PG CA S1-03 (DSC)

TWO YEAR MASTER'S DEGREE COURSE

SEMESTER - I

FUNDAMENTALS OF NETWORKING AND WEB TECHNOLOGY

MSc IT

Hiranya Chandra Bhuyan School of Science and Technology

KRISHNA KANTA HANDIQUI STATE OPEN UNIVERSITY

Subject Experts

Prof. Anjana Kakoti Mahanta, Gauhati University

Prof. Hiren Kumar Deva Sarma, Gauhati University

Dr. Lipi B Mahanta, IASST, Guwahati

Dr. Sanjib Kr. Kalita, Gauhati University

Course Co-ordinator : **Ridip Dev Choudhury,** KKHSOU

SLM Preparation Team

UNITS	CONTRIBUTORS
1,3,4,5,7	Dr. Tapashi Kashyap Das, KKHSOU
2	Dr. Sruti Sruba Bharali, KKHSOU
6	Dr. Chakradhar Das, Lakhimpur Polytechnic
8,9	Arabinda Saikia, KKHSOU
10,11,12,13,14	Dr. Ridip Dev Choudhury, KKHSOU
15	Sankhapani Bharali, Girls' Polytechnic

Editors

Content Dr. Jyotiprokash Goswami, Assam Engineering College
(Unit 1,2,3,4,5,6,7,8,9,15)

Dr. Sruti Sruba Bharali, KKHSOU
(Unit 10,11,12,13,14)

Structure, Format & Graphics : Dr.Sanjib Kr. Kalita, Gauhati University
(Former Faculty, KKHSOU)

Print : July, 2023

Copyright © Krishna KantaHandiqui State Open University

This Self Learning Material (SLM) of the Krishna Kanta Handiqui State University is made available under a Creative Commons Attribution-Non Commercial-ShareAlike4.0 License (International): http.//creativecommons.org/licenses/by-nc-sa/4.0.

Printed and published by Registrar on behalf of the Krishna Kanta Handiqui State Open University.

Headquarters: Patgaon, Rani Gate, Guwahati-781017
City Office: Resham Nagar, Khanapara, Guwahati-781022 Web: www.kkhsou.in

MSc IT

FUNDAMENTALS OF NETWORKING AND WEBTECHNOLOGY

DETAILED SYLLABUS

Page Nos.

Unit 1: Introduction to Computer Network **7-20**

Goals of Computer Network, Broadcast and Point-to-Point Networks, Connection- Oriented and Connection-Less Services, Types of Network, Local Area Network, LAN Transmission Methods, Peer-to-Peer LAN & Server Based LAN, Metropolitan Area Network, Wide Area Network

Unit 2: Data Communication Modes **21-32**

Data Communication,Types of Communication, Modes of Communication

Unit 3: Network Devices **33-42**

Network Interface Card (NIC), Modem, Repeater, Bridges and Gateways, Hub, Switch, Router

Unit 4: Network Topology **43-52**

Network Topology, Bus Topology, Ring Topology, Star Topology, Mesh Topology, Tree Topology

Unit 5: DataTransmission **53-63**

Parallel Transmission, Serial Transmission, Asynchronous Transmission, Synchronous Transmission, Switching Techniques, Circuit Switching, Message Switching, Packet Switching

Unit 6: Network Models **64-79**

Design Issues of Layers, Protocol Hierarchy, ISO-OSI Reference Model: Functions of Layers, Important Terminologies, Connection-Orientedand Connectionless Service, TCP/IP Reference Model, Comparison of ISO-OSI&TCP/IPModel

Unit 7: Transmission Media **80-98**

Transmission Medium, Guided Media: Coxial Cable, Twisted Pair Cable,Fiber Optics Cable, Unguided Media: Radio Waves, Infrared, Microwave, Satellite, Wireless LANs (IEEE802.11)

Unit 8: Internet **99-122**

Definition of Internet, Internet Architecture, Peer-to-Peer, Client-Server, Accessing Technique,Internet Service Providers, Organization of Internet,

Internet Protocol Suite, IP Address, Domain Name System, Uniform Resource Locators (URL), Socket, Application of Internet.

Unit 9: Word Wide Web **123-135**
Web Page, Web Browsers, Web Server, Web Search Engines, Web Cookies, Technologies in WWW.

Unit 10: Introduction to HTML **136-155**
Basics of HTML, HTML Editor, HTML Tags, Document Structure Tags, Formatting Tags, Hyperlink and Image tags

Unit 11: HTML Lists and Tables **156-167**
HTML Lists: Unordered, Ordered, Definition Lists; HTML Table tags;

Unit 12: Advanced HTML Tags **168-184**
Frametags, HTML Form, Additional Advanced HTML Tags.

Unit 13: Introductionof XML **185-200**
Introduction to XML, Comparison of HTML and XML, XML Basics, Introduction to DTD

Unit 14: Introduction to CSS **201-218**
Cascading Style Sheets (CSS), Inline Style, Embedded Style, External Style Sheet, Imported Style Sheet, Ruleset, @ rule, Class Selector, ID Selector, Contextual Selector, Attribute Selector, CSS Properties – background properties, text properties and border properties

Unit 15: Introduction to JavaScript **219-236**
Client Side Programming, Introduction to JavaScript, Advantages of JavaScript, Basic Programming Concepts: Variables, data types, operators; Control Structures in JavaScript, Array; Function

COURSE INTRODUCTION

Welcome to the comprehensive course on “Fundamentals of Networking and Web Technology.” This course is designed to provide you with a solid understanding of the key concepts and principles underlying computer networks and web technologies. Through a structured approach, we will explore various topics to equip you with essential knowledge and skills in this dynamic field.

In the initial units, we will delve into the fundamentals of computer networks, covering topics such as network goals, broadcast and point-to-point networks, connection-oriented and connectionless services, and different types of networks including local area networks (LANs), metropolitan area networks, and wide area networks. You will also gain insights into network devices such as network interface cards, modems, repeaters, bridges, gateways, hubs, switches, and routers.

As we progress, we will explore network topology, examining different types such as bus, ring, star, mesh, and tree topologies. Additionally, we will dive into data transmission methods, including parallel and serial transmission, asynchronous and synchronous transmission, and switching techniques such as circuit switching, message switching, and packet switching.

To gain a comprehensive understanding of networking, we will study network models, including the ISO-OSI reference model and the TCP/IP reference model, understanding their functions, protocol hierarchy, and important terminologies. We will also explore transmission media, both guided and unguided, such as coaxial cables, twisted pair cables, fiber optics cables, radio waves, infrared, microwave, satellite, and wireless LANs.

In the latter part of the course, we will shift our focus to web technologies. You will learn about the Internet, its architecture, accessing techniques, Internet Service Providers (ISPs), IP addressing, the Domain Name System (DNS, Uniform Resource Locators (URLs), and the application of the Internet. We will then dive into the World Wide Web (WWW), exploring web pages, web browsers, web servers, web search engines, web cookies, and various technologies employed in the WWW.

Furthermore, we will introduce you to HTML, the backbone of web development, covering the basics of HTML, document structure tags, formatting tags, hyperlink and image tags, lists, tables, and advanced HTML tags. Additionally, you will gain knowledge of XML, its comparison with HTML, XML basics, and an introduction to Document Type Definitions (DTD). Lastly, we will delve into Cascading Style Sheets (CSS) for web page styling, dynamic HTML (DHTML), and the Document Object Model (DOM).

By the end of this course, you will have a strong foundation in networking and web technology, enabling you to understand the core concepts, design principles, and practical applications in this exciting field. Whether you aspire to pursue a career in networking, web development, or related areas, this course will provide you with the necessary knowledge and skills to embark on your journey with confidence.

Each unit includes some along-side boxes to help you know some of the difficult, unseen terms. Some “EXERCISES” have been included to help you apply your own thoughts.You may find some boxes marked with: “LET US KNOW”. These boxes will provide you withsome interesting and relevant additional information. Again, you will get “CHECK YOUR PROGRESS” questions. These have been designed to self-check your progress of study. Itwill be helpful for you if you solve the problems put in these boxes immediately after you go through the sections of the units and then match your answers with “ANSWERS TO CHECK YOUR PROGRESS” given at the end of each unit.

UNIT 1 : INTRODUCTION TO COMPUTER NETWORK

UNIT STRUCTURE

1.1 Learning Objectives

1.2 Introduction

1.3 Goals of Computer Network

1.4 Broadcast and Point-to-Point Networks

1.5 Connection-Oriented and Connection-Less Services

1.6 Types of Network

- 1.6.1 Local Area Network
  - 1.6.1.1 LAN Transmission Methods
  - 1.6.1.2 Peer-to-Peer LAN & Server Based LAN
- 1.6.2 Metropolitan Area Network
- 1.6.3 Wide Area Network

1.7 Let Us Sum Up

1.8 Further Reading

1.9 Answers to Check Your Progress

1.10 Model Questions

1.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- define the goals of computer network
- learn about broadcast and point-to-point networks
- describe connection-less and connection-oriented services
- describe the types of computer network
- differentiate server-based and peer-to-peer LANs

1.2 INTRODUCTION

A computer network is a collection of several computers and terminals that shares information across wireless or wired technology. The

main goal of a network system is the transfer and exchange of data between the computers and terminals.

In this unit, we will begin our discussion with the goals of computer network. We will introduce various concepts associated with computer network such as the concept of broadcast and point-to-point network, connection-less and connection oriented services, various internetrking devices etc. We will further discuss the various types of network together with the concept of network topology. At the end, we will briefly discuss the different transmission types and communication modes followed by various switching techniques.

Before discussing the technical issues in detail, it is worth devoting some time to pointing out why people are interested in computer networks and what they can be used for.

1.3 GOALS OF COMPUTER NETWORKS

Before the advent of computer networks, papers and diskettes were the only means to share information. Now a days, computer networks which comprise a number of computers as well as other devices like printers, scanners etc. are widely used to share resources in a more efficient and faster way.

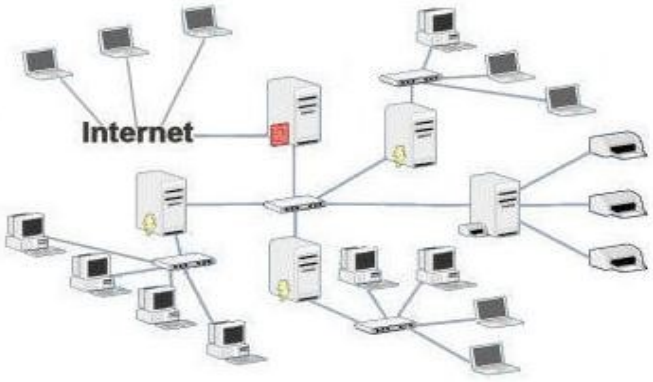


Fig. 1.1 : A typical computer network

In a network, whether big or small, all the devices are interconnected to transmit and receive data from one another. These connections are usually made not only by copper wires; instead, fiber optics, microwaves, infrared,

and communication satellites are also used for efficient communication. The main goals of these networks can be summarized as follows :

- **Resource Sharing :** Using computer networks, it is possible to share programs, data and other resources among several users on the network. The sharing of resources is independent of the physical location of the user and the resource. Files on a particular user's computer can be shared on the network or files can be placed on a file server which provides a central location for all files needed by the users on the network. Users can also share devices such as printers, CD-ROM drives, and hard drives etc. It also makes upgrading an application easier because the upgrade only has to be performed only on the server itself. Thus, we do not need seperate resource for each computer.
- **High Reliablity :** A second goal is to provide high reliability by having alternative sources of supply. For example, all files could be replicatied on two or three machines; so if one of them is unavailable, the other copies could be available.
- **Cost Reduction :** Resource sharing automatically reduces cost and hence money can be saved. For example, let us there are ten users and each user requires a printer. If they could have been working individually, then ten printers would have to be purchased. If these ten users are allowed to work in a network, then only two or three printers would be sufficient.
- **Communication Medium :** Computer networks provide a powerful communication medium among the people who are in a geographically same or different location. A file that was updated or modified on a network, can be seen by the other users on the network immediately. Thus, it becomes easy for two or more peoples living far apart to work on a same project by dividing it using a network. They can write programs, can discuss or can even change or modify some data using a network while they are far off. Otherwise, they will have to wait for several days for

communication through letters or some other media. Thus, it makes speedy co-operations and enhances human to human communication.

1.4 BROADCAST AND POINT-TO-POINT NETWORK

On the basis of *transmission technology*, computer networks can be divided into two categories: *broadcast* and *point-to-point*.

- ***Broadcast networks*** have a single communication channel which is shared by all the machines on the network. Here, messages are broken into *packets* which are then broadcast to all machines in the channel. A packet sent by any machine is received by all the machines but only that machine processes the packet for which it is intended. An address field (destination address), within the packet specifies for whom it is intended. Upon receiving a packet, a machine checks the address field. If the packet is intended for itself, it processes the packet; if it is intended for some other machine, it is just ignored. However, in broadcast networks, we have the problem of deciding who uses the channel, if there is competition for it. For this, *medium access control* (MAC) is used to determine who goes next.

  As an analogy, let us imagine a scenario where a teacher is taking the attendance in a particular class. He/she will call every roll number in the classroom, all will listen but only the respective student will respond. Broadcast implies "*sending a signal where multiple parties may hear a single sender*".

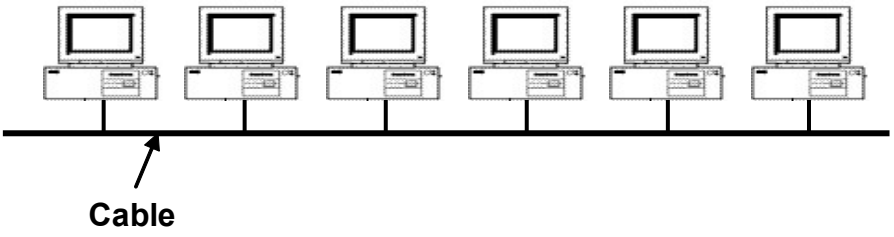


Fig. 1.2 : A Broadcast Network

- Again, let us imagine a scenario that a teacher is taking the attendance in an examination hall. He/she will go to each member of the room and will take the attendance individually. Here, only one person listen and he only responds to the call. ***Point-to-point network*** is a method of communication where one "point" (person or device or entity) speaks to another entity. Point-to-point network consists of many connections between individual pairs of machines. Packet, on this type of network, may have to visit one or more intermediate machines to reach from the source to the destination. Often multiple routes of different lengths are possible in point-to-point networks. Therefore, routing algorithms are required to determine the best routes out of multiple available routes.

1.5 CONNECTION-ORIENTED AND CONNECTION-LESS SERVICES

On the basis of *acknowledgement sent by the receiver*, there are two distinct techniques used in data communications to transfer data. These

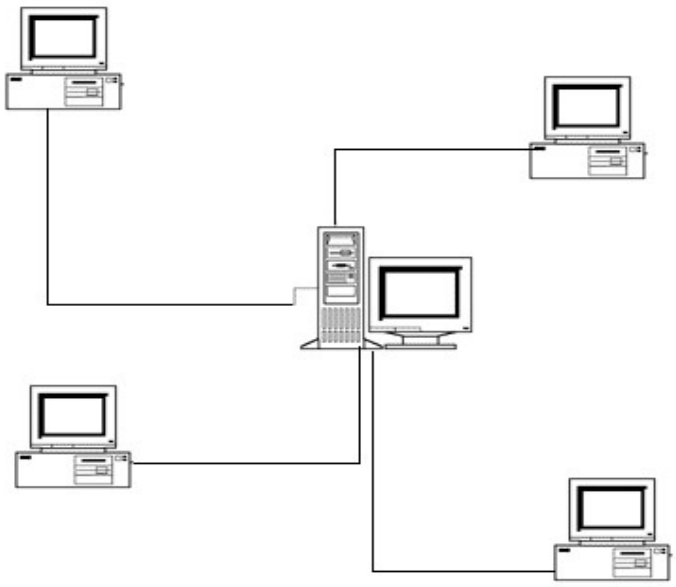


Fig. 1.3 : Point-to-Point Network

are: *connection-oriented* and *connection-less services*. Each service can be characterized by a quality of service. In general, a reliable service is implemented by having the receiver acknowledge the receipt of each message; so the sender is sure that it has arrived.

