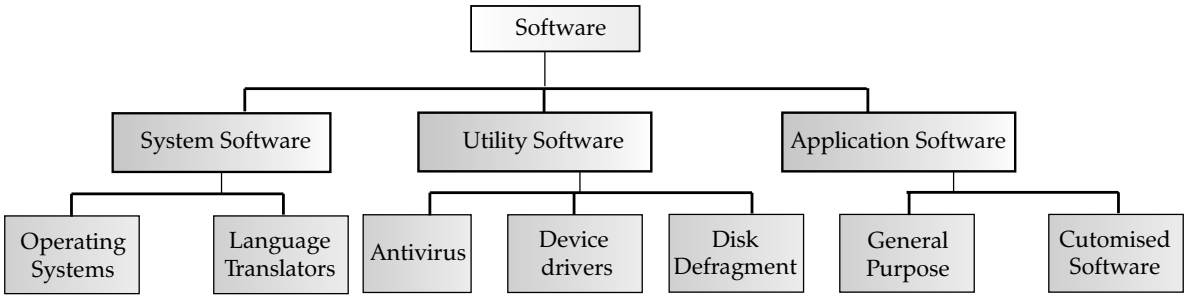
- Battery supplies power to a device so that it can do work without power cord.
- Laptops are powered by battery and can work for several hours.
- Laptops and cell phones use rechargeable batteries.
- Three types of computer batteries:
  (a) Backup or CMOS battery (power the CMOS chip)
  (b) Bridge battery is used in portable computers as a temporary backup for the main battery.
  (c) Main battery in portable computers is an alternative to a power cord.
- LI-ION (Lithium Ion) – used in cell phones. These are expensive but perform without the memory issue. These are fragile technology requiring protector circuit.
  - It is used where very high energy density is needed and cost is secondary.
  - It has a low self – discharge.
  - Energy density is twice that of the NICAD.
  - It has 500 – 800 charging cycles.
  - It is susceptible to damage from over charge and over discharge.
- Li-polymer (Lithium polymer) is a lower cost version of Li-ion.
- NICAD (Nickel Cadmium) is a rechargeable battery made of nickel and cadmium.
  - It can perform at low temperatures.
  - Should be charged only on complete discharge. Otherwise its maximum energy capacity would lower upon each charge/discharge cycle. This is known a memory loss.
  - 500 charging cycles.
  - Tolerant of overcharging.
- NIMH (Nickel Metal Hydride) is a rechargeable battery used primarily in portable computers.
  - Its capacity is 30% more than the NICAD.
  - These are less prone to memory loss.
  - Environment friendly.
  - Only 500 charge / discharge cycles.
  - Generate more heat during charge
  - Have high self discharge
- A computer battery is directly soldered to the motherboard.
- Functions of a computer battery is to power an integrated chip called an RTC or real time clock which runs, whether the system is on or off.
- Life span of a computer battery varies from 2 to 10 years depending on ambient temperature, duration of time the computer is powered off and the type of motherboard the battery is attached to.
- CMOS (Complementary Metal Oxide Semiconductor) battery recharges itself whenever the computer is turned on.

❑❑❑

CHAPTER-2
SOFTWARE CONCEPTS

Revision Notes

- Software represents the set of programs that govern the operations of a computer system and make the hardware run smoothly.
- **Classification of Software:**



(i) System Software : These software are mandatory for all computer systems to work. For example: Operating Systems like Windows and language processors like assembler, etc.

(ii) Application Software : These software are made to perform a specific task. For example: WordPad, MS-Word, etc.

(iii) Utility Software : These software are used to perform some additional functions which makes our computer more safe, secure and smooth working. These software are considered to be a part of system software. For example, antivirus, disk defragmenter, etc.

SYSTEM SOFTWARE

(i) Operating System : An operating system is a system program that acts as an interface between user and the machine. It allocates and manages the resources of the computer.

(ii) Language Translators are programs that are used to convert a program written in HLL or assembly language to machine language.

(a) Compiler: A complier is a system program which translates a program written in a high level language into its equivalent program in machine language.

Source program → Compiler → Machine Code

(b) Interpreter : It converts a HLL program into equivalent machine language program one line at a time and executing it, if no error is encountered.

Source program → Interpreter → Result

(c) Assembler: It can be defined as a system program than converts an assembly language program into the machine language of the host computer.

Assembler lang. Program → Assembler → Machine lang. Program

➢ **UTILITY SOFTWARE**

(i) Antivirus : An antivirus is utility- software which detects and removes computers viruses. The anti-virus keeps a watch on the functioning of the computer system.

(ii) Disk Defragmenter : A disk defragmenter scans the hard disk for fragmented files and brings all the fragments together.

(iii) Backup Utility : This utility is used to create the copy of the complete or partial data stored in a disk or CD on any other disk. In case the hard disk crashes or some other system failure occurs, the files can be restored using backup software.

(iv) Compression Utility : This utility is used to compress large files. Compression is useful because it helps reduce resources usage and the file transmission on the network becomes easier.

(v) Device Drivers - A device driver is a program that controls a particular type of device that is attached externally to a computer.

(vi) Disk Cleaner : This utility scans for files that have not been accessed/used since long. Such files might be occupying huge amount of memory space. In that case, the Disk Cleaner utility prompts the user to delete such files so as to create more space on the disk.

(vii) File Management Tools: This utility helps the user in storing, indexing, searching and sorting files and folders on the system. The most commonly used tool is the Windows Explorer and Google Desktop.

APPLICATION SOFTWARE

(i) General Purpose Application Software : Some of the application software is made for the common users for day to day application and uses. The users may use them in the manner they want.

- **Word Processor :** Word processor is a general purpose application software used to create documents. It allows us to create, edit and formal documents. Popular examples of Word processing software are Writer (Open Office) and Microsoft Word.
- **Presentation Tools :** Presentation. tools is a general purpose application software that lets us create pre-

sentations on any topic. We cannot only create a presentation and add slides but also can use different types of background, fonts', animations, audio, video, etc. We can add clipart and other graphics into our document. Even audio, video files can be added to the presentations. Popular examples of Presentation tools software are Impress (Open Office) and Microsoft Power Point.

- **Spreadsheet Packages :** Spreadsheet is a general purpose application software that lets us create and store data in tabular form. Both text and numerical values can be entered in a spreadsheet. All common mathematical and statistical formulae can be used on the numeric data. Popular examples of Spreadsheet software are Calc (Open Office) and Microsoft Excel.
- **Database Management System :** Database Management System is general purpose application software that lets us create computer programs that control the creation, maintenance, and the use of database for an organisation and its end users. We can not only store data but can also manage data in a database. Popular examples of Database Management System are Base (Open Office) and Microsoft Access.

(ii) Special - Purpose (customised software)

- **Customised Software :** Customised Software is one which is made as per the user's requirement. Such type of software is customer specific. It is made keeping in mind the individual needs of the user and so are also referred to as **Domain Specific Tools**. Such software cannot be installed and used by any other user/customer since the requirements may differ.
  - **Inventory Management System and Purchasing System :** Inventory Management System is generally used in departmental stores or other organisations to keep the record of the stock of all the physical resources. For Example, in a Computer store, it keeps record of the number of computers, printers, printing sheet, printer cartridge available. It also helps to place purchase orders, bills, invoices etc. Various reports as position of stock, sales made in a particular period, profit earned etc. can be generated.
  - **School Management System :** School Management System (sometimes called a School Information System or SIS) is a system that manages all the school's data in a single., integrated application. Having all of the information in a single system allows schools to more easily connect data together. For example, when viewing a student's record, the user can follow a link to the student's class, and from there a link to the student's teacher, and from there a link to the teacher's other classes, and so on,
  - **Payroll System :** Payroll Management System software is used by ail modern organisations to keep track of employees of the organisation who receives wages or salary. All different payment amounts are calculated by the payroll software and the record is maintained. The software keeps track of personal records of employees viz. name, address, dale of birth, qualification, date of joining etc. It also keeps track of professional record viz. allowances, perks, income tax, insurance etc. Different reports, pay slips etc can be generated through this software.
  - **Financial Accounting :** Financial Accounting System is used to prepare accounting information, maintain different accounts ledger, and account books. It also helps an organisation to make budget.
  - **Hotel Management:** Hotel Management software refers to management techniques used in the hotel sector These can include hotel administration, accounts, billing, marketing, housekeeping, front office or front desk, food and beverage management, catering and maintenance. Even advance bookings can be made through this software. Customers can have a look at the hotel and the rooms before making bookings. At any point of time the room availability, tariff for each type of room and even booking status can be checked.
  - **Reservation System :** Reservation System is software used to book (reserve) air flights, railway seats, movie tickets, tables in a restaurant, etc. In the case of a booking system, the inputs are booking requests. The processing involves checking if bookings are possible, and if so making the bookings. The outputs are booking confirmations/rejections.
  - **Weather Forecasting system :** This software makes it possible to forecast the weather for days and even months in advance. The detailed weather reports can also be generated.
- **Open Source Software** refers to those categories of software whose license does not impose much restrictions and conditions
- The software developers who support the open source concept feels that an application can be more useful and error free if the user can make changes in the program code as per their needs.

Know the Terms

- **Backup software** can make copies of all information stored on a disk and restore either the entire disk (e.g. in an event of disk failure) or selected files (e.g. in an event of accidental deletion).
- **Data compression** utilities output a shorter Stream or a smaller file when provided with a stream or file.

- **Data synchronisation** utilities establish consistency among data from a source to a target data storage and vice versa,
- **File synchronization** utilities maintain consistency between two sources. They maybe used to create redundancy or backup copies but are also used to help users carry their digital music photos and video in their mobile devices.
- **Disk cleaners** can find files that are unnecessary to and take up considerable amounts of space. Disk cleaner helps the user to decide what to delete when their hard disk is full.
- **Disk compression** utilities can transparently compress/ uncompress the contents of a disk, increasing the capacity of the disk.
- **Disk defragmenters** can detect files whose contents are scattered across several locations on the hard disk, and move the fragments to one location to increase efficiency.
- **File managers** provide a convenient method of performing routine data management tasks, such as deleting, renaming, cataloging, un-cataloging, moving, copying, merging, generating and modifying data sets.
- **Package managers** are used to configure, install or keep up to date other software on a computer.
- **Open Source Software :** Open source software is that computer software whose source code is made available with a license in which the copyright holder provides the rights to read, change and distribute the software for any purpose.
- **Freeware :** It generally refers to the software which are available for use at no cost or for an optional fee to avail all the functions of the software.
- **Shareware :** It is that software which is made available with a right to redistribute copies.
- **Proprietary Software :** It is a software which is owned by an individual or a company.

❑❑❑

CHAPTER-3
MEMORY UNITS AND NUMBER SYSTEM

TOPIC-1
Number Systems and Information Representation

Revision Notes

- A set of values used to represent different quantities is known as number system.
- Relation between various number systems is given below:

Hexadecimal	Octal	Decimal	Binary
0	0	0	0000
1	1	1	0001
2	2	2	0010
3	3	3	0011
4	4	4	0100
5	5	5	0101
6	6	6	0110
7	7	7	0111
8	10	8	1000
9	11	9	1001
A	12	10	1010
B	13	11	1011
C	14	12	1100
D	15	13	1101
E	16	14	1110

F	17	15	1111

- Remember the following table to perform binary addition.

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 10
1 + 1 + 1 = 11

- Binary numbers can be added column by column just like decimal numbers. e.g.

$$\begin{array}{r} 111_2 \\ +\ 101_2 \\ \hline 1101_2 \end{array} \qquad \begin{array}{r} 7_{10} \\ +\ 3_{10} \\ \hline 13_{10} \end{array}$$

Know the Terms

- Decimal (base 10) : It is the way most human beings represent numbers. Decimal is sometimes abbreviated as dec.
- Decimal counting goes : 0,1,2, 3, 4, 5, 6, 7,8,9,10,11,12,13,14,15,16,17,18 and so on.
- Binary (base 2) : It Is the natural way most digital circuits represent and manipulate numbers.
- Binary counting goes :0,1,10,11,100,101,110,111,1000,1001,1010,1011,1100,1101,1110,1111,10000,10001 and so on.
- Octal (base 8): It was previously a popular choice for representing a digital circuit number in a form that is more compact than binary. Octal is sometimes abbreviated as oct.
- Octal counting goes: 0,1,2,3,4,5,6,7,10,11,12,13,14, 15, 16,17,20,21 and so on.
- Hexadecimal (base 16): It is currently the most popular choice for representing digital circuit numbers in a form that is more compact than binary .
- Hexadecimal counting goes : 0,1,2,3,4,5,6,7,8,9, A, B, C, D, E, F, 10,11,12 and so on.