- ***Connection-oriented service*** is modelled after the telephone system. For making a call to someone, first we have to pick up the phone and dial the number. After the connection is established, we talk, and then hang up.

  Connection-oriented service requires a connection to be established before any data can be sent. It sets up virtual links between the end systems through a network. After the connection is established, the data is transferred. As soon as the transmission is completed, the service user releases the connection. This procedure requires a specific acknowledgement for the information whether the connection is established or not. This method is often called a "reliable" network service. In some cases, when the connection is established, the sender, receiver, and subnet conduct a negotiation about the parameters to be used such as maximum message size, quality of service required, and other issues. Typically, one side makes a proposal and the other side can accept it, reject it, or make a proposal as an alternative to the earlier proposal.
- The analogy of sending letters and postcards best explains the ***connection-less service.*** Each message (letter) carries a full destination address, and each one is routed through the system independent of all the others. Normally, when two messages are sent to the same destination, the first one sent will be the first one to arrive. But it is also possible that the first one sent can be delayed so that the second one arrives first. In a connection-less service, there is no initial end-to-end setup for a session; each packet is independently routed to its destination. The sender simply starts sending packets (called datagrams)

to the destination. This service does not have the reliability of the connection-oriented method, but it is useful for periodic burst transfers. Neither system must maintain state information for the systems that they send transmission to or receive transmission from.

1.6 TYPES OF NETWORK

Whenever we have a set of computers or networking devices to be connected, we make the connections, depending on the physical layout and our requirements. On the basis of geographical area covered, computer networks can be classified under the following three categories:

- Local Area Network
- Metropolitan Area Network
- Wide Area Network

1.6.1 Local Area Network

A ***local area network*** or **LAN**, is a high-speed data network that covers a relatively small geographic area such as a building, a laboratory, or a school. It typically connects workstations, personal computers, printers, servers, and other devices. LANs offer computer users many advantages, including shared access to devices and applications, file exchange between connected users, and communication between users via electronic mail and other applications. LANs differ in the way the computers are connected (i.e., their topology), how information moves around the network (i.e., their transmission technology) and their size.

IEEE (Institute of Electrical and Electronic Engineers) is a US publishing and standards organization responsible for many LAN standards such as the 802 series. *IEEE 802.3,* popularly called *Ethernet,* is the most popular LAN, usually operating at 10 Mbps to 10 Gbps and is present in most large organizations and offices. In a typical LAN configuration, one computer is designated as the *server*.

It stores all of the software that controls the network as well as the software that can be shared by the computers attached to the network. Computers connected to the server are called *workstations*. Two most commonly used LAN implementations are depicted below:

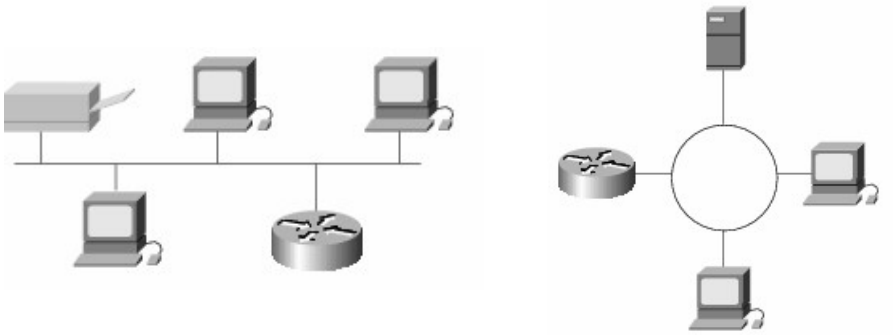


Fig.1.14 : (a) Ethernet or IEEE 802.3 (b) Token Ring or IEEE 802.5

LAN protocols function at the lowest two layers, i.e., the *physical* and the *data link layer* of the OSI reference model. We will discuss the layer concept of ISO-OSI reference model in the next unit which is “Network models”.

1.6.1.1 LAN Transmission Methods

LAN data transmissions fall into three classifications: *unicast*, *multicast*, and *broadcast*. In each type of transmission, a single packet is sent to one or more nodes.

- ***Unicast :*** Unicast is a one-to-one transmission method in which the network carries a message to one receiver, such as from a server to a LAN workstation. In a unicast environment, even though multiple users might ask for the same information from the same server at the same time, such as a video clip, duplicate data streams are sent. Unicast sends separate data streams to each computer requesting the data, thereby flooding the network with traffic.
- ***Multicast :*** Multicast is a one-to-many transmission method

in which the network carries a message to multiple receivers at the same time. Multicast is similar to broadcasting, except that multicasting means sending to a specific group, whereas broadcasting implies sending to everybody, whether they want the traffic or not. When sending large amounts of data, multicast saves considerable network bandwidth because the bulk of the data is sent only once. The data travels from its source through major backbones and is then multiplied, or distributed out, at switching points closer to the end users. This is more efficient than a unicast system, in which the data is copied and forwarded to each recipient.

- ***Broadcast :*** Concept of broadcast is already discussed in previous section. *Broadcast* is a one-to-all transmission method in which the network carries a packet to all devices at the same time, but a particular machice for which the packet is intended accepts it.

1.6.1.2 Peer-to-Peer LAN & Server Based LAN

On a LAN, we expect to share files, programs, or printers, all without being particularly aware of where the physical resources we are using are actually located. LANs providing these types of services are typically set up either as "*peer-to-peer*" or "*client-server*" LANs, or perhaps as a combination of the two.

- **Peer-to-Peer LAN**

  All the machines on a peer-to-peer LAN are equal. Provided that the file's owners give permission, a file on machine A can be accessed from machine B, and vice versa. Peer-to-peer LANs do not require any one machine to be a dedicated, high-performance server; service by a peer-to-peer LAN is often cheaper for this reason. Peer-to-peer LANs work well when only a small number of machines are connected to it. But as the size of the LAN grows, peer-to-peer services can

become quite disorganized. To serve all its peers, each machine on the LAN must be powerful enough and for this reason its cost increases. For larger LANs, the dedicated client-server LAN architecture becomes more cost effective.

Advantages :

Less initial expense – No need for a dedicated server.

Setup – An existing operating system (such as Windows XP) of the machine may only need to be reconfigured for peer-to-peer operations.

Disadvantages :

Decentralized – No central repository for files and applications.

Security – Does not provide the security available on a client/server.

- **Client-Server LAN**

  A client-server LAN consists of one or more server machines on which shared files and programs reside and many client machines where people do their task. The LAN server machines usually have higher configuration and are fast because they must serve many users, while the client machines need only to be fast enough for one person to use at a time. Shared printers are either attached directly to a server, or to a print server (a specialized computer attached to the network), or to a personal computer on network that acts as a print server.

  Advantages :

  Centralized – In case of client-server architecture, resources and data security are controlled through the server.

  Scalability – Any or all elements can be replaced individually as needs increase.

  Flexibility – New technology can be easily integrated into system.

  Accessibility – Server can be accessed remotely and across

multiple platforms.

Disadvantages :

Expense – Requires initial investment in dedicated server.

Maintenance – Large networks will require a staff to ensure efficient operation.

Dependence – When server goes down, operations will cease across the network.

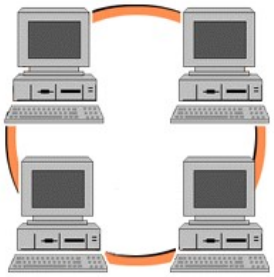


Fig.1.15 : Peer-to-Peer

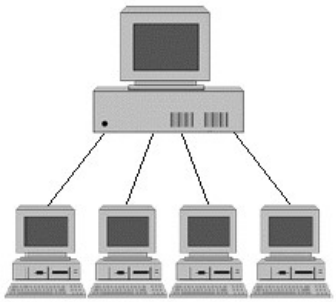


Fig. 1.16 : Client-Server

1.6.2 Metropolitan Area Network

A ***metropolitan area network,*** or **MAN**, is basically a bigger version of a LAN and normally uses similar technology. It might cover a group of nearby corporate offices or a city and might be either private or public.

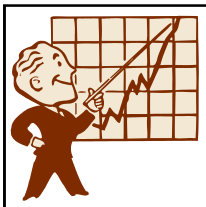
A MAN often provides efficient connections as it has high-speed transmission capabilities which uses some types of telecomminication components to handle long-distance transmission. One very common example of MAN is the cable television network. Another important example is the high-speed wireless Internet access, which has been standardized as *IEEE 802.16.*

1.6.3 Wide Area Network

A ***wide area network*** or **WAN** is a computer network covering multiple distance areas, which may spread across the entire world. WANs often connect multiple smaller networks, such as local area networks (LANs) or metropolitan area networks (MANs).

Typically, a WAN consists of a number of interconnected switching nodes. A transmission from any one device is routed through these internal nodes to the specified destination device. These nodes are not concerned with the content of the data; rather, their purpose is to provide a switching facility that will move the data from node to node until they reach their destination. Traditionally, WANs have been implemented using either *circuit switching* or *packet switching* technologies. More recently, frame relay and *ATM* (*Asynchronous Transfer Mode*) networks have assumed major roles. Frame relay provides higher data rates, lowar costs, efficient handling of bursty data transmission in less expenditure. ATM offers more bandwidth to end users at less cost.

The ***Internet*** is the best known example of a WAN. Some segments of the Internet are also WANs in themselves. The Internet is a system of linked networks that are worldwide in scope and facilitate data communication services such as *remote login*, *file transfer*, *e-mail*, the *World Wide Web* etc. With the rise in demand for connectivity, the Internet has become a communications highway for millions of users. The Internet was initially restricted to military and academic institutions, but now it is a full-fledged conduit for any and all forms of information and commerce. Internet websites now provide personal, educational, political and economic resources to every corner of the planet.



CHECK YOUR PROGRESS

Q.1. What are the three LAN transmission methods?

Q.2. What are the basic difference between point-to-point and Broadcast network?



1.7 LET US SUM UP

This unit provides the basic concept of computer networking. **Networking** is the linking of computers (not necessarily over large distances) so they can communicate, sharing hardware and software, thus uniting processing power. Various concepts such as connection-oriented and connection-less services, broadcast and point-point networks etc. are also presented in this unit. We are also acquainted with the different types of computer networks with their relative advantages and disadvantages.



1.8 FURTHER READING

- *"Computer Networks",* Andrew S. Tanenbaum, Prentice Hall India.
- *"Data and Computer Communications",* William Stallings, Pearson Prentice Hall.



1.9 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : Unicast, Multicast, Broadcast.

Ans. to Q. No. 2 : Broadcast networks have a single communication channel which is shared by all the machines on the network. Broadcast implies sending a signal where multiple parties may hear a single sender. Point-to-point network is a method of communication where one 'point' (person or device or entity) speaks or communicates to another entity.



1.10 MODEL QUESTIONS

Q.1. Define computer network. What are the goals of computer networks?

Q.2. What is the difference between connection oriented and connection less services?

Q.3. Distinguish between peer-to-peer LAN and client-server LAN.

Q.4. Differentiate between broadcast and point-to-point network.

Q.5. Mention the advantages and disadvantages of Peer-to-Peer LAN.

Q.6. What is Ethernet?

Q.7. Discuss briefly about WAN and MAN.

****

UNIT 2 : DATA COMMUNICATION MODES

UNIT STRUCTURE

2.1 Learning Objectives

2.2 Introduction

2.3 Data Communication

2.4 Types of Communication

2.5 Modes of Communication

2.6 Let Us Sum Up

2.7 Further Reading

2.8 Answers to Check Your Progress

2.9 Model Questions

2.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- define data communication models
- describe the different types of modes of data communication
- describe the modes of transmission

2.2 INTRODUCTION

In the previous unit you have been introduced to the concept of data communication. Signals and different types of signals like analog and digital have been covered in the previous unit. Introductory definitions of many important concepts of data communication have also been covered in the earlier unit.

In this unit you will be introduced to different data communication modes. We will also learn about the types of communication - parallel and serial communication. Synchronous and asynchronous types of communication have also been covered in this unit. In addition to these the different communication modes - simplex, half-duplex and full duplex will also be introduced in this unit.

Once you are familiar with these basic modes of data communication, you will be introduced with digital transmission fundamental concepts like modulation and multiplexing in the next unit.

2.3 DATA COMMUNICATION

Data communication means sending data or information from a source to a receiver. This data or information can be anything. This information can be a simple data file, or an audio file or video file. Data communication has become an essential part of our lives. Transferring of data from one computer to another computer is an example of data communication.

Communication usually has three important parts - a source, a destination and a communication channel. Let us look at these three components:

- Source: This is where the data originates. Generally, it is a computer but it can be any electronic device like a mobile phone, an audio recorder etc.
- Destination: This is where the data is absorbed. It can be a computer system or a television set.
- Communication Channel: This is the entity which is used to send data or signals from a source to a destination. This can be an optical fiber or a wireless medium. There are provisions of data being lost in the communication channel.
    - **Resource Sharing :** Using computer networks, it is possible to share programs, data and other resources among several users on the network. The sharing of resources is independent of the physical location of the user and the resource. Files on a particular user's computer can be shared on the network or files can be placed on a file server which provides a central location for all files needed by the users on the network. Users can also share devices such as printers, CD-ROM drives, and hard drives etc. It also makes upgrading an application easier because the upgrade only has to be performed only on the server itself. Thus,

we don't need seperate resource for each computer.

- **High Reliablity :** A second goal is to provide high reliability by having alternative sources of supply. For example, all files could be replicatied on two or three machines; so if one of them is unavailable, the other copies could be available.
- **Cost Reduction :** Resource sharing automatically reduces cost and hence money can be saved. For example, let us suppose, there are ten users and each user requires a printer. If they could have been working indivually, then ten printers would have to be purchased. If these ten users are allowed to work in a network, then only two or three printers would be sufficient.
- **Communication Medium :** Computer networks provide a powerful communication medium among the people who are in a geographically same or different location. A file that was updated or modified on a network, can be seen by the other users on the network immediately. Thus, it becomes easy for two or more peoples living far apart to work on a same project by dividing it using a network. They can write programs, can discuss or can even change or modify some data using a network while they are far off. Otherwise, they will have to wait for several days for communication through letters or some other media. Thus, it makes speedy co-operations and enhances human to human communication.



Figure 2.1: A simplified communication system

A simplified communication system can be seen in figure 2.1. The data or signal is send from the source to the destination using the communication channel. Let us now look at what could be the different types of communication.

2.4 TYPES OF COMMUNICATION

The sender sends data to the receiver through a communication channel. The data sent could be sent using one communication unit or many units. Depending on these differences, communication can be broadly divided into two types: serial communication and parallel communication. The communication can be synchronized using a clock or some other mechanisms. Depending on these, the serial communication is again subdivided into two major categories: synchronous communication and asynchronous communication.

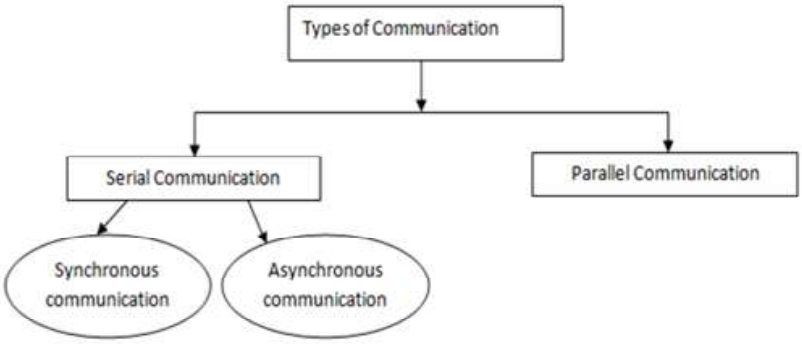


Figure 2.2: Types of Communication

Figure 2.2 shows the different types of communication. Let us now try to understand the definitions and concept of serial and parallel communications.

Serial communication is the sequential transmission of data using a single communication unit like a single copper wire or an optical fiber. In the digital domain, the data is represented as bits and can be sent over using a single electrical wire sequentially.

Parallel communication is the concurrent transmission of data using many communication units for transmission of data. In the digital

domain, multiple data bits are sent simultaneously using multiple electrical wires.

In serial communication, since there is only one communication unit so less data is sent in one transmission. Whereas in parallel communication more data can be sent since we can use multiple communication units for one transmission.

To find the suitability between serial and parallel transmission, let us take an example. Suppose we have a parallel connection with 4 wires and a serial connection with 1 wire. It would seem to us that the parallel connection could send more data than serial communication and faster than serial communication. But less data means the error in data transfer will be less as compared to parallel communication. This means that the error rates are higher in parallel communication corresponding to the increase in data.

Serial communication is preferred for sending data over longer distances due to less data error. It also has low cost and is more easy to use than parallel communication. Also this communication can be used when high data rates are not needed. Examples of serial communication are USB (Universal Serial Bus), Ethernet, Bluetooth among many. Parallel communication is used for faster data transfer since they use multiple wires and can transmit data on all the wires simultaneously. Examples of parallel communication are IDE hard disk connectors among many.

Serial communication can be again broadly divided into two categories: *Synchronous* and *Asynchronous* communication.

- **Synchronous communication:** In this mode of communication, the data bits are sent over a single communication unit, using a clock to synchronize the timing between the sending and the receiving communication devices.

  The clocks of both the sender and receiver are required to be at sync with each other in this communication. The sending and receiving time for data is already known to both parties and using

this protocol the data are being transferred. The disadvantage of synchronous communication is that eventually the clocks may get out of synch and may have to be resynchronized from time to time.

- **Asynchronous communication:** In this mode of communication, the data bits are sent over a single communication unit, and start and stop bits are used before and after the data to signify the starting and ending of the data sent.

  The data will be sandwitched between the start and stop bit in this communication. Since this communication does not use clocks, which is why it is necessary to use other means to signal the starting and ending of data to the receiver. For this purpose, the start bit is used to signal the beginning of data and the stop bit is used to signal the ending of data. For example, if the data to be sent is '0010', then it can be sent using a start bit '1' and an end bit 0' and would look like this : **'1**00101**'**. The start and stop bit are shown in bold here.



CHECK YOUR PROGRESS

Q1. Define serial and parallel communication.

Q2. Describe the difference between asynchronous and synchronous communication.

2.5 MODES OF COMMUNICATION

The manner in which information flow between two communication devices is known as the communication or transmission mode. The communication modes deal with the way traffic flows from one computer to another. In simple terms, it is the way or direction in which the flow of information takes place. This can be in any one direction or in both the directions between two systems. Depending on these differences the communication modes are basically divided into three groups. These are *simplex* and *duplex*. Further duplex communication can be *half-duplex* and full duplex.

Let us discuss these modes in detail now:

- **Simplex:** In this type of communication mode the data can travel through in only one direction. In other words, we can say that the communication is unidirectional.

In simplex mode of communication, a device can either receive or transmit information but cannot do both. For example, a television set can only receive information from the satellite channels but not transfer any information in return.

Let us take another example to understand this concept better. Suppose we need to write a paper on Indian Freedom Movement. One of the most common ways now-a-days is to collect the information from internet. For that we may use a computer and enter the information we need to search through the keyboard. The keyboard is only used to send the information to the computer but it cannot receive any information in return from the computer. This has been illustrated in the Figure 2.3.

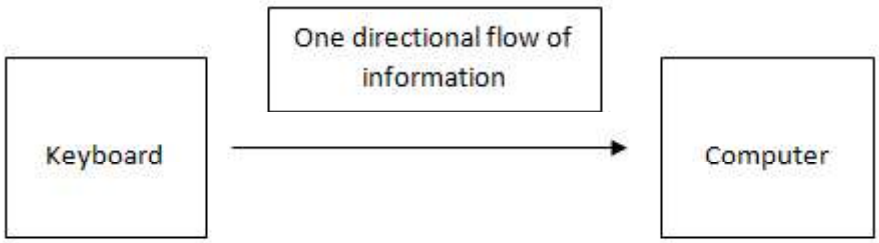


Figure 2.3 Example of Simplex communication

- **Half-Duplex:** In this type of communication mode the data can travel through in both directions but in only one direction at a time. In other words, we can say that this mode supports bidirectional transmission but can perform only one-directional transmission at a time.

In half-duplex mode of communication, a device can both receive and transmit information but cannot do both at the same time. That is, it can either receive or transmit at a particular instant of time. For example, a computer system cannot send information to another computer system while it is receiving information from that computer system. After the information has been received the computer system can now start to send

information to the other computer system.

Let us take another example to understand this concept better. We may have seen many police personnel talking in their walkie-talkie. The police officers can listen to information or send some information through their walkie-talkie. But the act of both listening and speaking at the same time using the walkie-talkie cannot be done. This is why we mostly see them either holding it against their ear or in front of their mouth while communicating with their officers and saying the word "over" when they have finished delivering the information. This has been illustrated in the figure below.

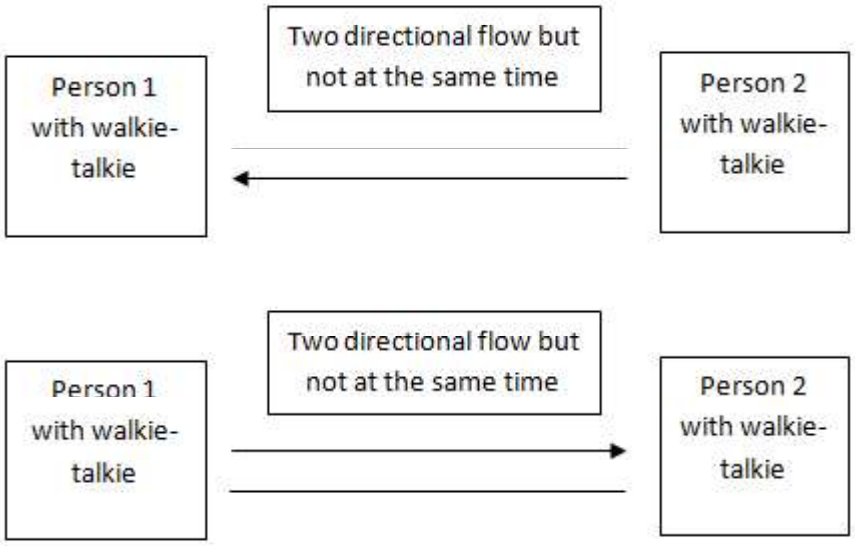


Figure 2.4 Example of half-duplex communication

- **Full-Duplex:** In this type of transmission mode the data can travel through in both directions at a time. In other words, we can say that this mode supports bidirectional communication and can perform communication in both directions at the same time.

In full-duplex mode of communication, a device can both receive and transmit information at the same time. It means that the device can receive information while sending some other information. For example, on a local area network that has full-duplex transmission, one computer system can transmit information while it is receiving data from another computer system which is also receiving information.

Let us take another example to understand this concept better. We have used the telephone for communication many times. We can both talk and listen using the telephone at the same time. Even though it may be difficult for us to hear and speak at the same time, but it is possible to use it in full-duplex transmission mode. This has been illustrated in the Figure 2.5.

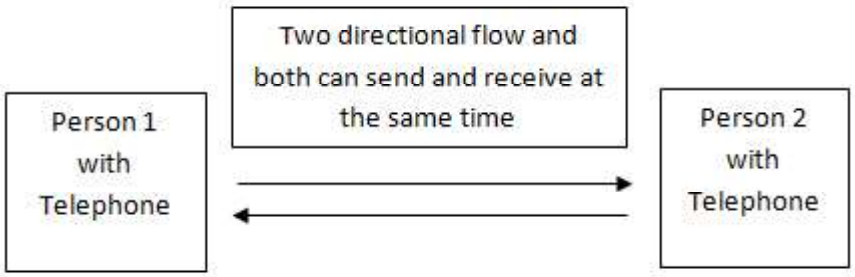


Figure 2.5: Example of full-duplex communication

The simplex communication is a one directional communication. The half-duplex is bi-directional but only one way communication at a time and full-duplex is bi-directional communication at all times. The concept of simplex, half-duplex and full-duplex communication can be illustrated further using the figure 2.6.

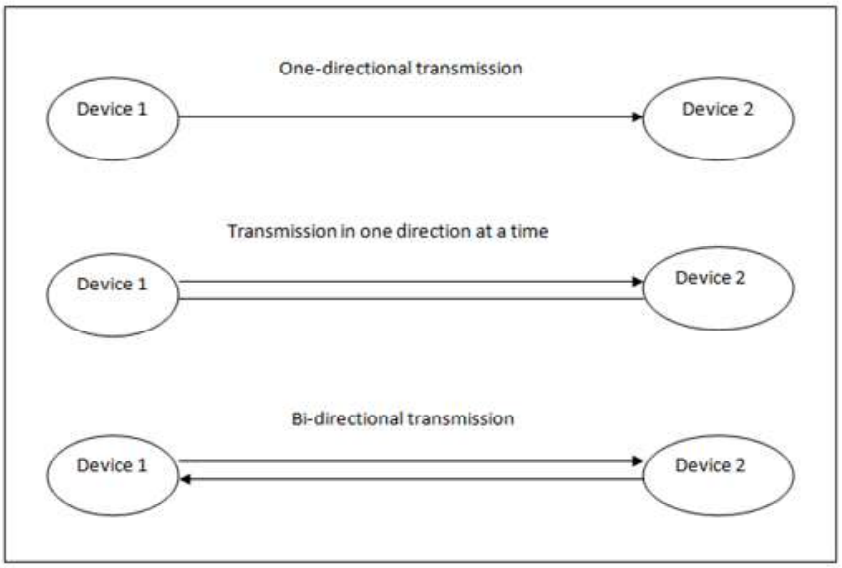


Figure 2.6: Simplex, half-duplex and full-duplex communication



CHECK YOUR PROGRESS

Q3. Define communication modes.

Q4. Describe the different types of communication modes.



2.6 LET US SUM UP

- Data communication means sending data or information from a source to a receiver.

  Communication usually has three important parts - a source, a destination and a communication channel.
- Serial communication is the sequential transmission of data using a single communication unit like a single copper wire or an optical fiber. In the digital domain, the data is represented as bits and can be sent over using a single electrical wire sequentially.
- Parallel communication is the concurrent transmission of data using many communication units for transmission of data. In the digital domain, multiple data bits are sent simultaneously using multiple electrical wires.
- In synchronous communication, the data bits are sent over a single communication unit, using a clock to make it happen at the same time concurrently. The timing between the sending and receiving devices are the controlling factors common to both ends. synchronize the timing between the sending and the receiving communication devices.
- In asynchronous communication, the data bits are sent over a single communication unit, and start and stop bits are used before and after the data to signify the starting and ending of the data sent.
- There are two broad types of communication modes. They are: simplex and duplex, Duplex modes may be half-duplex and full-duplex.

- In simplex communication mode, the data travel through in only one direction.
- In half-duplex communication mode, the data travel through in both directions but in only one direction at a time.
- In full-duplex communication mode, the data travel through in both directions at a time.



2.7 FURTHER READING

- "Computer Networks", Andrew S. Tanenbaum, Prentice Hall India.
- "Data and Computer Communications", William Stallings, Pearson Prentice Hall.



2.8 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: Data communication means sending data or information from a source to a receiver. This data or information can be anything. It can be a data file, or audio or video files. Communication usually has three important parts - a source, a destination and a communication channel.

Ans. to Q. No. 2: Depending on whether data sent could be sent using one communication unit or many units, communication can be broadly divided into two types: serial communication and parallel communication. The communication can be synchronized using a clock or some other mechanisms. Depending on these, the serial communication is again subdivided into two major categories: synchronous communication and asynchronous communication.

Ans. to Q. No. 3: The direction of flow of information between two communication devices is known as communication or transmission mode. The communication modes deal with the way traffic flows from one computer to another. In simple terms, it is the way or direction in which the flow of information takes place. This can be in any one

direction or in both the directions between two systems. Depending on these differences the communication modes are basically divided into two groups. They are: simplex and duplex. Duplex communication is further subdivided into half-duplex and full-duplex.

Ans. to Q. No. 4: There are two types of communication modes. They are: simplex and duplex. Duplex communication takes place in half and full-duplex. In simplex communication mode the data can travel through in only one direction. In other words, the communication is unidirectional. In half-duplex communication mode data can travel through in both directions but in only one direction at a time. In other words, we can say that this mode supports bidirectional communication but can perform only one-directional communication at a time. In full-duplex communication mode the data can travel through in both directions at all time. In other words, we can say that is mode supports bidirectional communication and can perform communication in both directions at the same time.



2.9 MODEL QUESTIONS

Q.1. What do you understand by data communication? Explain in detail.

Q.2. What are the different types of communication?

Q.3. Give the different modes of communication.

Q.4. What are the basic differences between half-duplex and full-duplex modes of communication?

Q.5. Give some applications where half-duplex communication is used.

Q.6. Give some applications where full-duplex communication is used.

Q.7. Give some applications where simplex communication is used.

****

UNIT 3 : NETWORK DEVICES

UNIT STRUCTURE

3.1 Learning Objectives

3.2 Introduction

3.3 Network Interface Card (NIC)

3.4 Modem

3.5 Repeater

3.6 Bridges and Gateways

3.7 Hub

3.8 Switch

3.9 Router

3.10 Let Us Sum Up

3.11 Further Reading

3.12 Answers to Check Your Progress

3.13 Model Questions

3.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- learn about different network devices and their characteristics
- differentiate bridges and gateways
- learn the functions of hub, switch and router.

3.2 INTRODUCTION

The internetworking devices are the vital tools for communication. Here, we will gain some information about the various hardware that is used to maintain connections between the network.

3.3 NETWORK INTERFACE CARD (NIC)

A ***network interface card***, also known as a ***network adapter***, is avery important hardware which is responsible for connecting a particular computer to a network. The NIC mediates between the computer(and its

user) and the network. Usually, an NIC consists of a printed circuit board containing different electronic components. It possesses a ROM chip that contains a unique number, known as the Media Access Controll (MAC) address. The MAC address identifies the device uniquely on the LAN. It also contains DMA (Direct Memory Access) circuit that allows the NIC to transmit or receive bits from memory without the involvement of the CPU. A NIC contains connector which provides plug-in facility to the network cable. It exists on the Data Link Layer of the ISO-OSI reference model. NIC handles all the details of packet transmission and reception.



Figure 3.1 : A Network Interface Card

3.4 MODEM

A modem (***mo***dulation-***dem***odulation) is a communication device that converts binary electrical signals into analog signals for transmission over telephone lines and converts these signals back into binary form at the receiving end. Conversion to analog form is called *modulation;* and the reverse process is known as *demodulation.*



Figure 3.2 : A Modem

At the transmitting end, a bits stream arrives at the modem from computer's serial port. These bits are converted into analog form (modulation) using a predefined modulation technique. This analog signal is transmitted along telephone lines. At the receiving end, the analog signal is converted back to digital form(demodulation). The resulting serial stream is sent on to the computer at the receiving end.

3.5 REPEATER

Signals can only travel so far through media before they weaken and become garbled. This weakening of signals is called *attennuation*. Repeaters are used to regenerate the analog as well as the digital signals which are distorted by transmission loss. These are Layer 1(Physical layer) internetworking devices used to combat attenuation. Repeaters take in weakened signals, clean them up, regenerates them, and send them on their way along the network. By using repeaters, the distance over which a network can operate is extended.

For example, 10Base-T (a wiring standard) is allowed to run 100 meters. One repeater can double this distance to 200 meters, as depicted in Figure 3.3

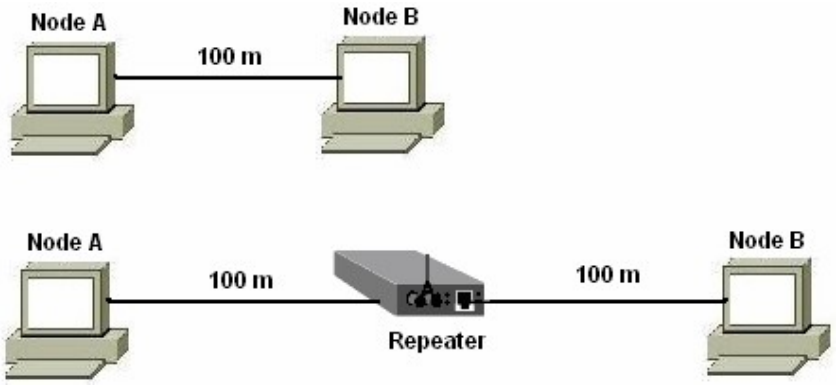


Figure 3.3 : A repeater extending network without attenuation

3.6 BRIDGES AND GATEWAYS

As the size of local area networks grew, the management of the networks became more difficult. Controlling traffic flow, maintaining fault monitoring, and adding and deleting stations from the system required more and more sophisticated system management. An alternate method to handling larger networks is to divide them into smaller networks that are interconnected using hardware/software systems called *bridges* and *gateways*.