TOPIC-2

Representing Integers and Characters

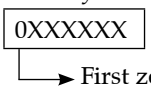
Revision Notes

- Integers may be represented in various ways in computers. These are
  (a) Sign and magnitude representation
  (b) One's complement, Two complement
- As we know that all digital computers store data consisting of digits, alphabets, symbols, etc. But this data is not stored in the form we entered but it is stored in the form of binary code. Each alphabet or digit or symbol have their unique binary code.
- Some of the main computer codes are:
  (i) ASCII Code : It is acronym for the American Standard Code for Information Interchange. It is used in most microcomputers and minicomputers and in many mainframes. It is a 7-bit code so it has 27 — 128 possible code groups.
  (ii) ISCII Code : It is acronym for Indian Standard Code for Information Interchange. It is a 8-bit code so it has 28-= 256 possible code groups. It retains all ASCII characters and offers coding for Indian characters also.
  (iii) UNICODE : It is the new universal coding standard being adopted by all newer platforms. Unicode provides a unique number for every character, no matter what the platform or program or the language is.
- **"UNICODE"**
  - Encoding scheme is a predefined way of converting information character by character in a machine intelligible code.
  - Code space refers to all the codes that an encoding scheme users e.g. ASCII encoding scheme has a code space from 0 to 127.
  - Code point represents a single character from the character set.
  - Code Unit refers to unit of storage used to represent one encoded code point e.g UTF 8 encoding scheme user 8 bits units but it may also use more bits units to represent some characters.

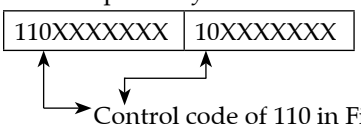
- **Unicode Encoding Schemes**
  Unicode defines multiple encoding systems to represent characters e.g. UTF 8, UTF 16 and UTF16 and UTF32
  (i) UTF 8 (Unicode Transformation Format) is a variable width encoding. Its code unit is 8 bits or octet. It can use 1 to maximum 6 octets to represent code points.
  Unicode code points are written as
  U + <code point number>
  e.g. U + 0041 -~ `A'

(decimal)	No. of octet used
U-0 – U-127	1 Octet (8 bits)
U-128-U-2047	2 Octets (16 bits)
U-2048-U-65535	3 Octets (32 bits)
U-65536-U-2097151	4 Octets (64 bits)

- In every UTF-8 (8 bit) octet representation left most bit is used as control bit and does not store actual data.



- In every UTF-8 (16 bit) octet representation, control codes 110 and 10 are stored in left most bits of two octets respectively.



- In every UTF 8,3 octet (24 bits representation, control codes of 110, 10 and 10 are stored in left most bits of the three octets respectively.
  1110XXXX 10XXXXXX 10XXXXXX
  Control code of 110, 10 and 10 in first, second and third octet indicates 3 octet coding.
- In every UTF 8,4 octet (32 bits representation) control codes of 1110, and 10 are stored in left most bits of all the 4 octets respectively.
  1110XXX 10XXXXXX 10XXXXXX 10XXXXXX
  control codes of 11100 in first octet and 10 in rest of the three octets represent 4 octet coding.
  (ii) UTF 32 is a fixed length encoding scheme that uses exactly 4 bytes to represent all Unicode points.

❑❑❑

CHAPTER-4
OPERATING SYSTEM

Revision Notes

- Operating System is defined as a collection of programs that coordinates the operations of computer hardware and software. It acts as a bridge or the interface between man and machine.
- Operating system is a system software which is mandatory for all computer systems to operate.
- Main functions of operating system are :
  - Device management
  - Processor management
  - Memory management
  - File management
- **Functions of an Operating System**
  - **Device management :** Operating system ensures the smooth functioning of all the peripheral devices.
  - **Processor management :** Operating system enables the activities of planning, performing and monitoring the performance of any process.
  - **Memory management :** It is the act of managing memory i.e., provide required memory for the process to happen and then save the desired result back in the memory.
  - **File management (Information management) :** Every user of a computer system wants to manipulate

some kind of information. This function of operating system allows the user to create a file, write into it, open it for reading, close it after reading or writing is over and delete it, if it is not required any more.

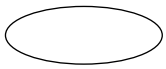
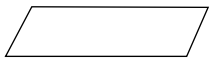
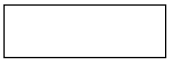
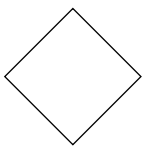
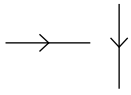
- **Types of Operating Systems**
  - **Real Time Operating System (RTOS) :** It is a multi-taking operating system that aims to execute real time applications. It is designed to respond to an event within a predetermined time. These are used to control machinery, scientific instruments and industrial systems.
    For example : Windows CE, Linux, TRON, etc.
  - **Multi user operating system :** It is the type of operating system that allows many users to take the advantage of computer's resources simultaneously.
    For example : UNIX, VMS, etc.
  - **Single user operating system :** As it is clear from the name, single user operating system is designed for one user to effectively use a computer at a time.
    For example : Windows 2007, Windows 10, etc.
  - **Single tasking operating system :** It is an operating system which can run only one program at a time. Palm operating system is the example of single tasking operating system.
  - **Multi-tasking operating system :** This operating system allows the execution of multiple tasks at a time.
    For example : Windows 2007/2008/2009 /XP, etc.
  - **Time sharing operating system :** It allows the users to share the computer resources simultaneously.
    For example : A mainframe computer that has many users logged on to it.
  - **Distributed operating system :** Distributed operating system uses multiple central processors to serve multiple real time applications.
  - **Command Line Interface :** It is a type of operating system in which user has to give instructions in the form of commands i.e. in the form of successive lines of text.
    For example : DOS (Disk Operating System).
  - **Graphical user interface :** It is a type of operating system in which user interacts with the machine graphically. In this, commands are given by selecting or clicking on the options.
    For example : All the versions of Windows, Mac OS, Android, etc..

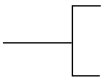
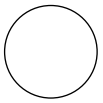
❑❑❑

CHAPTER-5
BASIC CONCEPTS OF FLOWCHART

Revision Notes

- A Flowchart is a pictorial representation of step by step solution of a Problem.
- Symbols used in a flowchart are as follows:

Symbol	Purpose
	Start /Stop
	Input / Output
	Processing
	Decision box
	Flow lines

	Annotation
	Connectors

❑❑❑

CHAPTER-6
BOOLEAN ALGEBRA

Revision Notes

- Boolean algebra developed by George Boole in 1854 deals with binary values and logical operations and hence is also known as Binary Algebra or Logical Algebra. It helps us in analysing and simplifying the digital circuits It is also called Switching Algebra.
- A **Boolean function** is defined by a Boolean expression consisting of binary variables, constants 0 or 1 and logical operators. For example: A Boolean function F (A,B,C) can be defined as
  F (A,B,C) = A.B + B.C + $\overline{A}$. $\overline{C}$
  Where LHS represents output Y of the logical expression. Hence, we can say
  Y = A.B + B.C + $\overline{A}$. $\overline{C}$
- A **Boolean expression** is a logical expression that produces a Boolean value when evaluated.
- Boolean Algebra uses three basic logical operators :-
  - **NOT operator –** It operates on single input and gives complement of the input as output. It is represented by an over bar on the variable to be operated. Thus if A = 0, then A' =1 and vice versa. Most programming languages use a '!' (exclamation sign) as NOT operator,
  - **OR operator-** It is a binary operator equivalent to a logical (+) addition. Thus **A + B C** implies **A OR B OR C**. The operator gives a true (i.e. 1) if any of the operand is true or 1. In programming '||' (double pipe) is used as a symbol for OR operator.
  - **AND operator-** It is a binary operator equivalent to logical (.) multiplication. Thus **A. B. C** implies **A AND B**. It gives a true (i.e. 1) only if both the operands are true or 1. In programming '&&' (double ampersand) is used as a symbol for operator AND.
- A logical variable can take only two values i.e. either a binary 1 or a binary 0. Although, there can be infinite number of variables in a Boolean expression. While evaluating boolean expression the order of evaluation of logical operators is NOT AND then OR Parenthesis is evaluated first.
- **Truth Table-** It is tabulated form of all the possible input combinations and their respective outputs. The number of possible input combinations for a Boolean expression is 2^n, where n is the number of input variables. Hence for a Boolean function F(A,B), number of possible combinations will be 2^2 i.e. 4 and for a Boolean function F(A,B,C,D) it will be 2^4 i.e. 16.
  For the function F(A,B,C) = A.B + B.C + $\overline{A}$. $\overline{C}$. there will be 8 possible combinations and hence 8 rows in the truth table as given below.

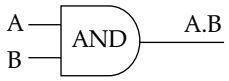
A	B	C	A.B	B.C	$\overline{A}$. $\overline{C}$	Y
0	0	0	0	0	1	1
0	0	1	0	0	0	0
0	1	0	0	0	1	1
0	1	1	0	1	0	1
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	1	0	0	1
1	1	1	1	1	0	1

- Laws used in Boolean Algebra are summed up in the table given below

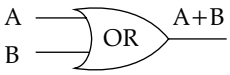
LAW	AND	OR
Commutative Law	A.B = B.A	A + B = B + A
Associative Law	A.(B.C) = (A.B).C	A + (B + C) = (A + B) + C
Distributive Law	A.(B + C) = A.B + A.C	A + (B.C) = (A + B).(A + C)
Identity	A.1 = A	A + 0 = A
Idempotent	A.A = A	A + A = A
Complement	$A.\overline{A} = 1$	$A + \overline{A} = 1$
Annulment Law	A.0 = 0	A + 1 = 1
DeMorgan's Law	$\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A} . \overline{B}$
Absorption Law	A.(A+B) = A	A + (A.B) = A
Double Negation or Inversion Law	$\overline{\overline{A}} = A$	

- Logic Gate is simply an electronic circuit which takes one or more electronic signals as input to give an output.
  - **NOT Gate** is an inverter circuit that operates on single input. It gives complement of the input as the output. It is represented by the following symbol.

  - **AND Gate** gives a high output if both the inputs are high. Algebraically it is represented by a (.) dot. Its electronic symbol is



  - **OR Gate** gives a high input if any one of the input is high. Its algebraic symbol is (+) Plus. and electronically it is represented as



  - **NAND Gate** is inverter of AND gate. It gives a High Input if any of the inputs is low. Its symbol is NAND Gate with a small circle at the output implying inversion.

  - **NOR Gate** is inverter of OR. Its output is low if any of the inputs is high.

- **Absorption Law -** According to this law: (a) X + XY = X
  (b) X (X + Y) = Y
- $X + \overline{X}Y = X + Y$
  - **XOR Gate** gives a high output if either of the inputs is high i.e. odd number of inputs is high. It is used to detect the mismatch of bits. Its algebraic symbol is.

$Q = A \oplus B$ or
$Q = \overline{A} . B + A . \overline{B}$

  - **XNOR Gate** gives a high output when both the inputs are same *i.e* both are either high or both are low.

$Y = A \odot B$ or
$Y = \overline{AB} + AB$

- A combined Truth table for all the gates is given below

INPUTS		OUTPUTS					
A	**B**	**AND**	**NAND**	**OR**	**NOR**	**XOR**	**XNOR**
0	0	0	1	0	1	0	1
0	1	0	1	1	0	1	0
1	0	0	1	1	0	1	0
1	1	1	0	1	0	0	1

NOT GATE	
A	$\overline{A}$
0	1
1	0

- **Tautology-** It is an expression or assertion that is always true.
- **Fallacy -** An expression that always yields a 0 or false is called Fallacy.
- **Basic Postulates of Boolean Logic**
  - **(i)** If $x \neq 0$ then $x = 1$ and if $x \neq 1$ then $x = 0$
  - **(ii)** Logical Addition (or relations)

0+0=0
0+1=1
1+0=1
1+1=1

  - **(iii)** Logical Multiplication (AND Relations)

0.0=0
0.1=0
1.0=0
1.1=1

  - **(iv)** Complement Rules
    $\overline{0} = 1$
    $\overline{1} = 0$
- Principle of Duality states that starting with a boolean relation, another boolean relation can be derived by
  1. Changing each OR sign (+) to an AND sign (.).
  2. Changing each AND sign (.) to an OR sign (+).
  3. Replacing each 0 by 1 and each 0 by 1.
- **Theorems Of Boolean Algebra**
- **Annulment Law –** A term AND´ed with a "0" equals 0 or OR´ed with a "1" will equal 1.
- **Identity Law –** A term OR´ed with a "0" or AND´ed with a "1" will always equal that term.
- **Idempotent Law –** An input that is AND´ed or OR´ed with itself is equal to that input.
- **Complement Law –** A term AND´ed with its complement equals "0" and a term OR´ed with its complement equals "1".
- **Commutative law –** It states that changing the sequence of the variables does not have any effect on the output of a logic circuit.
- **Associative law -** This law states that the order in which the logic operations are performed is irrelevant as their effect is the same.
- **Distributive law -** Distributive law states the following condition.

$$A.(B + C) = A.B + A.C$$

- **Absorption Law**

$$X+(X.Y)=X$$
$$X.(X+Y)=X$$

- Involution Law $(\mathbf{X}) = \mathbf{X}$ also called double - invasion law
- **Demorgan's Law**
  - **I.** $\overline{X+Y} = \overline{X}\,\overline{Y}$
  - **II.** $\overline{X.Y} = \overline{X} + \overline{Y}$
- **Universal gates –** NAND And NOR gates are known as universal gates as any possible circuit can be built using these two gates.