Bridges are mainly used to connect multiple LANs which operate in the Data Link Layer to construct a larger LAN. The type of data transferred between the LANs must have the same format. A bridge can regenerate a signal it receives like a repeater. The basic function of a bridge is to transfer the frames from one LAN to another LAN. Bridge also has the capacity of frame filtering. It examines the header of each incoming frame and based on this information bridge decides whether to discard or forward this frame. Bridge stores the MAC address of received frame, keeps record on which port it has received, and which address belongs to which computer on the LAN.



Figure 3.4 : A typical bridge

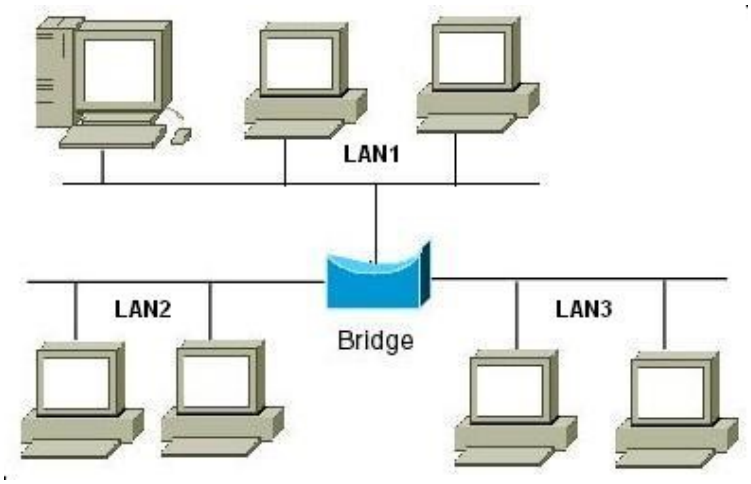


Figure 3.4 : A bridge connecting 3 LANs

Gateways are similar in functions to bridges in that they connect more than one networks. The main difference between bridges and gateways is that the latter can interconnect networks with differing message formats. A bridge does not perform any protocol translation in transferring packets between networks. whereas gateway takes an application message, reads it and interprets it. Hence, it can be used as a connecting device between two inter networks that use different model of references. One of them may use OSI model whereas the other uses TCP/IP model. A gateway is capable of receiving a frame coming from one model, makes necessary conversions and sends it to the other system of model. Thus, gateway relays packets among networks that have different protocols (e.g., between a LAN and a WAN). A gateway can provide security also.

3.7 HUB

Hubs are Physical layer(Layer 1) devices. Data that comes in one port is sent out to all other ports, except for the port it came in on. Hub does not read data passing through it and does not know the source and destination points. It simply receives the data packets, amplifes the electrical signal and then retransmit data packets on the network. Hubs usually accomodate four or eight nodes, and many hubs include connectors for linking to other hubs. A hub connects nodes that have a common network architecture.

The types of hubs that exist in the network are: *Passive hub* and *active hub*. *Passive hub* receives data packet and retransmits them on the network without amplifying the electrical signal. A multiport repeater is called an *active hub*. They receives the data packets, amplify them and retransmit on the network. For amplification they need the electrical signal.



Figure 3.5 : A hub

3.8 SWITCH

Switches are the core devices in today's modern LANs. A network ***switch*** is a hardware device that is used to connect computers that require high bandwidth and is also used to connect hubs to form large network. Switches are more expensive but provide better performance to hubs. Unlike hubs, network switches are capable of inspecting data packets as they are received, determining the source and destination device of each packet, and forwarding them appropriately. By delivering messages only to the connected device intended, a network switch conserves network bandwidth and offers generally better performance than a hub.



Figure 3.6 : A Switch

Two types of switches exist, which are :

Layer- 2 switches : These types of switches are based on the bridging technology. They establish connection between the ports based on the MAC address stored in the address table.

Layer-3 switches : These types of switches are based on the routing technology and are responsible for seting up connection between the ports based on the network address.

3.9 ROUTER

A router is the primary device responsible for connecting multiple networks together. The function of a router is to provide a path from a node on one network to a node on another network. The two networks may be separated by several intervening networks and possibly, by many miles. The purpose of a router is to forward data packets between networks. It examine incoming packets, chooses the best path for them through the network, and then switches them to the proper outgoing port. Routers are normally used to connect LANs and WANs in the Internet. It performs routing of packets based on their logical addresses. It has a routing table to take decision about the routes of packets. The table is constructed and updated dynamically by following some routing algorithm. Routers work at Layer 3, the Network Layer. In the Fig.1.9(b), routers connect nodes on different networks.



Figure 3.7(a) : Some images of Cisco routers

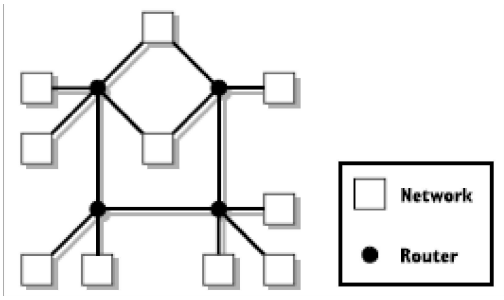


Figure 3.7 (b) : Networks connected with routers



CHECK YOUR PROGRESS

Q.1. State True or False :

i) Conversion of analog signals to digital form is called modulation; and the reverse process is known as demodulation.

ii) A hub connects nodes that have a common network architecture.

iii) Routers operates at physical layer.

iv) Broadcast networks share a single communication channel.



LET US KNOW

The terms DTE and DCE are very common in the field of networking. ***DTE*** stands for ***data terminal equipment*** and ***DCE*** stands for ***data -communication equipment***. **DTE** is an end instrument that converts user information into signals or reconverts received signals. It is used to describe the end user machine, which is usually a computer or terminal. The function of communication network is to interconnect DTEs so that they can share resources, exchange data, provide backup for each other, and allow users to perform their work from any location. The DCE is used to connect the DTEs into the communications line or channel. It also contains a portion of an application process but the primary function of the DCE remains to provide an interface to the DTE with the communication network.



3.10 LET US SUM UP

- Network devices are physical components used to connect with the computers.
- NIC is responsibile for connecting a particular computer to a network.
- A modem is responsible for converting analog signal to digital form and vice-versa.
- Repeater is responsible for regenerating analog as well as the digital signals which are destroyed by transmission loss.
- Bridges are mainly used to connect multiple LANS.
- Gateways are responsible for interconnecting networks with different message format.
- Hubs are used to connect nodes that have a common network architecture.

- Switches are very efficient network device. Unlike hub, switches are more expensive but provide better performance to hub.
- A router is the primary device responsible for connecting multiple network together.



3.11 FURTHER READING

- "Computer Networks", Andrew S. Tanenbaum, Prentice Hall India.
- "Data and Computer Communications", William Stallings, Pearson Prentice Hall.



3.12 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : i) False, ii) True, iii) True, iv) True



3.13 MODEL QUESTIONS

Q.1. What are the different types of networking devices?

Q.2. List the relative advantages and disadvantages of hub and switch.

Q.3. Mention the role of router in a network.

Q.4. Mention the main differences between bridges and gateways.

Q.5. What is Media Access Control (MAC) address?

Q.6. In which layer Repeater works? What is the role of a repeater in a computer network?

****

UNIT 4 : NETWORK TOPOLOGY

UNIT STRUCTURE

4.1 Learning Objectives

4.2 Introduction

4.3 Network Topology

4.4 Bus Topology

4.5 Ring Topology

4.6 Star Topology

4.7 Mesh Topology

4.8 Tree Topology

4.9 Let Us Sum Up

4.10 Further Reading

4.11 Answers to Check Your Progress

4.12 Model Questions

4.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- learn about various kinds of network topologies
- describe the topology structure

4.2 INTRODUCTION

In the previous unit we have learnt various internetworking devices with their uses. In this unit we will discuss how computers and other devices are connected in a network for the purpose of communication.

4.3 NETWORK TOPOLOGY

In computer networking, ***topology*** refers to the layout of connected devices. Network topologies can be *physical* or *logical*. ***Physical topology*** means the physical design of a network including the devices, location and cable installation. It defines how the systems are physically connected.

Several physical topologies are in use for networks today. Some of the common topologies include the *bus*, *ring*, *star*, *tree* and *mesh*. More complex networks can be built as *hybrids* of two or more of the above basic topologies. The ***logical topology*** defines how the systems communicate across the physical topologies. The two most common types of logical topologies are ***broadcast*** and ***token passing***.

When we decide which topology we should choose for our network, then there are few basic points that we need to take care of. The factors that decide which topology we should choose are:

- ***Cost*** : Cost is a factor that plays an important role for the decision of topology. If we want to create a network for 4 to 5 computers, we should not expense very much in network.
- ***Scalability***: What is the size of the network that we need to make and is it possible in that kind of topology
- ***Bandwidth capacity****:* The required speed of the network that can be taken care of by any particular topology.
- ***Ease of installation***: It is easy to install the network using selected topology.

4.4 BUS TOPOLOGY

A network of bus topology consists of a long single cable to which all the computers and other devices are connected. Any node attached to the bus can send signals down the cable to all the nodes of the network; that means, a bus is a broadcast medium. When more than one node start sending data through the bus, they mix with each other and the sent data becomes a garbage. This is called *collision.* To avoid collision there must be some agreement between the nodes so that when one computer starts to send data, others refrain themselves from sending data. To ensure correct data communication, both ends of the cable are terminated by a special device called end terminator.

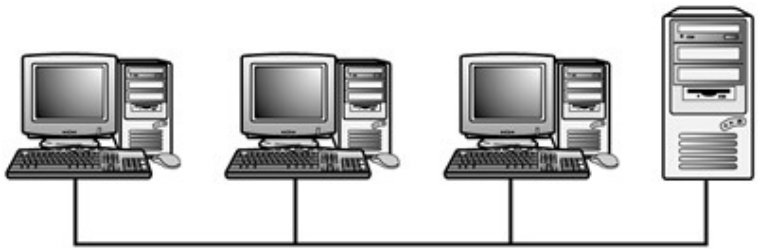


Figure 4.1(a) : A Bus Topology

Ethernet bus topologies are relatively easy to install and do not require much cabling compared to the alternatives. 10Base-2 (Thin) and 10Base-5 (Thick) both were popular Ethernet cabling options for bus topologies. Some advantages and disadvantages of the bus topology are listed below:

Advantages :

- Bus topology is inexpensive in installation
- It requires less cable than other topologies
- Good for smaller networks not requiring higher speed.
- Easy to add systems to network.

Disadvantage :

- Out-of-date technology. Bus topology was used in the early days of networking because it was inexpensive to use and relatively easy to set up.
- If the backbone cable fails, the entire network effectively becomes unusable.
- Unmanageable in a large network. If more than a few dozen computers are added to a network bus, performance problems are likely to occur.
- Difficult to troubleshoot.

4.5 RING TOPOLOGY

In ring topology the computers are connected between each other in a way that forms a close loop. In practice, a cable connects the first computer to the second computer, another cable connect the second computer to the third and so on until the last computer is connected back to

the first to complete the loop. It should be noted that the topology may not physically look like a circle. The ring means the computers are connected with each other in a logical ring. The interconnecting cables may take any shape in practice. All messages travel through a ring in the same direction (either “clockwise” or “counterclockwise”).

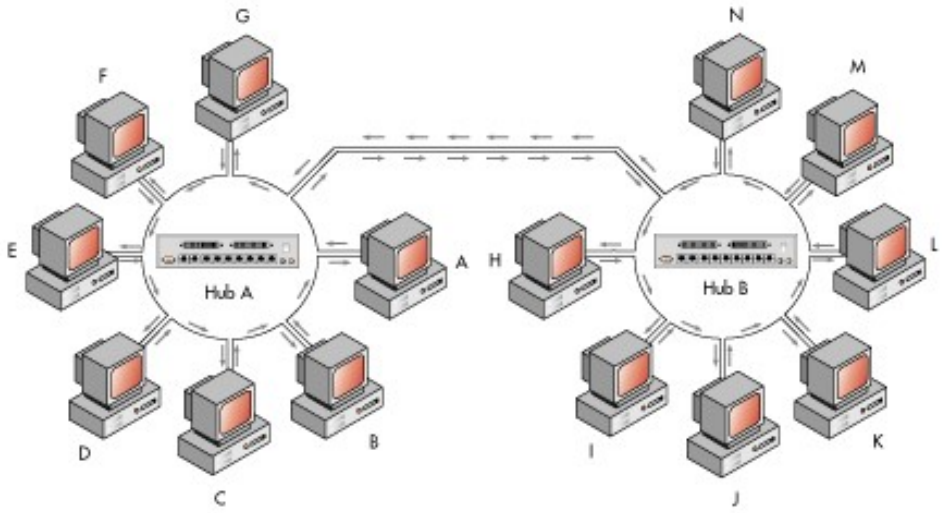


Figure 4.2 : A Dual Ring Topology

In ring topology the communicating computers must follow some agreement between them to avoid collision as it is also a broadcast network. The main difference between the bus and ring is that the ring topology does not require termination. Because the systems are connected all together in a loop, there are no beginning and end point as there is with the bus topology. A failure in any cable or device breaks the loop and can take down the entire network.

Two major network types that use the ring topology are :

i) *Fiber Distributed Data Interface* (FDDI) where a large, high speed networks use fiber optic cables in a physical ring topology.

ii) *Token-Ring networks* that use logical ring topology.

Advantages :

- No collision of data as data travel in one direction only.
- Easier to fault find. If any point gets broken we can trace that easily.
- No terminator required.

Disadvantages :

- Ring topology requires more cable than bus topology
- A break in ring will take the whole network down.
- Addition or removal of any node can affect the entire network.

4.6 STAR TOPOLOGY

In the *star* topology, all computers and other network devices connect to a central device(controller) called a ***hub*** as depicted in the Figure.4.3. Each connected device requires a single cable to be connected to the hub. A hub normally accepts data from a sending computer and delivers it to the computer for which the data is addressed. Hence, a star network is not a broadcast network, rather a point-to-point network. Using a separate cable to connect to the hub allows the network to be expanded without disruption to the network. Because each computer uses a separate cable to connect to the hub, the failure of a network connection affects only the single machine concerned. The other computers can continue to function normally.

Fast Ethernet (100Base-TX & 100Base-FX) in a star topology is the most commonly used LAN today. *10Base-T Ethernet* and *1000Base-TX Gigabit Ethernet* also use star topology.

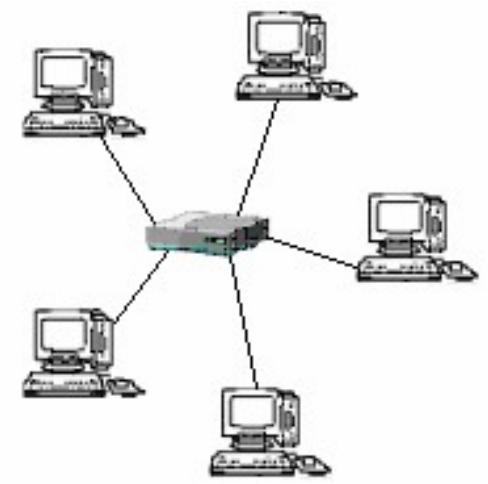


Figure 4.3 : A Star Topology

Adavntages :

- Star networks are easily expanded without disruption to the network.

- Easy to add/remove devices to/from network
- One break does not bring the whole network down. Cable failure affects only a single user.
- Easy to troubleshoot and isolate problems.
- Widely used centralized management

Disadvantages :

- Costs are usually higher than with bus or ring networks.
- Requires more cable than most of the other topologies.
- If the hub fails, any device connected to it will not be able to access the network.

In the following table, various networks with their cable types, topologies are shown:

Table 4.1 : Network cable types and topologies

Network Type	Standard	Cable Type	Topology
Ethernet	10Base-2	Thin(RG-58)	Bus
Ethernet	10Base-5	Thick(RG-59)coaxial	Bus
Ethernet	10Base-T	CAT3 or CAT5 UTP	Star
Fast Ethernet	100Base-TX	CAT5 UTP	Star
Gibabit Ethernet	1000BaseTX	CAT5, 5e or UTP	Star
Token Ring	All	UTP or STP	Logical Ring

4.7 MESH TOPOLOGY

The mesh topology incorporates a unique network design in which each computer on the network connects to every other, creating a point-to-point connection between every device on the network. A mesh topology is used when there can be absolutely no break in communications; for example, the control systems of a nuclear power plant. The purpose of the mesh topology is to provide a high level of redundancy. If a network cable, computers or other components fail, the data always has an alternative path to get to its destination.

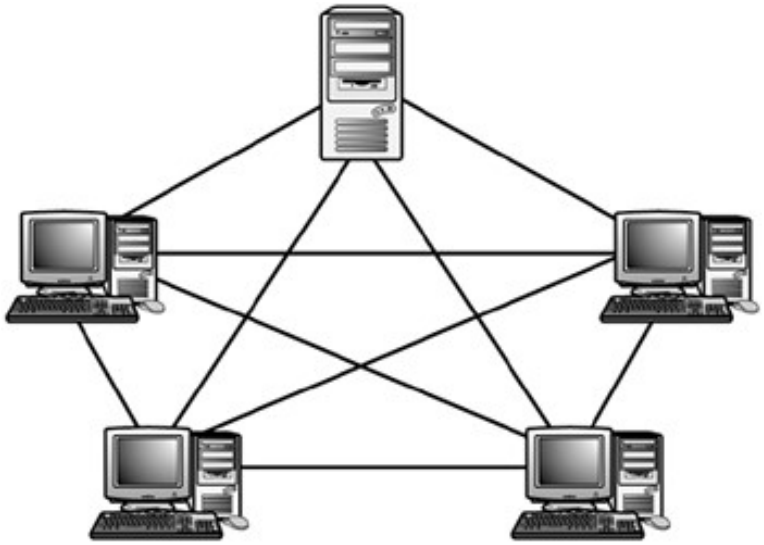


Figure 4.4 : A Mesh Topology

A fully connected mesh network has n(n-1)/2 cables to link 'n' devices. Therefore, every device on the network must have 'n-1' input/output (I/O) ports. For example, in the Figure 4.4, we have five systems that require 10 cables to create a mesh network. This topology is mainly used in environments where high availability outweighs the costs associated with this amount of interconnection. We can see in the figure that the wiring for a mesh network can be very complicated. Further, troubleshooting a failed cable can be tricky. Because of this, the mesh topology is rarely used.

Advantages :

- Provides alternative paths between devices in the network.
- The network can be expanded without disruption to current users.

Disadvantages :

- It is expensive as because a large amount of cabling are required.
- Routing network traffic can be difficult because of all the different possible paths between nodes.
- It is very expensive to wire up.

There are also partial-mesh networks where some of the nodes are connected to all the others, but others are only connected to nodes with which they exchange the most of the data.

4.8 TREE TOPOLOGY

A tree topology combines the characteristics of bus and star topologies. It consists of groups of star-configured workstations connected to a linear bus backbone cable (Figure 4.5). This bus/star hybrid approach supports future expandability of the network much better than a bus (limited in the number of devices due to the broadcast traffic it generates) or a star (limited by the number of hub connection points) alone.

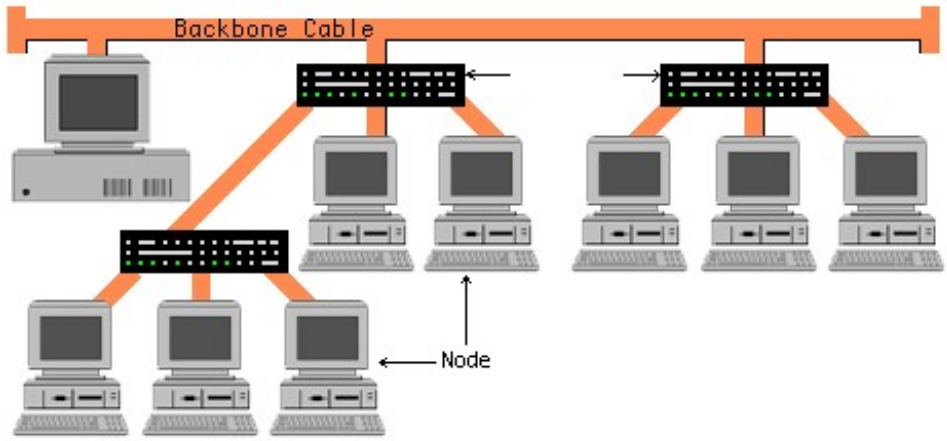


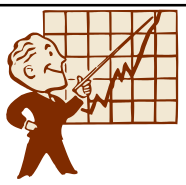
Fig. 4.5 : A Tree Topology

Advantages of Tree Topology :

- The network is easy to extend by just adding another branch.
- Fault isolation is relatively easy.

Disadvantages of Tree Topology :

- If the backbone cable breaks, the entire network goes down.
- More difficult to configure and wire than other topologies.
- If any hub goes down, all branches of that hub go down.



CHECK YOUR PROGRESS

Q.1. Choose the correct answer:

i) A network topology consists of a long single cable to is known as.

a) Star topology
b) Bus topology
c) Mesh topology
d) None of the above

ii) In which topology if the backbone cable breaks, the entire network goes down.
a) Star topology
b) Bus topology
c) Mesh topology
d) Tree topology

iii) In which topology all computers and other network devices connect to a central device(controller) called a *hub*.
a) Star topology
b) Bus topology
c) Mesh topology
d) Tree topology



4.9 LET US SUM UP

- Topology of a network refers to the way in which the network's computers and other devices that need to communicate are linked together.
- There are different types of netwrok physical topology such as Bus Ring, Star, Mess, Tree etc. In this unit we have discussed different network topologies with their merits and demerits.



4.10 FURTHER READING

- *"Computer Networks",* Andrew S. Tanenbaum, Prentice Hall India.
- *"Data and Computer Communications",* William Stallings, Pearson Prentice Hall.



4.11 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : i) Bus, ii) Tree, iii) Star



4.12 MODEL QUESTIONS

Q.1. Define network topologies. What are the goals of networks topologies?

Q.2. What is the difference between bus topology and star topology?

Q.3. What are the different types of topologies used in computer network?

Q.4. What are the advantages and disadvantages of star topology?

Q.5. List the relative advantages and disadvantages of bus and ring topologies.

****

UNIT 5 : DATA TRANSMISSION

UNIT STRUCTURE

5.1 Learning Objectives

5.2 Introduction

5.3 Parallel Transmission

5.4 Serial Transmission

5.5 Asynchronous Transmission

5.6 Synchronous Transmission

5.7 Switching Techniques

- 5.7.1 Circuit Switching
- 5.7.2 Message Switching
- 5.7.3 Packet Switching

5.8 Let Us Sum Up

5.9 Further Reading

5.10 Answers to Check Your Progress

5.11 Model Questions

5.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- define the goals of data transmission
- learn about serial and parallel transmission
- describe synchronous and asynchronous transmission
- learn about different switching techniques

5.2 INTRODUCTION

Digital transmission is the sending of information over a physical communications media in the form of digital signals. Analog signals must therefore be digitised first before being transmitted. However, as digital information cannot be sent directly in the form of 0s and 1s, it must be encoded in the form of a signal with two states, for example:

- two voltage levels with respect to earth

- the difference in voltage between two wires
- the presence/absence of current in a wire
- the presence/absence of light

Digital data transmission can occur in two basic modes:***parallel*** and ***serial.*** Data within a computer system is transmitted via parallel mode on buses with the width of the parallel bus matched to the word size of the computer system. Data between computer systems is usually transmitted in bit serial mode. Consequently, it is necessary to make a parallel-to-serial conversion at a computer interface when sending data from a computer system into a network and a serial-to-parallel conversion at a computer interface when receiving information from a network. The type of transmission mode used may also depend upon the distance and the required data rate. Now we will discuss about parallel and serial transmission.

5.3 PARALLEL TRANSMISSION

Parallel transmission communicates bits simultaneously over multiple lines(wires, frequency channels); typically, the total consists of one or more bytes at a time. Parallel devices have a wider data bus than serial devices and can therefore transfer data in words of one or more bytes at a time. As a result, there is a speedup in parallel transmission bit rate over serial transmission bit rate. The timing for parallel transmission is provided by a constant clocking signal sent over a separate wire within the parallel cable; thus parallel transmission is considered synchronous. Computers are typically connected to printers and external disk drives via parallel interfaces, ports, and buses.

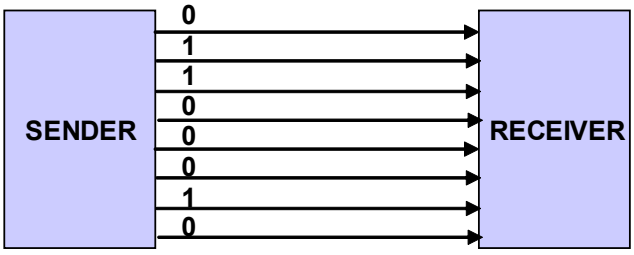


Figure 5.1 : (a) Parallel Transmission;
The eight bits are sent through eight wires at a time

5.4 SERIAL TRANSMISSION

Serial transmission sends one bit at a time over a single transmission line. The cost of communication hardware is considerably reduced since only a single wire or channel is required for the serial bit transmission which also slows the speed of transmission. Telephone lines use serial transmission for digital data. Thus, modems are connected to the computer via a serial port. A ***serial port*** is a socket on a computer used for connecting the serial interface to a serial line or bus. A ***serial interface*** is a data channel that transfers digital data serially; it is typically implemented as a cord that plugs into an expansion slot on a computer motherboard. Serial interfaces have multiple lines, but only one is used for data. An external ***serial bus*** carries serial data to any device connected to it, e.g., Ethernet. Serial transmission can be either *synchronous* or *asynchronous*.

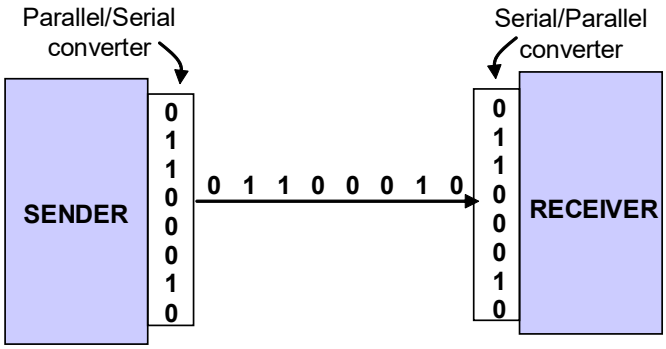


Figure 5.1 : (b) Serial Transmission;
The eight bits are sent one after another through a single wire

Serial transmission technology is increasingly used for the transmission of digital data. A large number of up-to-date communication networks apply serial transmission. The numerous applications include computer networks for office communications, building and manufacturing automation, and finally, Internet. The transmission of a stream of bits from one device to another accross a transmission media involves a great deal of cooperation and agreement between the two sides (sender and receiver). One of the most fundamental requirements is ***synchronization***. The receiver must know the rate at which bits are being received so that it can sample the medium at appropriate intervals to determine the value of each received bit. For achieving the desired synchronization, there are two approaches.

Synchronous and ***asynchronous transmissions*** are two different methods of transmission synchronization.

5.5 ASYNCHRONOUS TRANSMISSION

Communication is called *asynchronous* if the transmitter and the receiver do not need to synchronise before each transmission. A sender can wait arbitrarily long between transmissions and the receiver must be ready to receive data when it arrives. Most PC serial devices such as mouse, keyboards and modems are asynchronous.

As the name implies, asynchronous communication is performed between two or more devices which operate on independent clocks. Thus, there is no guarantee that when Point A begins transmitting, Point B will begin receiving, or that Point B will continue to sample at the rate Point A transmits. In asynchronous transmission, data are transmitted one character at a time, where each character is five to eight bits in length. Timing or synchronization must only be maintained within each character; the receiver has the opportunity to resynchronize at the beginning of each new character. The following Figure 5.2 depicts what happens when transmission clocks differ significantly. In Figure 5.2(a), we see that, the receiver samples at mid point of each bit of the incoming data. In Figure 5.2(b), the receiver clock is too slow; causing bit 4 to be skipped and as a result, the data is corrupted.To combat this type of timing problem, asynchronous communication requires additional bits to be added around actual data in order to maintain signal integrity. The bits of the character are transmitted beginning with the least significant bit.

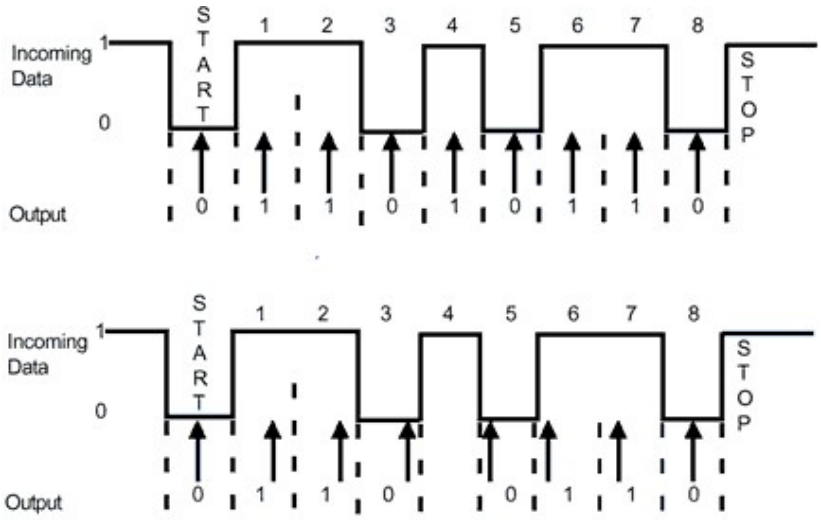


Figure 5.2 : Asynchronous Data Sampling

Asynchronously transmitted data is preceded with a ***start bit*** which indicates to the receiver that a character is about to begin. The end of a character is followed by a ***stop bit***, which tells the receiver that the character has come to an end, that it should begin looking for the next start bit, and that any bits it receives before getting the start bit should be ignored. To avoid confusion with other bits, the start bit is twice the size of any other bit in the transmission. To ensure data integrity, a ***parity bit*** is often added between the last bit of data and the stop bit. The parity bit is set by the transmitter so that the total number of ones in the character, including the parity bit, is even (even parity) or odd (odd parity), depending on the convention being used. The receiver uses this bit for error detection. The parity bit makes sure that the data to be received is composed of the same number of bits in the same order in which they were sent.

5.6 SYNCHRONOUS TRANSMISSION

The term synchronous is used to describe a data transfer method in which a continuous stream of data signals is accompanied by timing signals (generated by an electronic clock) to ensure that the transmitter and the receiver are synchronized with one another. These types of connections are used when large amounts of data must be transferred very quickly from one location to the other. The speed of the synchronous

connection is attained by transferring data in large blocks instead of individual characters. A block of bits is transmitted in a steady stream without start and stop bits. Data or information is moved from one place to another at instants in time that are measured against the clock signal being used. This signal usually comprises of one or more high frequency rectangular shaped waveforms, generated by special purpose clock circuitry. These pulsed waveforms are connected to all the devices that operate synchronously, allowing them to start and stop operations with respect to the clock waveform.

Typical examples of synchronous signals include the transfer and retrieval of address information within a computer via the use of an *address bus*. For example, when a processor places an address on the address bus, it will hold it there for a specific period of time. Within this interval, a particular device inside the computer will identify itself as the one being addressed and will acknowledge the commencement of an operation related to that address.

In synchronous transmission, each block begins with a preamble bit pattern and generally ends with a postamble bit pattern. In addition, there are some other bits that convey control information. The data with the preamble, postamble, and control information are termed as a ***frame***. A general frame format for synchronous transmission is shown in Figure 5.3. Typically, the frame starts with a preamble called a flag, which is 8 bits long.