- The decision which result into either Yes (TRUE) or No (FALSE) is called a Binary Decision.
- Value true and false are called True values.
- Statements which can be determined to be true or false are called logical statements truth functions and the results TRUE or FALSE are called Truth values.

Know the Terms

- **AND – OR - Invert Logic :** When the output of an AND – OR circuit is complemented (inverted), it results in an AND – OR – Invert circuit.
- **AND – OR Logic :** AND - OR circuit can have any number of AND gates each with any number of inputs.
- **Exclusive – NOR Logic :** The exclusive – NOR can be implemented by simply inverting the output of an exclusive – OR (XNOR).
- **Exclusive – OR logic :** This circuit is considered a type of logic gate with its own unique symbol it is actually a combination of two AND gates, one OR gate, and two inverters (XOR).
- **NAND Gate :** The NAND gate is a universal gate because it can be used to produce the NOT, the AND, the OR, and the NOR functions.
- **NOR Gate :** The NOR gate is a universal gate because it can be used to produce the NOT, AND, OR and NAND functions.

❑❑❑

CHAPTER-7
CLOUD AND PARALLEL COMPUTING

TOPIC-1
Understanding Translation Process

Revision Notes

- Source code refers to the Original code written in a programming language by a programmer.
- Machine code refers to the code converted into computer understandable form that a computer can understand and can directly execute.
- The special software that translates a source code into machine understandable form is known as the translator.
- Translator software or simply the translators.
- Two types of translators are
  1. Compiler
  2. Interpreter
- Compilation process undergoes following steps
  1. Preprocessing
  2. Compilation
     - Analysis (Front End phase)
     - Synthesis(Back End phase)
  3. Assembly
  4. Linking
- Preprocessing phase removes extras (such as comments) From source code and add all that is to be added / expanded so that code is completely ready for the next phase.
- Analysis phase of compilation identifies all the tokens in the source code and creates a symbol table with it.
- Synthesis phase passes the source code and generates syntax tree out of it.
- The final product is an executable file (.exe file) which is the object code packaged with the Libraries.
- The part of the compiles which performs the linking of libraries is called linker.
- The Loader is a part of compiler that loads the computer executable module into memory for execution.
- A running program or the executing entity is called a process

TOPIC-2

Introduction to parallel and cloud computing

Revision Notes

- The simultaneous working of a multiple processors to solve a computational problem is parallel computing.
- Parallel computing may be achieved through.
  - a single computer with multiple processors / cores.
  - An arbitrary number of such computers connected by a network.
- When a program is divided into independent units and these independents units are run as separate processes it is called threading.
- Cloud computing refers to storing and accessing data and programs over the Internet instead of your computer's hard drive.
- Clouds are of two types
  (i) Public cloud **(ii)** Private cloud
- Public cloud refers to a common cloud service made available to multiple subscribers.
- Organizations or individuals that use the same cloud are called Cloud "tenants."
- A private cloud consists of computing resources exclusively owned by one business or organization.

UNIT-II
Computational Thinking And Programming

CHAPTER-8
PYTHON FUNDAMENTALS

Revision Notes

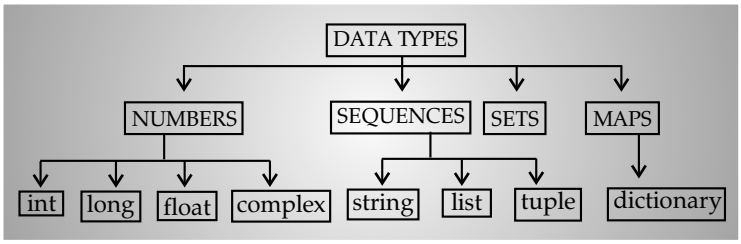
- Python programming language was developed by Guido Van Rossum in Feb 1991. It was named after famous BBC comedy show Monty Python's Flying Circus.
- It is an easy to learn yet powerful object oriented programming language. It is a very high level programming language.
- Variables are reserved memory locations to store values. A variable has a name, a type and a value.
- Interactive mode of working means we type the command – one command at a time, and the Python executes the given command there and then gives us output.
- Interactive mode is used for testing code.
- Script mode is useful for creating programs and then run the programs later and get complete output
- Python is an interpreted language.
- Python's interactive interpreter is also called Python shell.

❑❑❑

CHAPTER-9
DATA HANDLING

Revision Notes

- Data type is a set of values and the allowable operations on those values. Python has a great set of useful data types. Python's data types are built in the core of the language. They are easy to use and straightforward.



- **Numbers** can be either integers or floating point numbers.
- A **sequence** is an ordered collection of items, indexed by integers starting from 0. Sequences can be grouped into strings, tuples and lists.
  - **Strings** are lines of text that can contain any characters. They can be declared with single or double quotes.
  - **Lists** are used to group other data. They are similar to arrays.
  - A **tuple** consists of a number of values separated by commas.
- A **set** is an unordered collection with no duplicate items.
- A **dictionary** is an unordered set of key:value pairs where the keys are unique

Data type conversions

There are several built-in functions to perform conversions from one type to another.

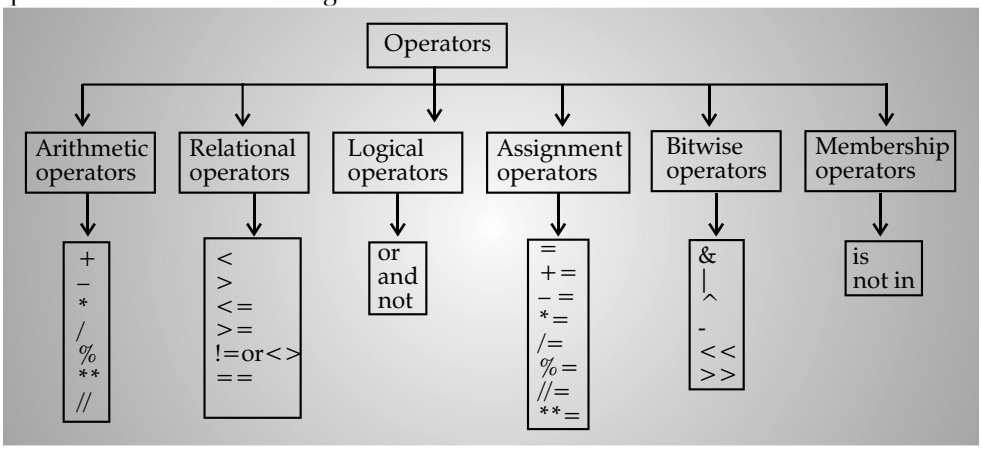
Function	Description
int(x [,base])	Converts x to an integer. base specifies the base if x is a string.
long(x [,base])	Converts x to a long integer. base specifies the base if x is a string.
float(x)	Converts x to a floating-point number.
complex(real [,imag])	Creates a complex number.
str(x)	Converts object x to a string representation.
repr(x)	Converts object x to an expression string.
eval(str)	Evaluates a string and returns an object.
tuple(s)	Converts s to a tuple.
list(s)	Converts s to a list.
set(s)	Converts s to a set.
dict(d)	Creates a dictionary. d must be a sequence of (key,value) tuples.
frozenset(s)	Converts s to a frozen set.
chr(x)	Converts an integer to a character.
unichr(x)	Converts an integer to a Unicode character.
ord(x)	Converts a single character to its integer value.
hex(x)	Converts an integer to a hexadecimal string.
oct(x)	Converts an integer to an octal string.

- An **implicit** type conversion is a conversion performed by the compiler without programmer's intervention.
- An **explicit** type conversion (type promotion), is done by Python where it converts all the operands to the largest operand.

  Example: a,b=3,5

  c= a/b → c will always have floating point result.

- Operators are special symbols which perform some computation. Operators and operands form an expression.
- Python operators can be classified as given below.



- **Arithmetic Operators**
  - Arithmetic operators help us to perform various arithmetic calculations. The arithmetic operators are explained in the table below. Let's assume value of a=2 and b=3

Operator	Description	Example
+	Addition - Adds values on either side of the operator	a + b results in 5
-	Subtraction - Subtracts right hand operand from left hand operand	a –b results in -1
*	Multiplication - Multiplies values on either side of the operator	a * b results in 6
/	Division - Divides left hand operand by right hand operand	a/b results in 0
%	Modulus - Divides left hand operand by right hand operand and returns remainder	a %b results in 2
**	Exponent - Performs exponential (power) calculation on operators	a**b results in 8
//	Floor Division - The division of operands where the result is the quotient in which the digits after the decimal point are removed.	11//2 results in 5 11.0//2.0 results in 5.0

- Relational/Comparison Operators
  - These operators help us to make decisions based on certain conditions.

Operator	Description	Example
= =	Checks if the value of two operands are equal or not, if yes then condition becomes true.	(a == b) is not true.
!=	Checks if the values of two operands are equal or not, if values are not equal then condition becomes true.	(a != b) is true.
< >	Checks if the value of two operands are equal or not, if values are not equal then condition becomes true.	(a <> b) is true. This is similar to != operator.
>	Checks if the value of left operand is greater than the value of right operand, if yes then condition becomes true.	(a > b) is not true.
<	Checks if the value of left operand is less than the value of right operand, if yes then condition becomes true.	(a < b) is true.
>=	Checks if the value of left operand is greater than or equal to the value of right operand, if yes then condition becomes true.	(a >= b) is not true.
<=	Checks if the value of left operand is less than or equal to the value of right operand, if yes then condition becomes true.	(a <= b) is true.

- **Assignment Operators**
  - These operators assign the right side value to the left side variable.

Operator	Description	Example
=	Simple assignment operator, Assigns values from right side operands to left side operand	c = a + b will assign the value of a + b into c
+=	Add AND assignment operator, It adds right operand to the left operand and assign the result to left operand	c += a is equivalent to c = c + a
-=	Subtract AND assignment operator, It subtracts right operand from the left operand and assign the result to left operand	c -= a is equivalent to c = c – a
*=	Multiply AND assignment operator, It multiplies right operand with the left operand and assign the result to left operand	c *= a is equivalent to c = c * a
/=	Divide AND assignment operator, It divides left operand with the right operand and assign the result to left operand	c /= a is equivalent to c = c / a
%=	Modulus AND assignment operator, It takes modulus using two operands and assign the result to left operand	c %= a is equivalent to c = c % a
**=	Exponent AND assignment operator, Performs exponential (power) calculation on operators and assign value to the left operand	c **= a is equivalent to c = c ** a
//=	Floor Division AND assigns a value, Performs floor division on operators and assign value to the left operand	c //= a is equivalent to c = c // a

➢ **Bitwise operators**

- Bitwise operator works on bits and perform bit by bit operation. Assume if a = 65; and b = 12; Now in binary format they will be as follows:

```
a    = 0100 0001
b    = 0000 1100
a&b  = 0000 0000
a|b  = 0100 1101
a^b  = 0100 1101
~a   = 1011 1110
```

- The table below lists the bitwise operators in Python. Let's assume a=65 and b=12 for this example.

Operator	Description	Example
&	Binary AND Operator copies a bit to the result if it exists in both operands.	(a & b) results in 0 (0000 0000)
|	Binary OR Operator copies a bit if it exists in either operand.	(a | b) results in 77 which is 1001101
^	Binary XOR Operator copies the bit if it is set in one operand but not both.	(a ^ b) results in 77 which is 1001101
~	Binary Ones Complement Operator is unary and has the effect of 'flipping' bits.	(~a) results in -66 which is 10111110.
<<	Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.	a << 2 results in 260 which is 100000100
>>	Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand.	a >> 2 results in 16 which is 10000

➢ **Logical Operators**

- There are following logical operators supported by Python language.