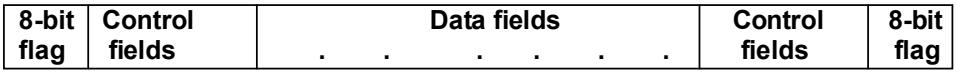


Figure 5.3 : Synchronous Frame Format

The same flag is used as a postamble. The receiver looks for the occurence of the flag pattern to signal the start of a frame. This is followed by some number of control fields, then a data field, more control fields, and finally the flag is repeated.

5.7 SWITCHING TECHNIQUES

When we have multiple work stations, a problem of connecting them for one to one communication arises. One solution for this problem is to

establish point to point connections between each individual pair of stations. To interconnect ***n*** stations, ***n(n-1)/2*** individual connections are needed. For instance, to connect 10 components, a total of individual 45 connections are required; for 100 components, 4950 connections are required. The requirement increases drastically as the number of stations to be interconnected increase, and for a large number of stations, it is almost impossible to maintain individual connections. A well known solution for such type of problems is, ***switching***. Let us think, how things would be if we could only use our telephone to talk to just one other person! So there are requirements for switching systems to route our calls around the world. The switches can be placed on the transmission path so that the stations do not need to be interconnected directly. In this section, we will briefly discuss the various switching techniques which are in use currently.

In general, there are three methods of switching techniques. These are: ***circuit switching***, ***message switching*** and ***packet switching***.

5.7.1 Circuit Switching

Circuit switching is the transmission technology that has been used since the first communication networks in the nineteenth century. The basic idea of circuit switching is to set up a dedicated logical path between two users or machines, prior to the commencement of data communication. In circuit switching, a caller must first establish a connection to a callee before any communication. During the connection establishment, resources are allocated between the caller and the callee. The most common application of circuit switching is *telephone* network. Circuit switching mechanism is divided into three phases: *circuit establishment*, *data transfer* and *circuit disconnect.*

Characteristics of Circuit Switching :

- An end to end dedicated path is needed.
- Connection path must be established before the beginning of data transmission.

- Channel capacity must be reserved between each pair of nodes whether that capacity is utilized or not.
- The technology is developed to handle voice data because of its key requirement that there should be no delay in the transmission and a constant signal transmission rate must be maintained.

Advantages of Circuit Switching :

- It is well suited for real time communication purposes.
- The service provided by circuit switching is guaranteed since the channel remains dedicated to users throughout the communication session.

Disadvantages of Circuit Switching :

- Any unused ***bandwidth*** over the allocated circuit is wasted.
- Line may be idle most of the time.
- Both sender and the receiver must function with the same speed. This limits the utility of the network in interconnecting a variety of host computers and terminals.

5.7.2 Message Switching

Message switching refers to a switching technique involving transmission of messages from node to node through a network. In this technique, the source computer sends data or the message to the switching office first, which stores the data in its buffer. It then looks for a free link to another switching office and then sends the data to this office. This process is continued until the data are delivered to the destination computers. Owing to its working principle, it is also known as *store and forward*. That is, store first (in switching office), forward later, one jump at a time.

Advantages of Message Switching :

- No waiting of the setup of path- As soon as a user has data to send, he/she may transmit it over the channel.
- The channel can be fully utilized.
- Priority of messages can be implemented.

- Broadcasting can be done very easily to all the nodes in a network.

Disadvantages of Message Switching :

- Not suitable for real time transmission
- In message switching, message size is not fixed. For very large message, the nodes are required to have large buffers, which may not be practically implemented due to economical and technical constraints.
- Since there is no size limitation of the message, long messages can keep the channel blocked for a long period of time.

5.7.3 Packet Switching

Packet switching is very similar to message switching. The main difference is that the length of the units of data that may be presented to the network is limited in a packet-switched network.

With message switching, there is no limit on block size.In contrast, packet switching places a tight upper limit on block size. A fixed size of packet which can be transmitted across the network is specified. Another point of its difference from message switching is that data packets are stored on the disk in message switching whereas in packet switching, all the packets of fixed size are stored in main memory. This improves the performance as the access time (time taken to access a data packet) is reduced and thus the throughput (measure of performance) of the network is improved.

There are two approaches to implement packet switching. These are : the datagram approach and the virtual circuit approach.



CHECK YOUR PROGRESS

Q.1. Fill in the blanks:

i) Telephone lines use _____ transmission for digital data.

(ii) A start bit and a stop bit is required in case of _____ transmission.

(iii) The three phasses of circuit switching are _____, _____ and _____.

(iv) For real time transmission _____ switching is suitable.

Q.2. What is a switched network?



5.8 LET US SUM UP

- Digital data transmission can occur in two basic modes:***parallel*** and ***serial.***
- ***synchronization means*** the receiver must know the rate at which bits are being received so that it can sample the medium at appropriate intervals to determine the value of each received bit.
- *Parallel transmission* communicates bits simultaneously over multiple lines
- ***Synchronous*** and ***asynchronous transmissions*** are two different methods of transmission synchronization.



5.9 FURTHER READING

- *"Computer Networks",* Andrew S. Tanenbaum, Prentice Hall India.
- *"Data and Computer Communications",* William Stallings, Pearson Prentice Hall.



5.10 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : (i) serial

(ii) asynchronous

(iii) circuit establishment data transfer, circuit disconnect.

(iv) circuit switching

Ans. to Q. No. 2 : A switched network consists of a series of interlinked nodes, called switches. Switches are devices capable of creating temporary connections between two or more devices linked to the switch. In a switched network, some of these nodes are connected to the end systems (computers or telephones, for example). others are used only for routing.



5.11 MODEL QUESTIONS

Q.1. Define the term digital transmission?

Q.2. What is the difference between serial and parallel transmission?

Q.3. List the relative advantages and disadvantages of serial transmission.

Q.4. Explain the concept of message switching.

Q.5. Discuss the advantages and disadvantages of message switching techniques.

Q.6. Why switching is required?

Q.7. What are the characteristics of circuit switching?

Q.8. What is packet switching?

****

UNIT 6 : NETWORK MODELS

UNIT STRUCTURE

6.1 Learning Objectives

6.2 Introduction

6.3 Design Issues of Layers

6.4 Protocol Hierarchy

6.5 ISO-OSI Reference Model : Functions of Layers

6.6 Important Terminologies

6.7 Connection-Oriented and Connectionless Service

6.8 TCP/IP Reference Model.

6.9 Comparison of ISO-OSI & TCP /IP Model

6.10 Let Us Sum Up

6.11 Further Reading

6.12 Answers to Check Your Progress

6.13 Model Questions

6.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- describe design issues of network architecture
- describe functions of various network layers
- define various network terminologies
- distinguish between connection-oriented and connection-less service
- compare between OSI and TCP/IP model

6.2 INTRODUCTION

The computer network and its structure can be best understood by studying its underlying communication software and the related hardware. Historically, first computer networks were designed to work on hardware which proved to be fatal afterwards. It leads to inflexibility to the computer networks. The later development of computer networks were based primarily

on highly structured software implemented on ever changing hardware. The communication software in a computer network is a much thought out software which is organized as layered software. The internal structure of the layered software is discussed in this unit.

6.3 DESIGN ISSUES OF LAYER

Before proceeding to learn the layered structure of network software and the various aspects of computer network, we should be acquainted with different design issues of ***layer***.

- To interchange data between two processes running on two computers there must be some addressing technique for the communication correctly.
- The communication may be unidirectional (simplex communication) or in both directions alternately one after another (half-duplex communication) or bidirectional (full-duplex communication).
- Error control is also an important issue because physical communication channels are not perfect, rather, they are error prone. Now both communicating parties must agree on which error detecting and correcting code is to be used. The receiver should also let the sender know which message is received correctly and which is not.
- During transfer, messages are often divided into pieces and all the pieces may not be transferred in proper order. This loss of sequencing has to be taken care of by the protocol concerned.
- There must be provision to refrain a fast sender from swamping a slow receiver.
- The layer concern may decide to use the same connection for several unrelated conversations by the multiplexing and de-multiplexing techniques.
- When there are many paths between the sender and the receiver, a decision must be taken as to which route is to be taken. The decision may be static or dynamic.

6.4 PROTOCOL HIERARCHY

From the design issues discussed above it is apparent that the computer network design is a complex job. To reduce the complexity of design, most of the computer networks are organized as a stack of layers, one above another, taking the service offered by the layer below. In this structure, a lower layer gives certain services to the layer just above it; at the same time it shields the details of the implementation of those services from the upper layer.

In the layered organization, a particular layer on one machine talks with a peer layer on another machine. During the conversation, they follow some rules and conventions. These rules and conventions are collectively known as **protocol**. Here we have used the term **peer** which means entities comprising the corresponding layer on different machines. So, it is the peers that talk using their own protocol.

In Fig 6.1 a five layer network is shown. Here we can see dotted arrow between peers and solid arrow between any two adjacent layers. Dotted arrow indicates that layer *n* on one machine communicates with layer *n* on the other machine. During communication they follow their protocol.

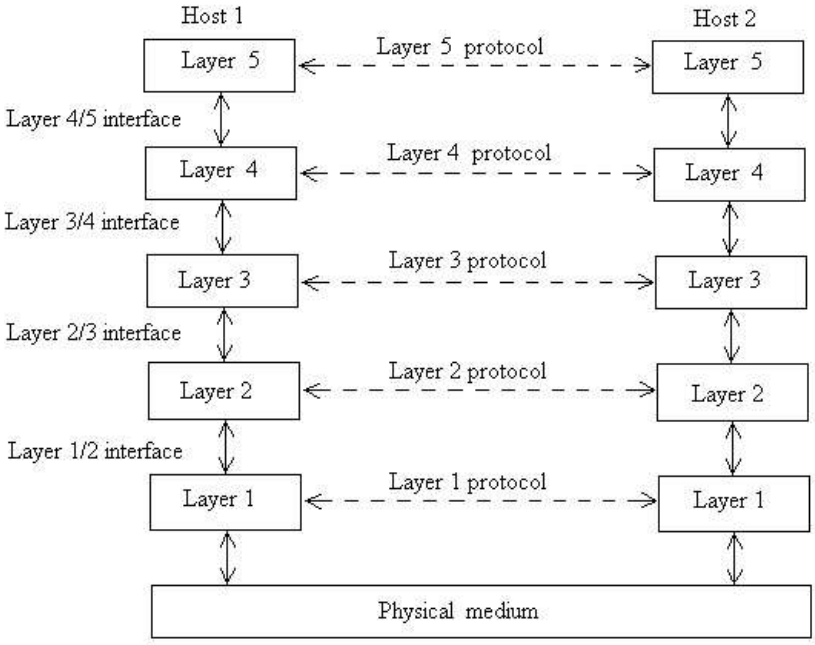


Fig. 6.1 Layers, protocols and interfaces

Every layer communicates with the peer layer on the other machine. But all these communications are virtual since no data really moves through these dotted lines; rather data moves through solid arrow lines. In the process of communication, each layer hands over their data to the layer immediately below it. In this way, data moves downward until the lowest layer, below which lies the physical medium through which the the actual communication takes place. This is the picture in the sending machine. The picture is just opposite in the receiving machine where data moves upward after it enters the computer from the physical medium.

In the Fig 6.1 each layer gets **service** from the layer just below it. A layer performs some primitive operations to give the service required by the immediate upper layer. The transaction between two adjacent layers takes place as per a predefined guideline where each layer performs a specific collection of functions. So, there exists an **interface** between each pair of adjacent layer through which they exchange data and information and in the process, the upper layer gets service from the lower layer. As long as the interface is unchanged, one can change the underlying hardware at any time without affecting the network operations. As for example, underground copper cables are gradually replaced by fiber optic cables but it is not hampering the operations of the existing network because the interfaces between the layers remain unchanged.

The layers and their protocols are collectively known as network **architecture**. In network architecture each layer has its own protocol. The protocols used by all the layers of a network system is called **protocol stack**. Stack means something arranged one above another, like stack of bricks or something similar. In a layered network architecture, layers are arranged from bottom to top, one upon another, hence their respective protocols also lie one upon another. That's why the name protocol stacks. In the same sense, *protocol hierarchy* indicates that the protocol of any layer is hierarchically above the protocol of the layer immediately below it.

6.5 ISO-OSI REFERENCE MODEL

The OSI reference model is shown in Fig 6.2. The International Standard Organization (ISO) developed a proposal for a network model and the resulting model is known as ISO – OSI reference model. This model has seven layers and the layers are :

i) Physical layer
ii) Data link layer
iii) Network layer
iv) Transport layer
v) Session layer
vi) Presentation layer
vii) Application layer

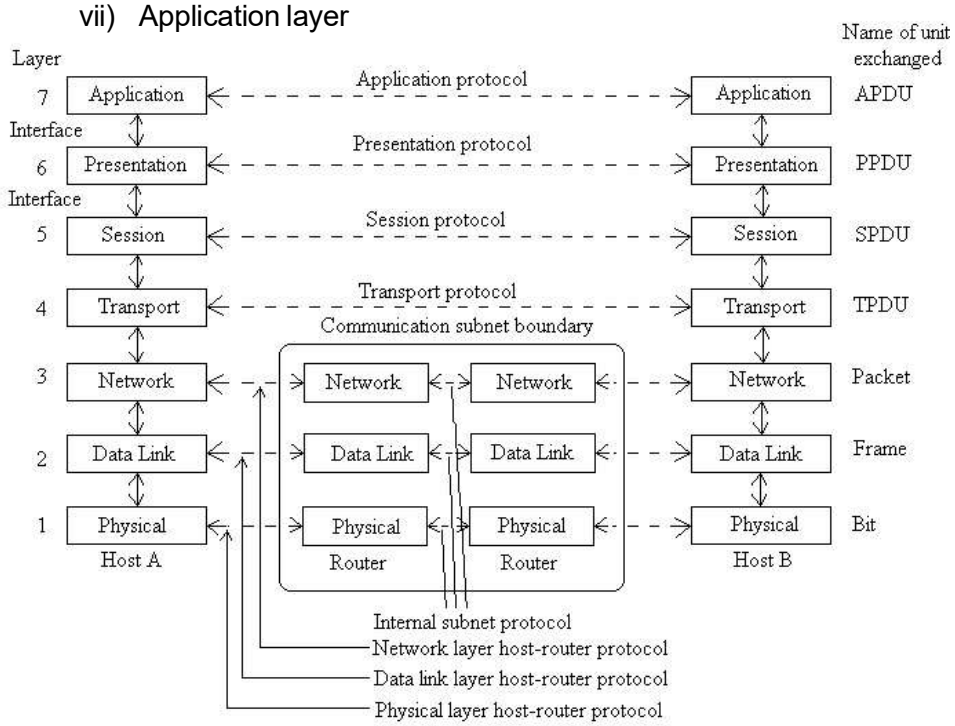


Fig. 6.2 : Layers, protocols and interfaces

(i) The Physical Layer : The physical layer is responsible for transmitting raw bits over the communication channel. This layer is to ensure how to send a 1 bit from the sending computer as 1 bit to the receiving computer; not as 0 bit. The physical layer also deals with the issues of how many bits per second will be transmitted, what level of voltage will be used to represent 1 and

0, whether transmission will be unidirectional or bidirectional, how the initial connection be established and terminated at the end, how many pins of network the connector has and which pin is for what etc etc. Hence, the design issues of the physical layer is mostly mechanical, electrical and procedure oriented.

(ii) The Data Link Layer : The data link layer use the raw transmission facility and transforms it to an apparently error-free facility to be used by the network layer. This layer breaks the input data into frames by inserting appropriate frame boundary, transmits the frames sequentially and processes the acknowledgment sent back by the receiving computer. If a frame is completely destroyed by noise burst, it is the duty of data link layer to retransmit it from the source machine. Data link layer also ensures that a fast sender is not allowed to swamp a slow receiver by sending data at a higher rate than it can be handled by the receiver. This is called ***flow control***. In a broadcast network it is the duty of MAC (Medium Access Control) sub-layer of data link layer to decide who will access the transmission medium at a particular time.

(iii) The Network Layer : The network layer controls the operation of the subnet. The layer is to determine how packets are routed from source to destination. The routing may be static or dynamic depending on traffic load and availability of channel.

Too many packets may cause congestion (traffic-jam) and control of such congestion is also a duty of the network layer.

The subnet operation requires cost; hence, some accounting function is also there being built into the network layer. When a packet crosses national boundary some other aspects of accounting has to be dealt with by network layer.