Operator	Description	Example
AND	Logical AND operator. If both the operands are true then then condition becomes true.	a=5 b=10 (a and b) is true.
OR	Logical OR Operator. If any of the two operands are non zero then then condition becomes true.	(a or b) is true.
NOT	Logical NOT Operator. Use to reverses the logical state of its operand. If a condition is true then Logical NOT operator will make false.	not(a) is false.

➢ **Membership Operators**

- Python has membership operators, which test for membership in a sequence, such as strings, lists, or tuples. There are two membership operators explained below:

Operator	Description	Example
In	Evaluates to true if it finds a variable in the specified sequence and false otherwise.	Y = [] If x in y: return 1 if x is a member of sequence y.
not in	Evaluates to true if it does not finds a variable in the specified sequence and false otherwise.	Y = [] If x not in y: here not in results in a 1 if x is not a member of sequence y.

- **Identity Operators**
  - Identity operators compare the memory locations of two objects. There are two Identity operators explained below:

Operator	Description	Example
Is	Evaluates to true if the variables on either side of the operator point to the same object and false otherwise.	x is y, here **is** results in 1 if id(x) equals id(y).
is not	Evaluates to false if the variables on either side of the operator point to the same object and true otherwise.	x is not y, here **is not** results in 1 if id(x) is not equal to id(y).

- **Operator precedence**
  - The following table lists all operators from highest precedence to lowest.

Operator	Description
**	Exponentiation (raise to the power)
~ + -	Complement, unary plus and minus (method names for the last two are +@ and -@)
* / % //	Multiply, divide, modulo and floor division
+ -	Addition and subtraction
>> <<	Right and left bitwise shift
&	Bitwise 'AND'
^ |	Bitwise exclusive `OR' and regular `OR'
<= < > >=	Comparison operators
<> == !=	Equality operators
= %= /= //= -= += *= **=	Assignment operators
is , is not	Identity operators
in not in	Membership operators
not or and	Logical operators

❑❑❑

CHAPTER-10
CONDITIONAL AND LOOPING CONSTRUCTS

Revision Notes

- **Flow of Control:** It is the order of execution of various statements written in a program.
- The control and flow of the program can be divided into:
  - Sequence – When the statements are executed in a sequence one after the other.
  - Selection/Conditional/Decision – When the statements are executed depending on the validation of the condition.
  - Iteration/Looping – When a given set of statements are executed more than once due to presence of a loop.
- **Flowchart:** It is pictorial representation of the order in which the steps that need to be performed a given task.
- The symbols commonly used in a flowchart are:

Name	Symbol	Function
Start/End		Used to markup the starting and ending point
Arrows		Used for connection
Input/Output		Used for input and output information
Process		Used to represent single step
Decision		Used for branching or decision making

➢ Types of Selection statements:

- if statement –
  - ♦ simplest form of if statement
  - ♦ Syntax

```
if condition:
    set of statements
```

  - ♦ the set of statements will be executed only if the condition is true.
  - ♦ the set of statements can have any number of statements.
- if-else statement –
  - ♦ allows two different sets of statements to be executed upon satisfaction or non-satisfaction of the test condition
  - ♦ Syntax

```
if condition:
    set1 of statements
else:
    set2 of statements
```

  - ♦ the set1 of statements will be executed if the condition tested is true.
  - ♦ the set2 of statements will be executed if the condition tested is false.
  - ♦ the two sets of statements can have any number of statements
- if-elif-else –
  - ♦ allows simultaneous testing of more than one conditions and also allows different sets of statements to be executed upon satisfaction or non-satisfaction of the test condition.
  - ♦ Syntax

```
if condition1:
        set1 of statements
elif condition2:
        set2 of statements
elif condition3:
        set3 of statements
elif condition4:
        set4 of statements
:
else:
        setN of statements
```

  - ♦ the set1 of statements will be executed if test condition1 is true. In case, condition 1 is false, then program flow will move to elif and tests condition 2.
  - ♦ the set2 of statements will be executed if test condition2 is true. In case, condition 2 is false, then program flow will move to next elif and tests condition 3.
  - ♦ the set3 of statements will be executed if test condition3 is true. In case, condition 3 is false, then program flow will move to next elif and tests condition 4.
  - ♦ the set4 of statements will be executed if test condition4 is true. In case, condition 4 is false, then program flow will move to next elif and tests the next condition.

- ♦ When all the conditions are false, the program flow will move to the else and its corresponding set of statements will be executed.

• Nested if-else statement –
- ♦ allows a conditional statement inside another conditional statement such as *if* or *else* or *elif* statement inside another *if* or *elif* or *else* statement.
- ♦ Syntax

```
if condition1:
        set1 of statements
        if condition11:
                set11 of statements
        elif condition12:
                set12 of statements
        else:
                set13 of statements
elif condition2:
        set2 of statements
else:
                set3 of statements
```

➢ Types of Looping statements:

• for loop / definite loop –
- ♦ use to repeat the set of instructions over a defined range of values.
- ♦ the condition for execution of loop is checked after every repetition.
- ♦ else statements is executed when the control exits the for loop.
- ♦ Syntax

```
for <Control_variable> in <range of values>:
        set1 of statements
else:
                set2 of statements
```

- ♦ range( ) function - used to create a list of values to be taken by loop with a start, stop and increment (optional) value. Syntax range (start, stop[, increment])

• while loop / indefinite loop –
- ♦ use to repeat the set of instructions till the test condition is true.
- ♦ the condition for execution of loop is checked after every repetition.
- ♦ else statements is executed when the condition becomes false.
- ♦ Syntax

```
while <condition>:
                set1 of statements
else:
                set2 of statements
```

• Infinite loop – It is a loop in which the test condition can never become false.

• Nested loop – When one loop is placed inside another loop. First, the condition of outer loop is tested, if it is true then condition of inner loop will be tested.

• Syntax of nested while loop

```
while condition :
        while condition :
                set1 of statements
        set2 of statements
set3 of statements
```

• Syntax of nested for loop

```
for <Control_variable1> in < range of values>:
        for <Control_variable2> in < range of values>:
                set1 of statements
        set2 of statements
set3 of statements
```

➢ Jump statements: These statements are used to take the control of the program out of the loop even if the test condition is still true.

➢ Types of jump statements:

- **Break statement –** It is used to stop execution of the loop immediately and transfer the flow of control to the statement immediately after the loop.

  Syntax

  ```
  break
  ```

- **Continue statement –** It is used to stop execution of the loop immediately and transfer the flow of control to the beginning of the loop again.

  Syntax

  ```
  continue
  ```

- **Pass statement –** It is used to define a statement body with no statements and that does nothing.

  Syntax

  ```
  pass
  ```

❑❑❑

CHAPTER-11
IDEA OF DEBUGGING

Revision Notes

- Error is an unusual condition that can occur in a computer program.
- **Types of errors**
  - **Syntax error –** An error is the syntax or grammatical rules in the sequence of characters of the programming language. For example, writing capital keyword of commands, forget to apply brackets, etc.
  - **Runtime error –** An error which occurs after interpretation of program by the interpreter and during execution of the program. For example, division by zero, accessing a file that does not exist, etc.
  - **Exceptions –** An error that occurs during execution of a program.
- **Exception handling** – It is the way in which Python handles exceptions so as to prevent termination of script and execution of some specific part of the code.
- **Traceback –** It is the lengthy error message which is displayed when an exception occurs.
- **Try block –** A block is created in some part of code that might result in an exception.
- **Catch / Except block –** A block with set of statements that have to be followed when some exceptions occur in a program.
- **Throw / Raise block –** A block that contains an action in response to an exception.
- **Uncaught / Unhandled exception –** It is an exception that results in abnormal termination of a program as a result of not catching a throw exception.
- **Exception handling situations in Python –**
  - raise statement – to raise exception in a program.
  - **Syntax**

    ```
    raise [exception name [, message] [,traceback] ]
    ```

  - Some common exceptions that occur in Python are:
    - **(i) NameError –** it occurs when a variable is used that does not exist.
    - **(ii) TypeError –** it is raised when an invalid operation for a specified data type is attempted.
    - **(iii) ValueError –** it is raised when an operation is performed on an incorrect type of value.
    - **(iv) ZeroDivisionError –** it is raised when division by zero occurs.
    - **(v) AttributeError –** it is raised when an attribute is not found by an object or a non-existing attribute is accessed.
    - **(vi) KeyError –** it occurs when dictionary is accessed by using a non-existing key.
    - **(vii) IndexError –** it is raised when an index is not found in a sequence.
    - **(viii) IOError –** it is related to incorrect input/output like opening a file which does not exist, deleting a file which is in use, etc.
    - **(ix) IndentationError –** it occurs when each statement written in a block does not have same indentation level.

- **try and except block –** to catch and handle exceptions

Syntax

- **First method**

```
try:
        set of instructions to run the code
except:
        set of instructions to execute in case of exceptions
```

- **Second method**

```
try:
        set of instructions to run the code
except:
        set of instructions to execute in case of exceptions
else:
        set of instructions to execute in case there is no exception
```

- **Third method**

```
try:
        set of instructions to run the code
except Exception I:
        set of instructions to execute in case of exceptions
except Exception II:
        set of instructions to execute in case of exceptions
……..
else:
        set of instructions to execute in case there is no exception
```

- Sometimes, the 'finally' clause is added as clean-up or termination clause as it is executed in all circumstances.

- **Debugging –** it is the process of finding errors in a program.
- Debugger helps to see the state of any variable in an application.
- Python's interactive source code debugger is pdb.
- **Some common debugger functions are:**
  - run (statement [, globals [, locals]]) – to execute the statement under debugger control.
  - runeval (expression [, globals [, locals]]) – to evaluate the expression under debugger control.
  - runcall (function [, argument,]) – to call the function with the given arguments.
  - set_trace ( ) – to hard-code a break point at a given point in the program.
- **Some common debugger commands are:**
  - l (list) – shows the current position in the script.
  - n (next) – steps into next line of execution, excluding functions.
  - s (step) – steps into next line of execution, including functions.
  - r (return) – steps to the end of current function.
  - c (continue) – executes the code.
  - q (quit) – exits the debugger.
  - ? (help) – shows all debugger commands available.
- set_trace is used to create a breakpoint in the code from where navigation of code can be done.
- The control and flow of the program can be di

❑❑❑

CHAPTER-12
LISTS, TUPLES AND DICTIONARIES

Revision Notes

- **Sequence** is an object that contains multiple items of data.
- Items are stored in a sequence one after another.
- Sequence may have repeated items in a list.
- The number of elements is called **length** of the sequence.

- Various sequences available in Python are:
  - Lists
  - Strings
  - Dictionaries
  - Tuples
  - Sets

LISTS

- **List** is a collection of values or an ordered sequence of values.
- The items in a list can be of any type such as string, integer, float, object, etc.
- Elements of a list are enclosed in square brackets [ ], separated by commas.
- Values in the list can be modified because it is mutable.
- The values that make up a list are called its **elements**.
- **Syntax** for creating a list:

  <list_name> = [ ]
- The name given as list_name is called **initialization**.
- A list with blank or no values is called an **empty list**.
- **Creating a list from an existing sequence:**
  - **Creating a list from a sequence**

    <new_list_name> = list (sequence)
  - **Creating an empty list**

    list_name = list( )
  - List can also be created through user input
- List index can be a positive or negative integer value.
- An IndexError appears if the user tries and accesses elements that do not exist in the list.
- **Traversing a list** means accessing each element of a list.
  - Using 'in' operator

```
for i in list_name:
        print (i)
```
  - Using range( ) function

```
for i in range(len (list_name)):
        print (list_name[i])
```
- If we assign the elements of one list to another list, both shall refer to the same object.
- Changes made with one alias get reflected in the other alias.
- Aliasing should be avoided.
- In Python, while comparing two lists, each element is individually compared in lexicographical order.
- Two lists can be compared if they are of comparable type, otherwise Python flashes an error.
- **Operations on Lists:**
  - **Concatenation** – A process in which multiple lists can be combined together with the help of certain operators using '+'.

    Syntax:

    list3 = list1 + list2
  - **Replication** – A process in which a list gets replicated or repeated a specific number of times using '*'.

    Syntax:

    list1 * 3
    - We cannot multiply two lists
  - **Membership Testing** – An operation carried out to check whether a particular element is a member of that list or not.
    - Using 'in' operator – returns "True" if the element appears in the list, otherwise returns "False".
    - Using 'not in' operator – returns "True" if the element does not appear in the list, otherwise returns "False".

    Syntax:

    print (<element> in/not in <list1>)
  - **Indexing** – An index value is assigned for each item present in the sequence.
    - In Python, indexing starts from 0.
    - Negative indices identify positions from the end of the list.

- **Slicing** – It is an operation in which the user can slice a particular range from that sequence.
  - List slices are sub-part of a list extracted out.

    Syntax:

    list1 [start: stop: step]

➢ **Built-in Functions for lists:**

- **append( )** – Adds a single item to the end of the list and does not create a new list.

  Syntax:

  list1.append(item)
- **extend( )** – Adds one list at the end of the other list.

  Syntax:

  list1.extend(list2)
- **insert( )** – Inserts an element at a specified index.

  Syntax:

  list1.insert(index_number, value)
- **reverse( )** – Reverses the order of the elements in a list.

  Syntax:

  list.reverse( )
- **index( )** – Returns the index of first matched item from the list.

  Syntax:

  list.index(<item>)
- **update( )** – Changes an item or a range of items using '='

  Syntax:

  list[index] = <new value>
- **len( )** – Returns the length of the list.

  Syntax:

  len(list)
- **sort( )** – Sorts the items of the list.
  - For ascending order

    Syntax:

    list.sort( )
  - For descending order

    Syntax:

    list.sort(reverse = True)
- **clear( )** – Removes all items from the list.

  Syntax:

  list.clear( )
- **count( )** – Counts how many times an element has occurred in a list and returns it.

  Syntax:

  list.count(element)

➢ **Deletion operation** – For deleting an item from a list.

- If index is known
  - **pop( )** – Removes the element from specified index and returns the removed element.

    Syntax:

    list.pop(index)

    If no index value is provided, the last element in the list is removed.
  - **del statement** – Removes the specified element but does not return the removed element.

    Syntax:

    del list(index) - to delete single element

    OR del list(start index : stop index) - to delete a range of elements
- If element is known but its index is not known
  - **remove( )**

    Syntax:

    list.remove(element)

➢ **Searching the list**

- For a particular element / index – **index( )**
  Syntax:
  list.index(element)
- For the maximum value in the list – **max( )**
  Syntax:
  max(list)
- For the minimum value in the list – **min( )**
  Syntax:
  min(list)

TUPLES

- A **tuple** consists of a number of values separated by commas.
- Tuples are enclosed within parentheses ( ).
- The values that make up a tuple are called its **elements**.
- Elements in a tuple need not be of the same type.
- The index value of tuple starts with 0.
- Tuples are faster and more efficient than lists.
- If a tuple comprises of a single element, the element should be followed by a comma. Such a tuple is called a singleton tuple.
- Creating tuple with a single element:
  Syntax:
  tuple_name = ("January",)
  OR tuple_name = tuple( )
- A tuple can be created by accepting input by user input using ***while*** loop.
- Tuples can be nested. This means that tuples can be placed inside other tuples.
- The individual elements of a tuple can be accessed through their indices given in square brackets [ ].
- A tuple can be traversed using
  - **'in' operator with *for* loop.**
    Syntax:
    for i in tuple_name:
    print(i)
  - **range( ) function**
    Syntax:
    for i in range(len (tuple_name)):
    print(tuple_name [i])
- **Slicing** is used to retrieve a subset of values.
  Syntax:
  tuple_name[start: stop: step]
- Two tuples can be combine together using '+' operator.
- The elements of a tuple can be repeated using '*' operator.
- The users can check whether a particular element is a member of that tuple or not.
  - Using 'in' operator – returns "True" if the element appears in the tuple, otherwise returns "False".
  - Using 'not in' operator – returns "True" if the element does not appear in the tuple, otherwise returns "False".

  Syntax:
  <element> in/not in <tuple_name>
- **Tuple functions:**
  - **len( )** – Returns the length of a tuple.
    Syntax:
    len(tuple_name)
  - **count( )** – Counts the occurrence of an item in the tuple.
    Syntax:
    tuple_name.count(element)
  - **any( )** – Returns True if a tuple is having at least one item and returns False if the tuple is empty.
    Syntax:
    any(tuple_name)

- **max( )** – Returns the element with maximum ASCII value in the tuple.
  Syntax:
  max(tuple_name)
- **min( )** – Returns the element with minimum ASCII value in the tuple.
  Syntax:
  min(tuple_name)
- **sorted( )** – Sorts the elements of a tuple.
  Syntax:
  sorted(tuple_name)
- **index( )** – Finds the first index of a given item and returns the index.
  Syntax:
  tuple_name.index(value, start, end)

➢ Tuples can be compared using comparison operators like <, >, = =, !=, etc.
➢ In Python, comparison operators start by comparing the first element from each sequence. If they are equal, it goes on to the next element until it finds the elements that differ. The subsequent elements are not considered.
➢ A tuple is deleted using del statement.
Syntax:
del tuple_name

DICTIONARIES

Python **Dictionary** is an unordered collection of items where each item is a key-value pair.

➢ A dictionary can be created by placing items inside curly braces { } separated by a comma.
Syntax:
<dictionary_name>={'key1':'value1', 'key2':'value2',…, 'keyn':'valuen'}
➢ To access elements in a dictionary, square brackets [ ] alongwith the key are used.
➢ **Traversing a dictionary** means accessing each element of a dictionary.
Syntax:
```
for i in dictionary_name:
    print(i, ':', dictionary_name[i])
```
➢ **To add new elements** to an existing dictionary
Syntax:
dictionary_name ['key'] = 'value'
➢ **To modify existing key-value pair** in a dictionary
Syntax:
dictionary_name ['key'] = 'value'
➢ **To merge two dictionaries**
Syntax:
dictionary_name.update(dictionary2)
- When two dictionaries are merged, the values of the same key are overwritten.

➢ **To remove an item from the dictionary**
- Using **del command**
  Syntax:
  del dictionary_name[key]
- Using **pop( ) method**
  Syntax:
  dictionary_name.pop(key)

➢ The users can check whether a particular key is present in a dictionary or not.
  - Using 'in' operator – returns "True" if the key is present in the dictionary, otherwise returns "False".
  - Using 'not in' operator – returns "True" if the key is not present in the dictionary, otherwise returns "False".
  Syntax: <key> in/not in <dictionary_name>
➢ **Dictionary functions:**
- **len( )** – Returns the number of key-value pairs in the dictionary.
  Syntax:
  len(dictionary_name)

- **clear( )** – Removes all items from the dictionary.
  Syntax:
  dictionary_name.clear( )
- **get( )** – Returns the value of a given key in the dictionary.
  Syntax:
  dictionary_name.get(key)
- **items( )** – Returns all the key-value pairs in the dictionary.
  Syntax:
  dictionary_name.items( )
- **keys( )** – Returns the list of keys used in the dictionary.
  Syntax:
  dictionary_name.keys( )
- **values( )** – Returns the list of values defined in the dictionary.
  Syntax:
  dictionary_name.values( )

❑❑❑

CHAPTER-13
SORTING ALGORITHMS

Revision Notes

- **Sorting** means to arrange a list in ascending or descending order.
- Python has two techniques for sorting lists:
  - **Bubble Sort**
    - Simple sorting technique.
    - Traverses the whole list by comparing its adjacent elements, sorting them and swapping them till the entire list is not sorted.
    - **Algorithm**
      1. Start
      2. Read the array elements and store it in an array A1 [ ]
      3. Store the length of list in Len1
      4. for i = 0 to Len1-1 repeat steps 5 and 6
      5. for j = 0 to Len1-i-1 repeat step 4
      6. if Al[j]>Al[j+l] then swap elements
      7. Display the sorted list
      8. End
    - **Applications**
      – Simpler to understand
      – Fastest and easiest sorting technique
      – One of the first sorting algorithms
      – Useful for huge data sets.
  - **Insertion Sort**
    - Starts from index 1 and not index 0 and places the values in its correct position.
    - It not only compares adjacent items and swaps but also slides up each larger element until it gets to the correct location in the sorted array.
    - **Algorithm:**
      1. Start
      2. Read the array elements and store it in an array A1 [ ]
      3. Store the length of list in Len1
      4. for i = 0 to Len1 repeat steps 4 to 8 else step 8

5. j = A1 .index(i)
6. while (Al[i-ll > Al[j] and j > 0) repeat steps 6 to 7 else step 3
7. A1[j-1], Al[j] = Al[j], Al[j-1]
8. j =j - 1
9. print the list
10. End

- **Applications:**
  - Efficient for small data sets.
  - Very inefficient for large lists.
  - Adaptive
  - Reduces total number of steps if partially sorted list is input
  - Does not require additional memory.

❑❑❑

CHAPTER-14
STRING MANIPULATION

TOPIC-1
STRING MANIPULATION

Revision Notes

- The language creates a structure that holds the individual characters together.
- **String** is a consecutive sequence of characters which are enclosed in single quotes (' ') or double quotes (" ").
- An **Empty String** is a string that has no characters.
- Quotes can be inserted inside a string by placing \ before opening and closing quotes.
- **\n** is used to display the output on the next line of the screen.
- **\t** is used to tabulate the output.
- Multiline strings are represented by enclosing the text in triple quotes ("' '").
- **Traversing** a string means accessing all the elements of a string.
- Each character in a string has an index value.
- Individual elements of a string can be accessed by enclosing index value in [ ].
- Using float or other types for index results in TypeError.
- Left index of a string starts from 0 and right index starts from -1.
- Each character of a string can be accessed using for loop.
  Syntax: for i in string_name:
  print(i)
- String operators
  - **Concatenation** – Creating a new string by adding two strings using +.
    Syntax: string3 = 'string1' + 'string2'
  - **Replication** –Repeating a string from multiple copies of same string using *.
    Syntax: string2 = integer * 'string1'
  - **Membership** – Checks whether a particular character exists in a string or not using 'in' or 'not in'.
    Syntax: <substring> in/not in <string1>
  - **Comparison** – Compares two strings on the basis of ASCII value of the characters using <, >, <=, >=, = =, !=.
    Syntax: string1 <= string2
- **Slicing** – Retrieves a subset of the string.
  Syntax: string_name [start:end]
- Content of the strings cannot be changed after creation. Modifying contents of string by replacing / deleting/ adding a character shows TypeError.
- **Functions in strings**
  - **len**( ) – Returns the length of the string.
    Syntax: len(0_string_name)
  - **capitalize**( ) –Returns copy of the string with first letter in uppercase.
    Syntax: 0_string_name.capitalize( )

- **split**( ) –Breaks a string into substrings at the specified place.
  Syntax: 0_string_name.split([separator [, maxsplit]])
- **replace**( ) – Replaces all occurrences of the old string with the new string.
  Syntax: 0_string_name.replace(old string, new string)

❑❑❑

UNIT-III
Data Mangement-1

CHAPTER-15
RELATIONAL DATABASES

TOPIC-1
Relational Database

Revision Notes

- Data , in the context of database,refer to all the single items that are stored, in a database , ether individually are as a set .
- Database is a collection of related data which stored in computer There & data serve multiple applications in various fields
- A database has the following property
  - It is a coherent collection of data with some inherent meaning , A random assortment as of data can't be connectedly refer as database.
  - A database is a design , build properly with data for a specific purpose.
  - A database can be of any size and complexity
  - A database may be generated and maintain manually or it can be computerized.

Database Management System (DBMS)

- It is a collection of programs that enable users to create and maintain a database
- The DBMS is a general purpose system that facilitates the process of defining, constructing, manipulating and sharing database among various users & applications.
  e.g. ORACLE, MS-ACCESS, FOXPRO, SQL etc.
- Defining a database involves specifying a data type, structure and constraint of the data to be stored in the data base.
- Sharing a database allows multiple users and programs to access the database simultaneously.
- The goal of a DBMS is to provide an environment that is both convenient and efficient to use in
  - retrieving information from the database.
  - String information into the database.
- **Components of Database systems**

(i)	Data
(ii)	Hardware
(iii)	Software
(iv)	Users

- **Data Abstraction**
  Main purpose of a database system is to provide users with an abstract view of the system. The system hides certain details of how data is stored and created and maintained complexity should be hidden from database users.
- **Levels of data abstraction**
  There are several levels of abstraction

(1) Physical Level

- How the data are stored:
  e.g. index, B-tree, hashing

- Lowest level of abstraction:
  e.g. Data compression and encryption techniques
- Complex low level structures

View 1 View 2 View 3

- Next highest level of abstraction
- Describes what data are stored
- Descr Conceptual level ips among data
- Database administrator level

(3) View L Physical level

- Highest level
- Describes part of the database for a particular group of users.
- Can be many different views of a database.
  e.g. tellers in a bank get a view of customer accounts, but not of payroll data.