Packets have to travel in between heterogeneous network running on different platforms using different network protocol and the network layer is also responsible to resolve all the problems arising out of such situations.

(iv) The Transport Layer : The transport layer is to accept the data from the session layer, breaks it into smaller units if necessary, hands over these to the network layer and ensures that the pieces all delivered correctly to the receiver. The above duties must be done efficiently and in such a way that it will not affect the upper layer in case there is any change in the hardware.

Normally, the transport layer creates individual connection for each session. If high throughput is required, the transport layer may establish multiple network connections, dividing the data among individual connections, thereby improving the throughput. To reduce cost the transport layer may also multiplex several transport connections onto the network connection. However, multiple connections or multiplexing must not be seen by the session layer.

The transport layer also determines the type of service given to the user of the network. Error free point-to-point connection is the most popular transport layer service where messages or bytes are delivered in order in which they were sent. The transport layer also performs flow control.

(v) The Session Layer : This layer offers facility to different users on different computers to establish session between them. A session allows an user to remotely log into a distant machine and transfer file between the two machines. Session layer perform token management to provide unidirectional communication. It also provides a service called synchronization.

(vi) The Presentation Layer : This layer performs data presentation job by following syntax and semantics rules. Before presenting data to the user, it transforms data into their acceptable form.

(vii) The Application Layer : This layer is the nearest layer to all the network users. It offers variety of protocols that are commonly needed. It helps to transfer file. Different file systems have different meanings in different machines with different data formats etc. When files are transferred from one machine to

another with different file systems, the application layer takes the necessary steps to resolve the abnormalities.



CHECK YOUR PROGRESS

Q.1. What are *service* and interface?

.....

.....

Q.2. What do you mean by network architecture?

.....

.....

Q.3. How many layers are there in ISO-OSI model?

.....

Q.4. What are the prime duties of Data Link Layer?

.....

6.6 IMPORTANT TERMINOLOGIES

When we study the subject of computer network, we encounter many terminologies associated with the subject. Already we have come across a few terminologies such as **peer, interface, protocol, service** etc. Some others are briefly described below:

- **Entity :** The active components in each layer is called an entities. Entity can be a software process or hardware like an intelligent input/output device etc.
- **Peer Entity:** Entities in the same layer running on different machines are called peer entities.
- **Service provider and Service user:** In layered architecture, layer *n* provides a service which is used by layer *n+1*. Here layer *n* is service provider and layer *n+1* is service user.
- **SAP (Service Access Points):** A layer *n* offers services to layer *n+1* at a place which is called service access point. Each SAP has a unique address for identification.

6.7 CONNECTION-ORIENTED AND CONNECTIONLESS SERVICE

A layer can offer two different types of services to the layer immediately above it – connection-oriented and connectionless service.

In ***connection-oriented*** service, the service user atfirst establishes a connection atfirst, communicates over the connection and at last release the connection. It is similar to a telephone system.

In ***connectionless*** service, no connection is established beforehand. Instead, like in a postal system, every massage carries its full destination address and each is routed independently to its destination. If a big message is broken into pieces and sent to the same destination, then in connectionless service sometimes it may happen that the first piece may arrive at the destination after the later pieces. That means at the receiving end the order of delivery may not be same as the order of transmission. In connection-oriented service this never happens.

6.8 TCP/IP REFERENCE MODEL

TCP/IP reference model was developed much earlier than the OSI reference model. It was evolved from the US Department of Defense's (DoD) research network- The ARPANET. Eventually the ARPANET connected many universities and other government organizations. During the process of interconnecting all these through the existing telephone lines, satellite link and radio, the protocol used for the ARPANET had trouble and therefore, a new architecture was needed to overcome that. The model has four layers, namely :

1. The Host-to-Network Layer.
2. The Internet Layer.
3. The transport Layer.
4. The Application Layer.

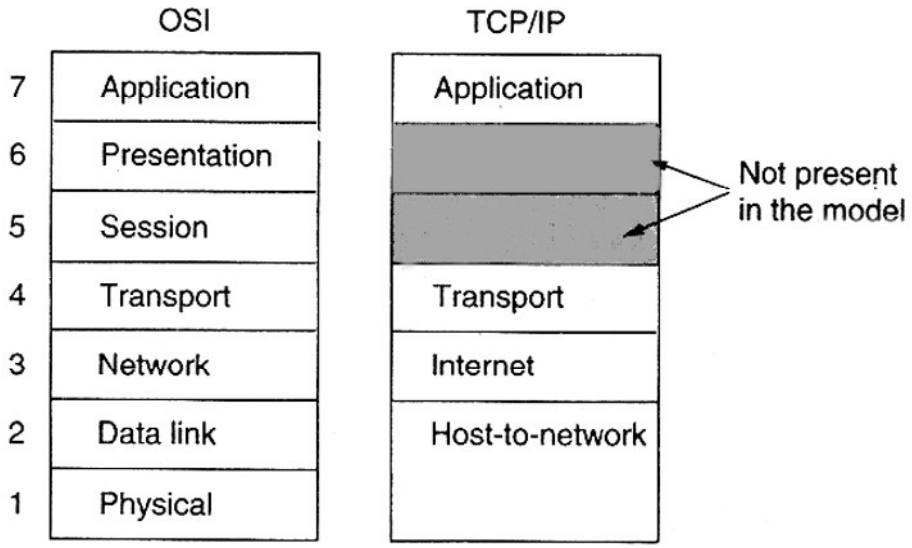


Fig. 6.3 : The TCP/IP reference model

1. **The Host-to-Network Layer :** In TCP/IP reference model the bottom layer is not defined clearly. Yet we can consider that whatever is there below the internet layer is the bottom layer in the TCP/IP model. In the model, the host has to be connected to the network using some protocol so that it can send IP packets over it.
2. **The Internet Layer :** The DoD planned to set up their inter-network in such a way that it had to survive even if a particular link failed in a probable war. So, this requirement led to connectionless packet-switching network instead of a connection-oriented circuit-switching network. The internet layer is designed to fulfil the goal of the architecture in such a way that it becomes the linchpin of the whole architecture. The job of this layer is to pump the packets from the host machine into any network and to help the packets to go to the destination independently. In this style of communication, packets may arrive in a different order than they were sent. This layer has its own protocol called **IP (Internet Protocol)** and a specified packet format. Packets routing as well as congestion avoidance are the major issues in this layer. This layer is similar to the network layer in OSI reference model.

3. **The Transport Layer :** The layer immediately above the internet layer is called the transport layer. This layer communicates with the peer layer on the other machine. It has two protocols namely **TCP and UDP.**

   (i)TCP: TCP is a reliable connection-oriented protocol that ensures delivery of byte stream from source machine to destination machine. This layer breaks the byte stream received from the upper layer into messages and passes them to the internet layer. At the receiving machine, these messages are reassembled by the TCP which passes them to the upper layer. Flow control is also another job of this layer to restrict a fast sender from swamping a slow receiver.

   (ii) UDP: UDP is another protocol used by this layer which offers an unreliable, connectionless service. Here, unlike TCP, sequencing and flow control is not done. When prompt delivery is more important than accurate delivery, UDP is used. It is also widely used in one time client-server type communication. Email, video or sound transmissions are some of these applications.

4. **The Application Layer :** In OSI model, there are session and presentation layers above the transport layer. In TCP/IP model these two layers are absent. So, on top of the transport layer, the application layer is present in TCP/IP reference model. All the high level protocols are present here. TELNET, FTP, SMTP are some of the early inclusions. Later, other protocols such as DNS, NNTP, HTTP are added to this layer.

6.9 COMPARISON OF ISO-OSI & TCP /IP REFERENCE MODEL

There is not much difference between the OSI and TCP/IP reference model. In both, the concept of independent stack of protocols is used. The functions of the layers are also more or less same. In both models, starting from bottom up to the transport layer, the function of the layers is to provide

end-to-end network independent transport service for communicating processes. The layers above the transport layer are application oriented.

On the other hand, there are some differences between the two models. In OSI reference model, three concepts are distinct. These are -

1. Services,
2. Interfaces,
3. Protocols.

The *Service* tells what a particular layer serves the layer just above it, not how these services are accessed by the above layer or how these are provided by the layer. The *Interface* tells how the above layer accesses the services provided by the layer just below it. It also does not say how these are provided. The *Protocols* used between peer layers are the set of rules agreed upon by both the layers to get the job done. Protocols can be changed without affecting the software in the higher layers.

In TCP/IP model, there is no such distinction between *Services, Interface and Protocol*. Therefore, in TCP/IP reference model the protocols are not so hidden as in the OSI reference model. As the technology advances, protocol between a peer layer can be replaced easily in OSI model than in TCP/IP model.

In OSI model, the layered structure was thought out before the protocols were invented. The designers were new in the network technology and hence in some layer a sub-layer has to be provided later to accommodate some new mode of communication. For example, in data link layer, the MAC sub-layer was introduced later when broadcast communication came to deal with channel allocation issues. On the other hand, in TCP/IP model, the protocols came first and then the model. So there was no problem of fitting the protocols into the model.

The two models also differ in their number of layers; In OSI model there are seven layers whethas TCP/IP model has four layers. Both have (inter)network, transport and application layers in common but the other layers are not the same.

In OSI model, network layer supports both *connection-oriented* and *connectionless communication* and the transport layer supports only

connection-oriented communication. On the contrary, in TCP/IP model the network layer supports only *connectionless* communication but the transport layer supports both the modes i.e. *connectionless* and *connection-oriented* communication.



CHECK YOUR PROGRESS

Q.5. What are the two services that can be offered by a layer to the layer above it?

.....

Q.6. How many layers are there in TCP/IP reference model?

.....

Q.7. Which model of network architecture was developed earlier – OSI or TCP/IP?

.....

Q.8. Which two layers of OSI model are absent in TCP/IP model?

.....



6.10 LET US SUM UP

- There must be some addressing technique for smooth communication between two computers.
- Simplex communication means one directional communication.
- In half-duplex communication, communication is bidirectional alternatively; not at the same time.
- Full-duplex communication means bidirectional communication at all time.
- Most of the computer networks are organized as a stack of layers, one above another, taking the service offered by the layer below.
- Protocol means a set of rules followed by two peers during communication.

- Entity: The active components in each layer are called entities. Entity can be a software process or hardware like an intelligent input/output device etc.
- Peer Entity: Entities in the same layer running on different machines are called peer entities.
- Service provider and Service user: In layered architecture, layer *n* provides a service which is used by layer *n+1*. Here layer *n* is service provider and layer *n+1* is service user.
- SAP (Service Access Points): A layer *n* offers services to layer *n+1* at a place which is called service access points. Each SAP has a unique address for identification.
- The layers and their protocols are collectively known as network architecture.
- OSI model is a seven layer network architecture.
- A layer can offer two different types of services to the layer immediately above it – connection-oriented and connectionless service.
- In TCP/IP model of network architecture, there are four layers.
- Most of the computer networks are based on TCP/IP model.



6.11 FURTHER READING

- Tanenbaum, Andrew, *Computer Networks*, PHI.
- Forouzan Behrouz A., *Tata Mcgraw Hill*.
- Norton Peter, *Complete Guide to Networking*.
- Comer, E. Douglas, Narayanan, M. S., *Computer Networks and Internets with Internet Applications*, Pearson Education.



6.12 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : A layer performs some primitive operations to fulfill the requirement of the layer immediately above it. This responsibility of the lower layer is called service.

On the other hand interface is the junction between two adjacent layers across which the lower layer provides the services to the layer above it.

Ans. to Q. No. 2 : The layers and their protocols are collectively known as network **architecture**.

Ans. to Q. No. 3 : Seven layers.

Ans. to Q. No. 4 : Framing, flow control , error control, and access control.

Ans. to Q. No. 5 : Connection-oriented service and connection-less service.

Ans. to Q. No. 6 : Four layers.

Ans. to Q. No. 7 : TCP/IP reference model was developed earlier than OSI model.

Ans. to Q. No. 8 : Session layer and presentation layer.



6.13 MODEL QUESTIONS

Q.1. What are the design issues of layered architecture of a computer network?

Q.2. What is the principal difference between connection-less and connection-oriented communication ?

Q.3. Describe ISO-OSI reference model of computer network.

Q.4. Define : peer entity, protocol, protocol stack, SAP.

Q.5. Which of the OSI layers handles each of the following:

a) Breaking the transmitted bit stream into frames.

b) Determining the route through which subnet is to be used.

Q.6. Describe the functions of various layers of the TCP/IP model.

Q.7. What are the responsibilities of the network layer in the TCP/IP model?

Q.8. What are the responsibilities of the transport layer in the TCP/IP model ?

Q.9. How do the layers of the TCP/IP model correlate to the layers of the OSI model ?

Q.10. Match the following to one or more layers of the OSI model:

a. Flow control

b. Route determination

c. Provides access for the end user

d. Interface to the transmission media.

****

UNIT 7 : TRANSMISSION MEDIA

UNIT STRUCTURE

7.1 Learning Objectives

7.2 Introduction

7.3 Transmission Medium

7.4 Guided Media

- 7.4.1 Coaxial Cable
- 7.4.2 Twisted Pair Cable
- 7.4.3 Fiber Optics Cable

7.5 Unguided Media

- 7.5.1 Radio Waves
- 7.5.2 Infrared
- 7.5.3 Microwave
- 7.5.4 Satellite

7.6 Wireless LANs (IEEE 802.11)

7.7 Let Us Sum Up

7.8 Further Reading

7.9 Answers to Check Your Progress

7.10 Model Questions

7.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- learn about different classes of transmission media
- describe the twisted pair, coaxial and fiber optics cables
- elaborate the radio waves, infrared, microwaves
- learn the concept of Wireless LANs

7.2 INTRODUCTION

So far, you must have been acquainted with the basic terminology of computer networks, concept of layers as well as ISO-OSI and TCP/IP

model. In this unit, we will concentrate on the transmission media and its various types.

7.3 TRANSMISSION MEDIUM

Transmission medium is the physical path that carries data from transmitter to receiver. The signals are transmitted from one device to another device in the form of electromagnetic energy with the help of transmission medium. The transmission medium can be divided into two classes : **Guided** (wired) media and **unguided** (wireless) media. Guided media includes twisted pair cables, coaxial cables and fiber optics cables. For guided media, the electromagnetic waves travel along a solid medium, such as different types of cables. For unguided media, the medium is wireless and occurs through air, space or water.

7.4 GUIDED MEDIA

Guided transmission media uses a cabling system that guides the data signals along a specific path. The data signals are bounded by the cabling system. Guided media is also known as bound media. “Cabling” is meant in a generic sense and is not meant to be interpreted as copper wire cabling only. The followings are the basic guided media used in networking :

- Coaxial Cable
- Twisted-pair cable
- Fiber optic cable

7.4.1 Coaxial Cable

The first version of Ethernet was based on coaxial cable. The original form of Ethernet 10Base-5 used a thick coaxial cable (called thick coax) that was not directly connected to the computer’s NIC (Network Interface Card). An attachment unit Interface (AUI) ran from a DB15 connector on the back side of the NIC to the thick coax. A “vampire tap” was inserted into the thick coax to which the AUI is connected.

Another version of coaxial cables are there which are thinner in size and hence its name is thin coaxial cable (called thin coax). 10Base-2 Ethernet cards are connected to this type of cable by a BNC (Bayonet-Neill-Concilman) –T connector on the rear side.

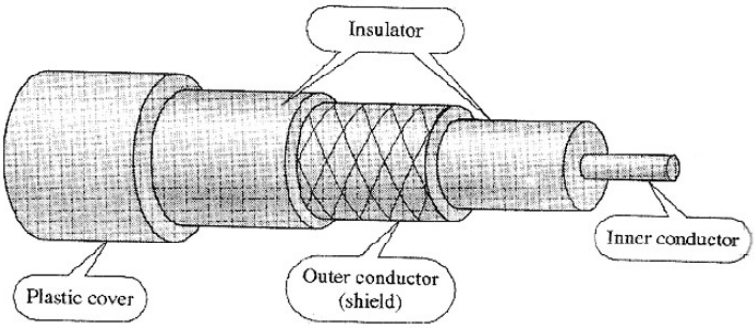


Fig. 7.1 : Construction of a coaxial cable

The physical construction of a coaxial cable can be seen in the Fig 7.2. We can see that the cable has a central core made of solid or stranded copper which carries the signals. Upon the central conductor there is an insulating layer upon which a braided wire mesh is provided as a shield to protect the cable from external noise. Upon this shield another insulating sheath is there enclosing which a plastic cover is provided to complete the construction of the cable.