➢ **Data Model**

- Data model is a collection of conceptual tools for describing data, data relationship, data semantics and consistency constraints.
- There are three different data models as

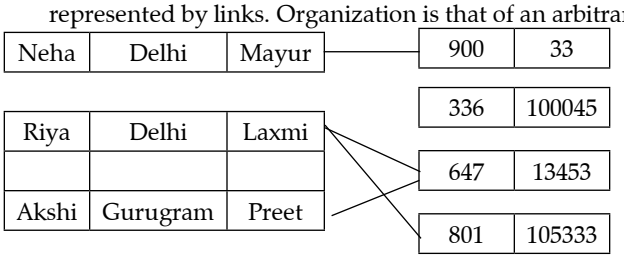
(i) Relational model
(ii) Network model
(iii) Hierarchical model

(i) **Relational model –** Data and relationships are represented by a collection of tables. Each table has a number of columns with unique names e.g. customer, account

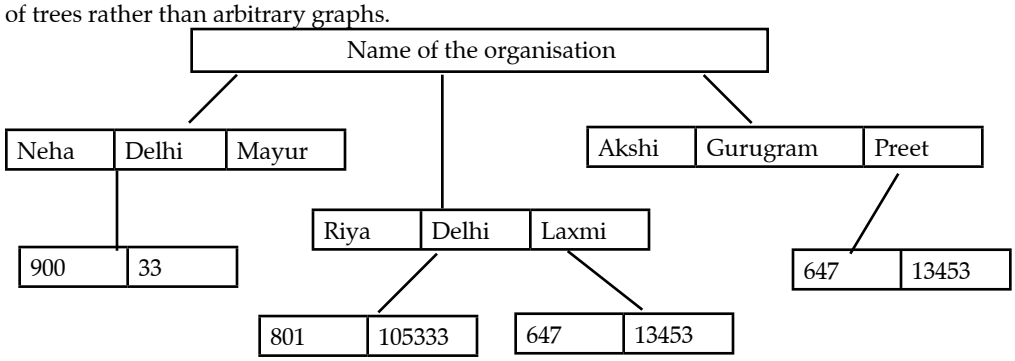
Name	City	Street	Number
Neha	Delhi	Mayur	900
Riya	Delhi	Laxmi	647
Riya	Delhi	Laxmi	804
Akshi	Gurugram	Preet	647

Number	Balance
900	33
336	100045
647	13453
801	105333

(ii) **Network Model –** Data are represented by collections of records. Relationships among data are represented by links. Organization is that of an arbitrary graph.



(iii) Hierarchical Model – It is similar to the network model. Organisation of the records is as a collection of trees rather than arbitrary graphs.



➢ **Relational Model :-**

- The relational model represents the database as a collection of relations. A relation is nothing but a table of values. Every row in the table represents a collection of related data values. These rows in the table denote a real world entity or relationship.
- The table name and column names are helpful to interpret the meaning of values in each row. The data are represented as a set of relations. In the relational model data are stored as table. However, the physical is storage of the data in interdependent of the way the data are logically organized.

➢ **Relational Model Terminology :-**

- **Attribute :-** In a database Management system (DBMS), an attribute refers to a database component such as table. It also may refer to a database field. Attribute describe the instance in the row database. **E.g.** Student, Roll no, Name, etc.
- **Relation :-** It is sometimes used to refer to a table in a relational database but is more commonly used to describe the relationships that can be created between those table in a relational database. Relations have three important properties a name cardinality and degree which is described as :-
- **Name :-** The first property of a relation is its name which is represented by the title or entity identifier.
- **Cardinality :-** It refers to the number of rows (tuples) in relation that defines the uniqueness of data values contained in a column.
- **Degree :-** It refers to the number of column (attributes) in each tuples.
- **Domain :-** It is defined as the set of all unique values permitted for an attribute. For example, a domain of data is the set of all possible valid dates, a domain of integer is all possible whole Number, a domain of day of week is Monday, Tuesday _ _ _ _ Sunday.

➢ **Keys**

- It is data item that allows to uniquely identify individual occurrences or an entity type.
- An entity type usually has an attribute whose values are distinct for each individual entity in the entity set such as attributes is called key attribute.
- A key is normally co-related with one column in table and it might be associated with

(i) Primary Key

- The primary key of a relation can be sail to be a minimal super key.
- The field or group of fields which forms the unique identifier for a table is called the table's primary key.
- The primary key uniquely identifies each record in the table and must never be the same for two records. **E.g.** Emp – code can be primary key for entity set Employees. The primary key should be chosen such that its attributes are never or very rarely changed. For instance, the address field of a person should not be part of the primary key, since it is likely to change.

  Emp – code, on the other hand, is guaranteed to never change, till he is in the organization.

(ii) Candidate Key

- There is only one primary key in a table. But there can be multiple candidate keys. A candidate key is an attribute or set of attributes that uniquely identifies a record.
- These attributes or combinations of attributes are called candidate keys.
- In such a case, one of the candidate key is chosen to be a primary key. The remaining candidate keys are called alternate keys.

(iii) Foreign Key

- In a relation, column whose data values correspond to the values of a key column in another relation is called foreign key.
- In a relational database, the foreign key of a relation would be the primary key of another relation.

(iv) Super Key

- For an entity, it is a set of one or more attributes whose combined value uniquely identifies the entities in the entity set.

  E.g. For an entity set Employees, The set of attributes (Emp-name, Address) can be considered to be a super key. If we assume that there are no two employees with the same name Emp – name and same address.

(v) Alternate Key

If any one of the candidate keys among the differences candidate key available is selected as primary key then remaining keys are called alternating keys.

➢ **Databases** are usually designed to manage large bodies of information. The involves :

- Definition of structures for information storage (data modeling)
- Provision of mechanisms for the manipulation of information (file and systems structure, query processing)
- Providing for the safety of information in the database (crash recovery and security)
- Con currency control if the system is shared by users.

❑❑❑

CHAPTER-16
STRUCTURED QUERY LANGUAGE (SQL)

Revision Notes

- **SQL (Structured Query Language)** consists of commands, each command has an operational part and condition part. Operation is executed through a search in all relations defined in relation database.
- Required solution (list of data) is returned by the command. The data listed in solution satisfies all conditions given in that command.
- SQL is a comprehensive data base language, it has statements for data definition, query and update. Hence, it is both DDL and DML.
- In addition, it has facilities for defining videos on the data base for specifying security and authorization, for defining integrity constrains and for specifying transaction controls. It also has rules for embedding SQL statements into a general purpose programming language such as C or Pascal.
- SQL provides commands for a variety of tasks including
  (i) querying data
  (ii) inserting, updating and deleting rows in a table
  (iii) creating, altering, and dropping objects
  (iv) controlling access into the database and its objects.
- **Type of SQL statements**
  SQL statements can be divided into three major categories
  (1) Data manipulation Language (DML) - These statements consist of queries that retrieve data from tables in a database and statements that change the data in the database. SQL statements under this category are SELECT, INSERT, UPDATE, DELETE ETC.
  (2) DATA Definition Language (DDL) - These statements define the structure of the database DDL consists of these statements that create, alter and drop database objects and statements that grant and revoke privileges and roles to users of the database. Statements under this category are CREATE, ALTER, DROP, GRANT, REVOKE, etc.
  (3) Transaction Control Language (TCL) - These commands manage changes made by data manipu lation language commands. These commands are COMMIT, ROLLBACK, etc.
- **Data Types**
  - **VARCHAR 2 (SIZE) :** It is Variable length characters string having maximum length size bytes. Maximum size is 2000.
  - **CHAR (size) :** It is specifies a fixed length characters string. Maximum size is 255. Default size 255. Trying to insert a value too large for column
  - **VARCHAR (size) :** This data type is currently synonymous with VARCHAR 2 datatype.
  - **NUMBER (PS):** It is used to store fixed or floating point numbers ranging from $1.0 * 10^{-130}$ to $9.99 * 10^{125}$.
  - **P-Precision** or total number of digits range 1 to 38.
  - **S-Scale** or numbers of digits to right of decimal point.
  - **LONG:** It stores variable length character strings containing up to 2 gigabytes.
  - **DATA:** It is used to store and time information. Default format is DD-MON-YY.
- **Creating database objects:** Create command
  Create command used to create database objects.
  Syntax
  CREATE TABLE name
  (Column_name 1 datatype [column constraint],
  Column –name 2 datatype [column constraint],

⋮
Column_name n datatype [column constraint]).

The following statement creates an EMPLOYEE table :

CREATE TABLE EMPLOYEE
(Emp_No Number (6) PRIMARY KEY,
Emp_Name char (30) NOT NULL,
DEPT CHAR (20)
Salary Numbers
);

- **Altering definition of table : Alter Command**

ALTER TABLE command is used to alter the definition of a table in the database.

Syntax

ALTER TABLE Table_name ADD column_name datatype size :

e.g ALTER TABLE employee ADD age numbers;

Some times we need to change the datatype of a column. To do this, we use ALTER TABLE with MODIFY column command syntax

Syntax

ALTER TABLE Table –name Modify column name. datatype **(size):**

e.g ALTER TABLE Employee MODIFY salary numbers (10):

- **INSERT Command**

INSERT command is used to add rows to a table.

Syntax

INSERT INTO Table_name VALUES (data_value1 data_value2);

The number and sequence of data values should match that of columns in the table, If the number of data values less then specify the column names into which data is being entered as illustrated.

INSERT INTO Table_name (column1, column2) VALUES (data_value1, data_value 2);

- To insert null values. NULL may be used,

INSERT INTO EMP VALUES (1001,'sharma',NULL, 3000, NULL);

- **SELECT Command**

SELECT command is used to retrieve the sub part of rows or columns from one or more tables.

Syntax

SELECT column_name FROM Table_name;

Table:Employee

Emp_code	Emp_name	Designation	Salary	Joining-Date
1001	Rahul	Accountant	25000	2011-5-25
1002	Krishna	Clerk	20000	2010-6-19
1003	Akshat	Accountant	22000	2012-7-22
1004	Apoorvi	Clerk	18000	2013-3-17
1005	Nishant	Supervisor	24000	2016-4-23
1006	Sonam	Accountant	22000	2010-5-24
1007	Pihu	Maneger	38000	2012-6-18

e.g SELECT Emp_Name FROM Employee;

Output

Emp_ Name
Rahul
Krishna
Akshat
Apoorvi
Nishant
Sonam
Pihu

If you want to display the details of all employee, asterisk * is used.

Syntax SELECT*FROM Table_name ;
e.g. SELECT* FROM Employee;

- **Using Distinct keyword**

DISTINCT keyword is used to eliminate the duplicate rows from the result of SELECT statement.
Syntax SELECT DISTINCT column_name FROM Table_name
e g SELECT DISTINCT Designation FROM Employee;
Output

Designation
Clerk
Accountant
Supervisor
Manager

- **Using ALL Keyword**

If you use All Keyword instead of DISTINCT, it will give all rows with duplicate from the result of a SELECT statement.
Syntax SELECT ALL column_name From Table_name ;
e.g. SELECT ALL Designation FROM Employe;
Output

Designation
Accountant
Clerk
Accountant
Clerk
Supervisor
Accountant
Manager

- **Using WHERE clause**

WHERE cluse in SELECT statement specifies the criteria for selection of rules to be returned. It gives the particular result based on some condition.
Syntax SELECT column_name 1, column_name 2
FROM Table name WHERE condition;
e.g SELECT Emp_name Designation FROM Employee WHERE salary >22000;
Output

Emp_ name	Designation
Rahul	Accountant
Sonam	Accountant
Pihu	Maneger

- **Logical Operator**

Logical operators are used with WHERE clause. These operators are

- **OR operator *e.g.*** To list the employee's details having Emp_code as 1004 or 1006 from table Employee.

SELECT Emp_code, Emp_name, Designation, salary FROM Employee WHERE (Emp_code =1004 or Emp_code 1006);

- **AND Operator *e.g.*** To list the employee's details having Emp_code as 1005 but with Salary less than 24000.

SELECT Emp_code Emp_name , Designation, Salary FROM Employee WHERE (Emp_code 1005 AND Salary < 24000);

- **NOT operator *e.g.*** To list all employees details whose Emp_code are other than 1004.
SELECT Emp_code Emp_name, Designation, Salary FROM Employee Where (NOT Emp_code =1004).
- **BETWEEN operator** This Operator defines the specified range of values that come to make condition true. In this, lower limit and upper limit are give as a range.

e.g. SELECT Emp_code, Emp_Name , Designation FROM Employee
WHERE Salary BETWEEN 21000 and 25000;
Output

Emp_code	Emp_Name	Designation
1001	Rahul	Accountant
1003	Akshat	Accountant
1005	Nishant	Supervisor
1006	Sonam,	Accountant

- **IN Operator**

  This operator allows you to easily test if the expression matches any value in the list of values. You can also use NOT IN to exclude the rows in your list.

  e.g. To list all the employee's with designation Clerk, Manager

  SELECT Emp_Code, Emp_name, Designation, Salary FROM Employes.

  WHERE Designation IN ('clerk ','Manager);

  But if you want to display the details of those employee whose designation not in Accountant.

  SELECT* FROM Employees

  WHERE Designation NOT IN ('Accountant');

- **LIKE Operator**

  These operator is used in a WHERE clause to search for specified pattern in a column. Pattern are defined using two special wildcard characters.

  - percent (%) matches any substring
  - underscore (_) matches any one character
  - To illustrate pattern matching
  - "abc%" matches any string that start with 'abc'
  - "abc%" matches any string in which abc comes in mid
  - "……" matches any string of exactly four characters.
  - "……%" matches any string of at least 3 characters

  e.g. SELECT Emp_code Emp_name FROM Employee WHERE Emp_Name Like "%00%";

  Output

Emp_code	Emp _Name
1004	Appoorvi

- **ORDER BY Clause**

  This clause is used with. SELECT statement for arranging retrieved data in sorted order. To sort the data in descending order DESC keyword is used while order by clause by default sorts the retrieved data in ascending order.

  Syntax SELECT column_name 1 , column_name 2
  FROM Table_name
  WHERE condition
  ORDER By column_name ;

  e.g SELECT Emp_Code, Emp_name Salary
  FROM Employee
  WHERE Salary <25000
  ORDER By Emp_Name;

- **Output**

Emp_code	Emp_Name	Salary
1003	Akshat	22000
1004	Apoorvi	18000
1002	Krishna	20000
1005	Nishant	24000
1006	Sonam	22000

- **DELETE Command**

  DELETE Command is used to remove the rows of the Table. It does not delete individual field from the table.

Syntax DELETE FROM Table_Name where condition;

e.g. DELETE.FROM Employee WHERE Emp_code = 1004;

SELECT* FROM Employee;

- **Output**

Emp_ code	Emp-Name	Designation	Salary	Joining Date
1001	Rahul	Accountant	25000	2011-5-25
1002	Krishna	Clerk	20000	2010-6-19
1003	Akshat	Accountant	22000	2012-7-22
1005	Nishant	Supervisor	24000	2016-4-23
1006	Sonam	Accountant	22000	2010-5-24
1007	Pihu	Manager	38000	2012-6-18

To remove the all rows from table use

DELETE FROM Table_name;

It will delete all rows but not a table structure.

- **DROP Command**

DROP command is used to delete a whole database or just a table. It destroys the objects like an existing database, table, index, etc.

Syntax DROP TABLE Table_Name;

Or

DROP DATABASE database_name ;

- **UPDATE Command**

UPDATE Command is used to modify the existing records in a table. You can use WHERE clause with the UPDATE command to update the selected rows, otherwise all rows would be affected.

Syntax

UPDATE Table_name

SET Column 1 = value 1 Column 2 = value .....,

WHERE condition;

e.g. To change the salary of all employees to 35000, use this

UPDATE Employee

Set Salary =35000;

But if you want to change the salary to 35000 only for those Employees that have salary less then or equal to 20000, use this

UPDATE Employee

SET Salary = 35000

WHERE Salary <=20000;

To update multiple columns change the Salary and designation of those Employees who have Emp_code 1002

UPDATE Employee

SET Salary = 40000, Designation = Supervisor WHERE Emp_code 1002;

❑❑❑

CHAPTER-17
AGGREGATE FUNCTIONS

Revision Notes

- Aggregate functions is a function where the values of multiple rows are grouped together as input on; certain criteria to from a single value of more significant value of more significant meaning.
- Aggregate functions are also Known as group functions. These functions appear in SELECT command and in ORDER By and HAVING clauses.
- Some of the aggregate functions are MIN , MAX, AVG, COUNT, SUM.

Table : Employee

Emp-Code	Name	Salary	Department	City
101	Neha	30000	HR	Meerut
102	Keshav	25000	Admin	Meerut
103	Amit	45000	HR	Delhi
104	Arif	42000	Accountant	Bengaluru
105	Ruchi	38000	Admin	Delhi
106	Anshul	35000	Admin	Delhi

MIN() function

MIN() function is used to return the minimum or smallest value from the selected column.

Syntax

SELECT MIN (Column –name FROM Table –Name ;

e.g. To show the minimum salary of employee from table named Employee , use following query

SELECT MIN (salary) FROM Employee;

Output

MIN (salary)
25000

Or

SELECT MIN (salary) "MINIMUM" FROM Employee ;

Output

MINIMUM
25000

MAX () function

MAX () function is used to return the maximum or largest value from the selected column.

Syntax

SELECT MAX (column name) FROM Table – name ;

e.g. TO show the maximum salary of employee from the table named Employee , use following query.

SELECT MAX (Salary) FROM Employee;

Output

MAX Salary
45000

or

SELECT MAX (Salary) "Maximum" FROM Employee ;

Output

Miximum
45000

- **AVG() function**

AVG () function is used to return the average value of a specified column.

Syntax

SELECT AVG (column_name) FROM table_name ;

e.g. To find the AVG salary of employee from table named Employee, use Following query.

SELECT AVG (salary) FROM Employee;

Output

AVG
35833.3333

- **COUNT() function**

COUNT() function is used to return the total number of values in the selected column This function returns the number of rows from a table. It includes all nu or duplicate values.

Syntax

SELECT COUNT(*) FROM Table_name;

e.g. To show the total number of records in a table named Employee, use following query

SELECT COUNT (*) FROM Employee;

Output

COUNT(*)
6

DISTINCT clause can also be used with count () function , to return the number of distinct rows in a table .

Syntax

SELECT COUNT (DISTINCT column_name) FROM Table_name ;

e.g. TO show the number of rows in city from table Employee SELECT COUNT (city) "city" FROM Employee ;

Output

City
6

e.g To show the distinct values of column city from Employee .

SELECT COUNT (DISTINCT city) "city" FROM Employee ;

Output

City
3

- **SUM() function**

SUM() function is used to return the sum of values in the selected columns. It works on numeric values and does not include NULL values.

Syntax

SELECT SUM (column_name) FROM Table_name ;

e.g. To show the sum of salary of table Employee .

SELECT SUM (salary) FROM Employee ;

Output

SUM(salary)
21500

- **GROUP BY Clause**

GROUP BY clause is used in a SELECT statement in conjunctions with aggregate functions to group the result based on distinct values in a column .

e.g. To show the sum of salary from table named Employee using SUM() function group by Department column

SELECT Department, SUM(salary) FROM) Employee

Output

Department	SUM(salary)
HR	75000
Admin	98000
Accountant	42000

- **Aggregate Functions and NULL values**
- None of the aggregate functions takes NULL into consideration. NULL is simply ignored by all the aggregate functions.

Table : Employee

Emp_Code	Name	Salary	Department
101	Neha	30000	HR
120	Keshav	NULL	Admin
103	Amit	45000	HR
104	Arif	42000	Accountant
105	Ruchi	38000	Admin
106	Anshul	NULL	Admin

e.g. To count the number of rows from a table named Employee in salary column .

```
SELECT COUNT (salary) FROM Employee ;
```

Output

COUNT (salary)
4

e.g TO calculate the average of salary column from Employee table .

```
SELECT AVG (salary ) FROM Employee;
```

Output

AVG(salary)
38750

❑❑❑

CHAPTER-18
BASICS OF NoSQL DATABASES

Revision Notes

- **DBMS** is Database Management System.
- It was first implemented in 1960.
- Every organisation manages its data and records using DBMS.
- DBMS requires Structured Query Language (SQL) to handle the data.
- SQL is used to access and manipulate databases.
- **RDBMS** is Relational Database Management System is a database that stores data in the form of Tables using SQL commands.
- SQL has some commands that are used to search any data, create tables and records, modify data, and remove table / data in an RDBMS.
- MySql is an RDBMS that is freely available and open source.
- A table containing rows and columns is called a relation is a table, a row is also called tuple or record and a column is also called field.
- **Types of SQL statements:**
  - **DDL –** Data Definition Language – Examples: Create Database, Create Table, Alter Table, Drop Table, Use command.
  - **DML –** Data Manipulation Language – Examples: Select statement, Insert Into statement, Update statement, Delete command.
  - **DCL –** Data Control Language – Examples: Grant and Revoke statements.
  - **TCL –** Transaction Control Language – Examples: Commit and Rollback statements.
- **Steps to start MySQL**
  - Download and install it using internet.
  - Start > All Programs > MySQL > MySQL Server > MySQL Command Line Client
  - MySQL will start with a command prompt like screen.
- To go out from MySQL screen, type
  mysql > quit
- MySQL supports different datatypes for defining different columns in a table and accordingly decides the type of values to be stored in that column.
  - **INTEGER datatype –** It can store whole numbers in the range -2^{31} to $2^{31}-1$.
  - **SMALLINT datatype –** It can store whole numbers in the range -32768 to 32767.
  - **NUMERIC (x,y) datatype –** It is used to store numbers with decimal point where defines total number of digits including digits after decimal point and defines the number of digits to the right of the decimal point. Numeric (9,3) can store 3 digits after the decimal point, 1 digit place for the decimal point and rest 5 digits before (to the left) the decimal point. It can hold 20 significant digits.
  - **DECIMAL (x,y) datatype –** Same as NUMERIC datatype but it can hold 19 significant digits.
  - **CHAR (x) datatype –** It can store characters (maximum 254) in a fixed length string. If the defined string length is left unused, it is then filled with spaces and occupies the defined memory.
  - **VARCHAR (x) datatype –** It can store characters in a variable length string. If the defined string length is left unused, it is not filled with spaces and releases the unused memory.

- **DATE –** It can store date in the format 'yyyy/mm/dd'.
- **TIME –** It can store time in the format 'hh:mm:ss'.
- **BOOLEAN –** It can store logical values true or false.
- **MEMO/LONG –** It can store characters or remarks upto 2 GB per record.

➢ **SQL Commands**

- To check for currently existing databases on the server

  mysql > show databases;
- To create a new database in RDBMS

  mysql > create database <database_name>;
- To open and use the existing database

  mysql > use <datab*ase_name>;*
- To remove a database along with its tables

  mysql > drop database <database_name>;
- To create a new table in a database

  mysql > create table <table_name>
  {
  <column1_name> <datatype> [(size)],
  <column2_name> <datatype> [(size)],
  :
  <columnN_name> <datatype> [(size)]
  };
- To check for currently existing tables in the database

  mysql > show tables;
- To see the structure of an existing table

  mysql > describe <table_name>*;*

  OR

  mysql > desc <table_name>;
- To add a new row or record in the existing table

  mysql > insert into <table_name> values(<value1>, <value2>…<valueN>);

  OR

  mysql > insert into <table_name> (column1, column2..columnN)
  values(<value1>, <value2>…<valueN>);

  OR

  mysql > insert into <table1_name> [(column1, column2..columnN)] select col1, col2..
  colN from <table2_name> [where condition];

  OR

  mysql > insert into <table_name> values('&Column1', '&Column2', ..'&ColumnN');
- To change or edit the values in the existing table

  mysql > update <table_name> set <column1> = <new_value1>,
  <column2> = <new_value2> ..
  where <condition>;
- To remove rows of a table

  mysql > delete from <table_name> [where condition];

  OR

  mysql > truncate table <table_name>;
- To remove an existing non useful table from the database

  mysql > drop table <table_name>;

➢ **Alter table command –** It is used to modify the structure of an existing table by changing the column definition.

- To add a new column in the table

  mysql > alter table <table_name> add (<column_name> <datatype> [(size)]

- To rename a column of the table

  mysql > alter table <table_name> modify <old_col_name> <new_col_name>;

  OR

  mysql> alter table <table_name> rename <old_col_name> to <new_col_name>;
- To add a column with default value in the table

  mysql > alter table <table_name> add (<column1_name> <datatype1> default data);
- To modify an existing column definition

  mysql > alter table <table_name>modify (<column1_name> <datatype1> (size)]);
- To remove a column from the table

  mysql > alter table <table_name> drop (<column_name>);
- To remove a table

  mysql > drop table <table_name>;

➢ Processing of SQL queries
- Query processing means to retrieve and view the data of the tables on the screen.
- SQL Select command is used to fetch data from one or more database tables.

  mysql > select <column_name> from <table_name>;
- To rearrange columns while displaying query results

  mysql > select <column_name3>, <column_name1>, <column_name5> from <table_name> [where <condition>];
- To eliminate duplicate data

  mysql > select distinct <column_name> from <table_name>;

➢ **SQL Operators –** While working with Select statement using Where clause, condition-based query processing is carried out using four types of SQL operators:
- **Arithmetic operators (+, -, *, /, %) –** These operators are used with conditional expressions and for performing simple mathematical calculations. The arithmetic operators with Select command are used to retrieve rows computed with or without reference to any table. For example, *mysql > Select 5+10;*
  - ♦ MySQL permits calculations on the contents of the columns and then displays the calculated result using Select statement.
- **Relational operators (>, <, =, >=, <=, < >) –** A relational operator is a mathematical symbol which is used to compare two values. It is used to compare two values of the same or compatible data types which result in either True or False.

  For example, mysql > Select <column_name1>, <column_name2> from <table_name> where <condition>*;*
- **Logical operators (AND, OR, NOT) –** These operators provide a means to make multiple comparisons with different operators in the same SQL statement.

➢ **SQL Aliases –** These are used to give an alternate name to a database table or a column in a table.

  mysql > Select <column_name1>, <column_name2> from <table_name> as <alias_name> where <condition>*;*
- Alias name can be given to a mathematical expression also.

➢ **Putting text in the Query output –** This makes the query output more presentable by giving a formatted output

  mysql > select <column_name1>, <column_name2> 'some text', <column_name3> from <table_name>;

➢ Conditions can be based on
- Range using
  - ♦ Between…And

    mysql > Select <column_name1> from <table_name> where <column_name> Between <value1> And <value2>;
  - ♦ Not Between…And

    mysql > Select <column_name1> from <table_name> where <column_name> Not Between <value1> And <value2>;
- List using
  - ♦ In

    mysql > Select <column_name1> from <table_name> where <column_name> In <value1, value2>;

- ♦ Not In

  mysql > Select <column_name1> from <table_name> where <column_name> Not In <value1, value2>;
- Pattern using
  - ♦ Like

    mysql > Select <column_name1> from <table_name> where <column_name> Like <pattern>;

➢ **SQL Joins –** An SQL Join clause is used to combine rows from two or more tables, based on a common field between them.
- Types of SQL Joins are:
  - ♦ Cartesian or Cross Product
  - ♦ Equi Join
  - ♦ Self Join
  - ♦ Non-Equi Join
  - ♦ Natural Join

➢ **Cartesian Product –** The Cartesian product is a binary operation and is denoted by X.
➢ **Equi-Join –** It is a simple SQL Join condition that uses = sign as a comparison operator for defining a relationship between two tables on the basis of a common field.
➢ % and _ are wild card characters. The % symbol is used to represent any sequence of zero or more characters. The _ symbol is used to represent a single character.

❑❑❑

UNIT-IV
SOCIETY, LAW AND ETHICS - CYBER SAFETY

CHAPTER-19
SOCIETY, LAW AND ETHICS

TOPIC-1
CYBER SAFETY

Revision Notes

➢ Tips for safely browsing the Internet:
- Update the software regularly.
- Keep the web browser protected.
- Observe safe online behaviour.
- Use strong and unique passwords.
- Download only trusted files and applications.
- Do net get lured with impossible offers.
- Handle online transactions carefully.
- Make payments through secured connections.
- Use ad blocker.
- beware of cookies. Cookies are small text files on the computer storing small pieces of information related to online habits of the user.
- Always bookmark important sites.
- Choose for private browsing through Incognito browsing, Proxy and Virtual Private Network (VPN).

➢ Cyber safety refers to the safe and responsible use of information and communication technology.
➢ Identity theft is a type of fraud that involutes using someone else's identity to steal money or gain other benefits.
➢ Whenever a user visits a website, the browser may reveal the location via the device's IP address.
➢ Browsers also provide search and browsing history.
➢ Websites track their visitors through:

- **IP Address:** It is a device's unique address to connect to the Internet. From this IP Address, a website can determine the rough geographical location of the device.
- **Cookies :** These are small pieces of information websites can store in the browser.
- **HTTP Referer** is a link on a web page to outside website. It automatically provides the visitor's information, such as IP address, location, web browser, machine type, etc. to the linked website.
- **Super cookies** are persistent cookies that come back even after being deleted.
- **User Agent** is a string or a line of text, that browser sends to every website, a device connects to. It tells the web server about the web browser and the OS being used.

- Anonymous browsers allow users to view websites without revealing any personal information of the user like their IP address, machine type, location etc.
- Confidentiality of Information ensures that only authorize users get access to sensitive and protected data.
- Confidentiality of information is maintained by:
  (a) Encryption of data.
  (b) Use of firewall
  (c) Private browsing
  (d) Accessing safe sites
  (e) Carefully handling e-mails
  (f) Avoiding use of public network.
- Cybercrime is a criminal act facilitated by use of electronic gadgets and information systems through internet. Cyber criminals are also called hackers.
- Some common cybercrimes are:
  (a) Cyber Trolls - These are the persons who start quarrels on the internet to distract a community and provoking the readers to behave emotionally.
  (b) Cyber Bullying - This is the attack upon an individual through electronic means to abuse or intimidate others.
  (c) Cyber Stalking - It is the use of internet to harass an individual or an organisation.
  (d) Spreading Rumors - This is the act of creating fake e-mail IDs, masking actual identification and posting false information on social media.
- Cyber Forensics - It is an electronic discovery technique used to determine and reveal technical criminal evidence.
- The Information Technology Act, 2000 or IT Act, 2000 is the primary law in India dealing with cybercrime and electronic commerce.

TOPIC-2
SOCIAL NETWORKING

Revision Notes

- Social Media refers to web and mobile technologies that people use to share content and experience online.
- A social networking site is an application or online platform that allows users to create a public profile and interact with other users on the website.
- Online friends are the connections that are made online.
- **Facebook :** It is a platform that allows people to share information in the form of posts, photos, videos, etc. in an easy and entertaining way. Users can also post comments and like posts shared by their friends.
- **Twitter :**It is a microblogging site where people communicate in short messages called tweets. The character limit for there twitter is 280 characters (It was 140 characters earlier).
- **LinkedIn** is a social networking site for professional the where they can create and upload their resumes or other work and experience related information.
- YouTube is a video-sharing service users can watch, like, upload and comment their own videos.
- Blogs or Weblogs are online journals displaying information in reverse chronological order.
- **Instagram** is most popular social networking site for sharing photos and short videos.
- Whatever one does online, leaves a permanent footprint storing trails of his /her online activities, This is digital footprint or digital tattoos.
- **Privacy Setting control:**

(a) who all can see your posts.
(b) who can send request to you.
(c) what post of your personal info is visible to others.

- Do's and Dont's of Social Networking:
  - Do not post personal information.
  - Do not give your password to anyone except your parents/guardians.
  - Delete unwanted messages.
  - Take care while sharing posts.
  - Update privacy settings on social media.
  - Respect privacy of others.
  - Update your software.
  - Be aware of spams.
  - Take caution for cyber bullying.
  - Control the troll.

❑❑❑

CHAPTER-20
ONLINE ACCESS AND SAFE COMMUNICATION

TOPIC-1
Safely Accessing Websites

Revision Notes

- Threat is a potential violation of security.
- Attack is when threat is actually executed
- People who execute attach are called attackers .
- Malware is an unwanted software that infects our computer and makes it behave in a way that's not acceptable to us.
- Common threats to a computer are
  (i) Viruses: (a) Worms (b) Trojans
  (ii) Spyware
  (iii) Adware
  (iv) Spamming
  (v) PC Intrusion : (a) Denial of service (b) Sweeping (c) Password Guessing.
  (vi) Phishing
  (i) VIRUS
    - Virus is a malicious program that damages data and files, thereby causing the system to malfunction.
    - A virus can attack any part of the software such as boot block, operating system, files etc.

    (a) Worms are programs that keep on replicating thereby unnecessarily eating up the disk space.
    (b) Trojan Horses is a malicious program that is disguised as harmless. It may delete or damage file.
- **Damage caused by VIRUS**
  - Viruses damage documents or may even delete specific files. Most of the time a virus is programmed to damage system files.
  - Virus tend to slow down the system by executing itself in the background.
  - Some viruses attach themselves to the contacts in victims address book and then spread through emails.

  (ii) Spyware is a software that spies on the activities of a computer and reports it to the people who can pay for it. These get installed on a computer without the user's consent by `Piggybacking' a file or from internet. These remain active unless someone switches them off or removes them properly

  A spyware affects the system in Following ways.
  (a) It monitors information about the owners computing habits, record the keystrokes when the user enters some information and thus lead to identity theft.
  (b) It alters PC settings like the browser home page or appearance of desktop.
  (c) It robs off the system or PC speed or internet access efficiency.

(iii) ADWARE are the software that deliver unwanted ads to your computer. Though this happens with the user's consent most of the times.
- An adware just like spyware tracks your computing habits and data to reduce targeted ads that pop-up on your screen.
- An adware infected PC displays a lot of frequent pop-up ads.
- As the adware is active in the background and there is a frequent display of ads the speed of system is inhibited.

(iv) Spamming refers to bulk mails that are sent by an identified or unidentified source.

It is two types

(a) malicious - In this case the attacker keeps on sending bulk mails until the mail server runs out of space.

(b) Non malicious – In this form bulk mail sent to many accounts mainly for the purpose of advertisement.

Damaging effects of spam
- Billions of spam messages that get circulated acros the internet disrupt mail delivery and degrade system performance and overall productivity.
- Deleting spam is time consuming and wastes a lot of valuable time.
- Spam messages may contain fraudulent messages and sometimes act as VIRUS carriers.

(v) PC intrusion – Computers connected to internet are under a constant attack from cyber criminal PC intrusion can occur in following form

(a) Sweeper Attack – Hackers use a malicious program to delete all the data from the system.

(b) Denial of services – In this attack all the system resources are used up unnecessarily and the system comes to halt.

(c) Password guessing – Hackers try to crack password of a system to break in the system and then use the system resources data and information for causing substantial damage.

(vi) Eavesdropping is the unauthorised monitoring of other people's communications.
- Eavesdropping can be carried out through any of the communication devices and media such as telephone, emails, Instant messaging, chat rooms, social networking sites, etc.
- When a message is intercepted in between its routes of transmission and defaced by the attacker, it is called Man-in-the Middle attack.

(vii) When an imposter tricks a user by sending him authentic looking email there by acquiring sensitive information such as usernames, passwords, credit card information, etc., it is called Phishing:
- Pharming occurs when a legitimate URL leads to a bogus website by a hacker.

(viii) Cookies are small piece of data sent from a website and stored in a user's web browser. Threats related to cookies are

(a) Session Data – On the sites that you visit regularly your username and password information is pulled from a tracking cookie. Chances are that someone may acquire your cookies and find the encryption key to get your passwords.

(b) Invasion of Privacy by cookies that track patterns of someone's web activity.

(c) Cookies of your web activity stored in a public computer a larger risk of information being accessed without authorisation.

TOPIC-2

Solutions to Computer Security Threats

Revision Notes

- Solution to computer security threats can be of two types
  - **(a)** Active Protection
  - **(b)** Preventive measures
  - **(a)** Active Protection involves installing and properly using an antivirus software that includes internet security which includes protection against threats such as viruses, spyware and PC intrusion.
  - **(b)** Preventive measures include steps to prevent security issues from arising
- **Active protection**
  - Use Anti-Virus and Anti – spyware software.
  - Download updates regularly
  - Run frequent full system scans.

- Use Anti –spam software
- Authorization while logging in
- Authenticate the user
- Use of Firewall
- Keep your computer offline to protect yourself from phishing and pharming
- Contact credit agencies to report any possibilities of Identity theft.

➢ **Preventive Measures**

- Keep your system up-to-date
- Use caution when downloading files on the internet
- Be careful with email
- Disable cookies
- Keep your email address private
- Use encrypted connection always.
- Install personal firewall.
- Avoid conducting online transactions on public
- Install internet security software
- Use proper file access permissions when sharing files on the internet
- Disconnect from internet when not in use
- Don't open emails from unknown sources
- Check the security guideline of websites
- Instead the of clicking on an embedded link type the general link.
- Do not click , when in doubt

➢ An internet firewall is a device or software that is designed to protect your computer from data and viruses that you do not want. It can be implemented in two ways

(a) **Software firewall –** It is a special type of computer software running on a computer that protects it from PC Intrusion, Trojan or e-mail worms.

(b) **Hardware firewall** is a physical equipment. It may be another computer also.

❑❑❑