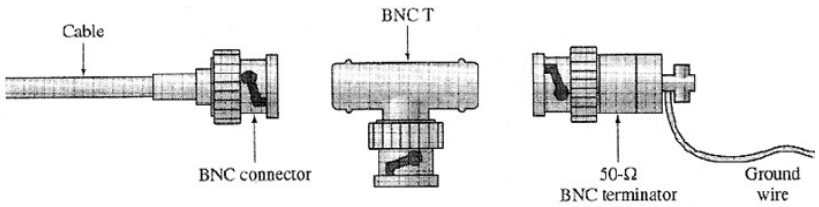


Fig. 7.2 : BNC and BNC-T Connector

Thick and Thin coax differs in size and cable impedance. Coaxial cables are categorized by their Radio Government (**RG**) ratings. Each RG number denotes a set of physical specifications uniquely.

Category	Impedance	Use
RG – 59	75 Ω	Cable TV
RG - 58	75 Ω	Thin Ethernet
RG - 11	50 Ω	Thick Ethernet

Table 7.1 : Categories of Coaxial Cables

Applications :

Coaxial cable was extensively used in analog telephone networks to carry 10,000 voice signals at a time. Later these cables were used in digital telephone networks which could carry digital data up to 600 Mbps. But coaxial cables are almost replaced today by fiber optic cables.

Cable TV networks are still running on coaxial cables in our country. However, in some advanced countries these are replaced by fiber optic cables.

Another most common application of coaxial cable is in traditional Ethernet LAN, which are also gradually replaced by twisted-pair cables.

Advantages :

- **Strong construction :** Constructional features make a coaxial cable strong in structure.
- **Good immunity to noise :** Coaxial cables have good immunity to external noise.
- **Good LAN length :** LAN implemented by using coaxial cables are of moderate length; it is around 500 meter for thick coax and 200 meter for thin coax.

Disadvantages :

- **Heavy weight :** Coaxial cables are heavy in weight. Therefore coaxial cables need strong support.
- **Unreliable tap :** When thick coax is connected by vampire tap, it provides an unreliable connection because with slight shaking the connection becomes loose.
- **High cost of installation :** Due to heavy weight laying of coaxial cables causes high labour cost.

7.4.2 Twisted Pair Cable

Twisted-pair cable is just what its name implies; insulated wires housed within a protective casing with a specified number of twist

per foot of distance. Twisting of wires reduces the effect of electromagnetic interference which may be generated from nearby electric motors, fluorescent tube lights etc. Shielded twisted-pair (STP) cables are provided with shielding and additional insulation over the cluster of wires to protect it from outside noise. Unshielded twisted-pair (UTP) cables are mostly used in small office or home LANs which are less costly. Fig 7.3 shows UTP and STP cable.

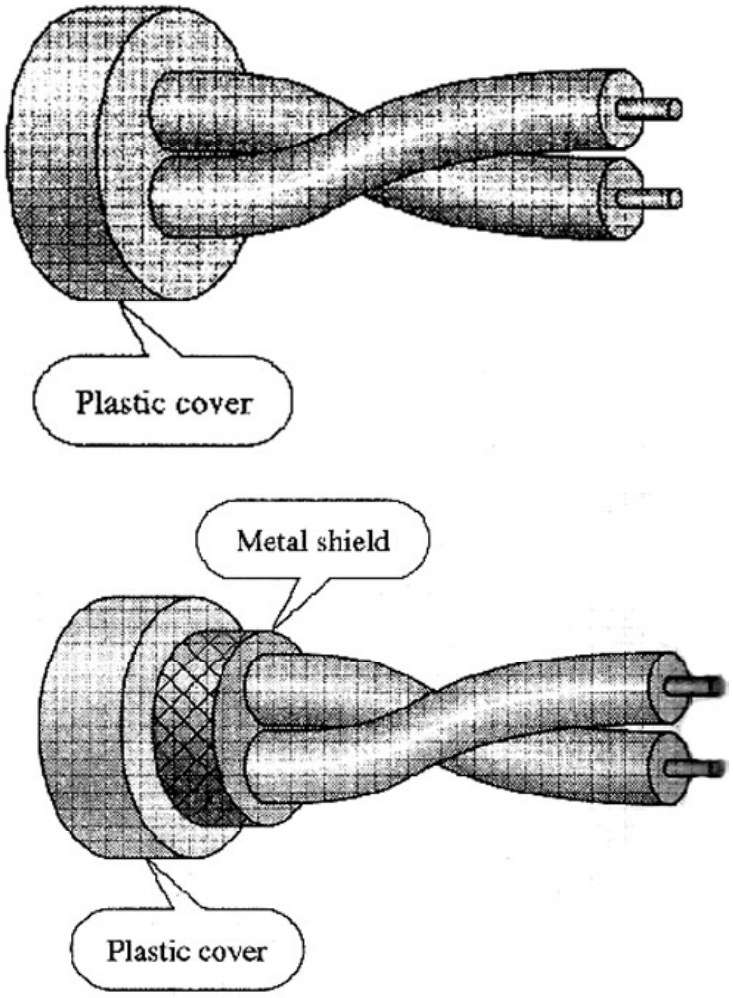


Fig. 7.3 : Twisted pair cable

The unshielded twisted-pairs are classified by The Electronic Industries Association (EIA) into seven categories. The categories are determined by cable quality. 1 is the lowest and 7 is the highest in quality. Table 7.2 shows the different categories of UTP cables.

Cate-gory	Specification	Data Rate (Mbps)	Use
1	Unshielded twisted-pair used in telephone	< 0.1	Telephone
2	Unshielded twisted-pair originally used in T-lines	2	T-1 lines
3	Improved CAT-2 used in LAN	10	LANs
4	Improved CAT-3 used in Token-Ring Networks	20	LANs
5	CAT-5 Cable wire with a jacket and outside sheath	100	LANs
5e	Extension of CAT-5, can minimize cross-talk & interference	125	LANs
6	CAT-6 cable with other matched components	200	LANs
7	Shielded screen twisted-pair (SSTP)	600	LANs

Table 7.2 : Categories of UTP cables

The most common UTP connector is RJ 45 (Registered Jack 45) which can be inserted in one way only.

Applications :

- Twisted-pair cables are extensively used in telephone lines as voice and data channels. The lines connecting subscribers to the telephone exchange are unshielded twisted-pair cables.
- People use twisted-pair cables to connect DSL lines for high speed Internet Connection.
- LAN such as 10Base-T and 100Base-T uses twisted-pair cables.

Advantages :

- **Low cost :** The cost of twisted-pair is not so high.
- **Light weight** : Twisted-pair cables are lighter in weight. Hence it does not require heavy support.
- **Easy installation & maintenance:** The technology of twisted-pair cable laying is a job of almost any person. Connecting the RJ45 connector to the cable is called crimping and it requires a few hours of practice.
- **Cheaper equipment:** Crimping tools and other tools necessary for twisted-pair cable preparation are cheap. Hence these are easy to purchase.

Disadvantages :

- **Susceptible to cross talk:** Twisted-pair cables are not totally immune to cross talk.
- **Distance limitation :** The maximum length of LAN using twisted-pair cable is short, in comparison to optical fiber.

7.4.3 Fiber Optics Cable

Fiber optic cable is a communication medium made of glass or plastic that can transmit signals in the form of light. In optics there is a principle which states that when a beam of light travels from denser medium to rarer medium then it deviates away from the normal drawn at the point of incidence at the junction of the two media. It can be seen from the Fig 7.4.

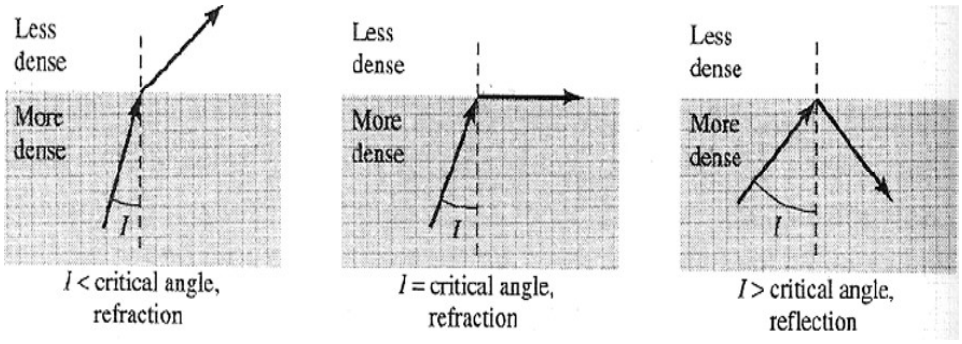


Fig. 7.4 : Bending of Light Ray

If the angle of incidence is such that the angle of refraction is 90^0 then the angle of incidence is called **critical angle**. Now if the angle of incidence is greater than critical angle then the refracted ray would re-enter into the denser medium. This phenomenon is called **total internal reflection.**

In optical fiber light signals travel by following the principle of total internal reflection. In an optical fiber, a glass or plastic **core** is surrounded by a **cladding** of less dense glass or plastic. This difference in density facilitates total internal reflection for a light signal that travels through the cable. Fig 3.5 shows the propagation of light through an optical fiber.

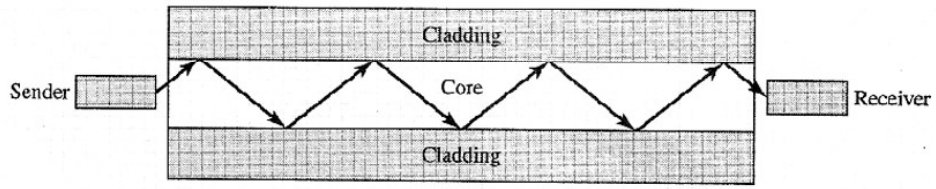


Fig. 7.5 : Light propagation through optical fiber

Current technology provides two modes of light propagation through an optical fiber. These two modes are: **mono mode** and **multi mode.** Further, fiber optic cable has two varieties: **step index** and **graded index**.

In step index fiber, core and cladding has a definite boundary. In graded index fiber, density of the fiber gradually decreases from inner core to the outer boundary.

In ***mono mode***, step index fiber is used. Mono mode fibers are of the order of 8 – 10 microns in diameter. Mono mode fibers are costly and they can transmit light signal to a longer distance than multi mode fiber. A highly focused beam of light is injected into the fiber at an incident angle close to the horizontal line. Its density is also much lower so that the critical angle is almost equal to 90^0. Therefore, the reflected beam also propagates in almost horizontal line. The delay is minimum here.

Multi mode operation can transmit multiple beams of light signal at a time. These fibers are thicker in diameter than the mono mode fiber, of the order of 50 microns. These are cheaper but can transmit signal to a shorter distance. In multi mode both step index and graded index cables are used.

Fiber optic cables are connected to NIC by three types of connectors: **Subscriber channel (SC) connector, Straight-tip (ST) connector** and **MT-RJ connector.**

Applications : Fiber optic cable is now extensively used as a communication medium for backbone networks due to its low cost wide bandwidth. Its data transfer rate now arrived at 1600 Gbps. Some cable TV companies use a combination of optical fiber and coaxial cable. Here optical fiber provides the backbone network and

the coaxial cable gives service connection to the users. 100Base-FX (Fast Ethernet) and 1000Base-X (Gigabit Ethernet) LAN uses optical fiber cable.

Advantages and Disadvantages of Optical Fiber :

Advantages : Fiber optic cables have several advantages over the metallic cables such as coaxial cable and twisted-pair.

- **Higher bandwidth :** Fiber optic cable can provide very high bandwidth and, therefore, high, data rate than the coaxial and twisted-pair cable.
- **Less attenuation :** It offers less attenuation than metallic cables. Hence, signal can travel up to 50 Km without repeater in a fiber optic cable.
- **Immunity to electromagnetic interference :** Electromagnetic noise cannot affect a light signal traveling through a fiber optic cable.
- **Resistance to corrosion :** Fiber optic cables are not affected by corrosive substances.
- **Light weight :** These are much lighter in weight than metallic cables.
- **Greater immunity to tapping :** Light signals are more difficult to tap than an electrical signal. Hence, the signal passing through a fiber optic cable has greater immunity to tapping.

Disadvantages : Despite many advantages, fiber optic cable has some disadvantages also. These are:

- **Difficult to instal and maintain :** Since fiber optic cable is a new technology, so skills personal for installation and maintenance of the fiber optic cable are not easy to find.
- **Unidirectional signal propagation :** Light propagation is unidirectional. So, if we require bidirectional communication, we need two fiber cables.
- **Cost :** The fiber optic cables and the interfaces are more expensive till date in comparison to other types of cables.



CHECK YOUR PROGRESS

Q.1. What are the different types of cables used in LAN?

.....

.....

Q.2. How can a thick coax be connected to the AUI?

.....

Q.3. What is the impedance of a coaxial cable used in cable TV?

.....

Q.4. What is the maximum data rate of SSTP cable?

.....

Q.5. Why is twisting done in twisted-pair cables?

.....

.....

Q.6. On what principle does light travels in fiber optic cable?

.....

7.5 UNGUIDED MEDIA

In wireless transmission, the data signals travel without any guides media. The data signals are not bound to a cabling media and are therefore often called unbound media. The transmission and reception are achieved by means of an antenna. In two way communication, the same antenna can be used for both transmission and reception.

For wireless communication, signals can travel from source to destination in several ways. They are - ***ground propagation, sky propagation*** and ***line-of-sight propogation***.

Radio waves travel through the lowest portion of the atmosphere. These low frequency signal (which is about 2 MHz) emanate in all directions from the transmitting antenna and follow the curvature of the planet. The distance depends on the amount of power in the signal. This type of propagation is called **ground propagation**.

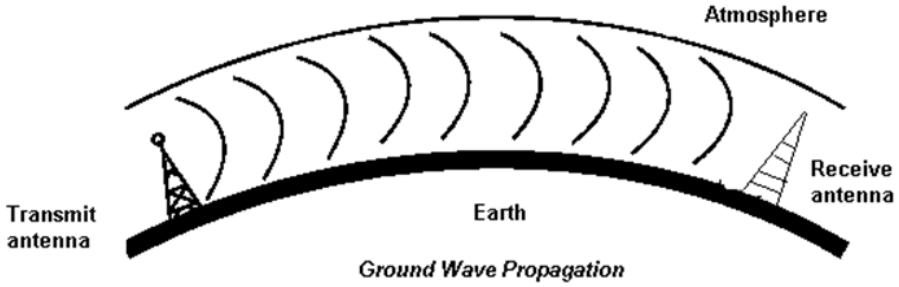


Fig. 7.7 : Ground wave propagation (below 2 MHz)

Higher frequency radio waves (2 to 30 MHz) propagate upward into the layer of the atmosphere where particles exist as ions (ionosphere). This type of transmission allows greater distances with lower power output and is called as **sky propagation**.

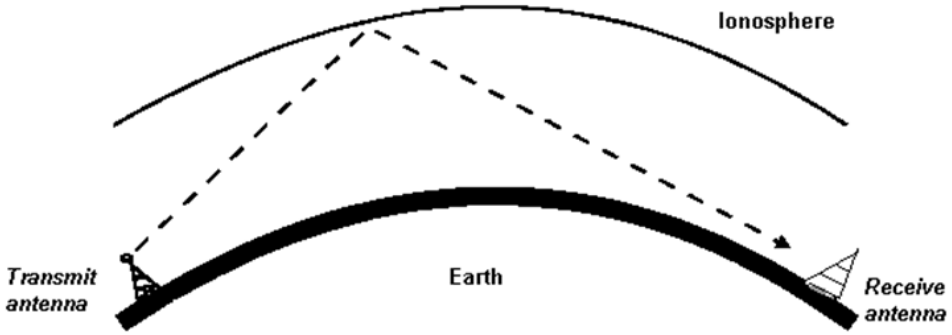


Fig. 7.8 : Sky wave propagation (2 to 30 MHz)

When very high frequency signals are transmitted in straight lines directly from antenna to antenna, it is called **line-of-sight propagation**. Antennas must be directional, facing each other, and either tall enough or close enough together not to be affected by the curvature of the earth.

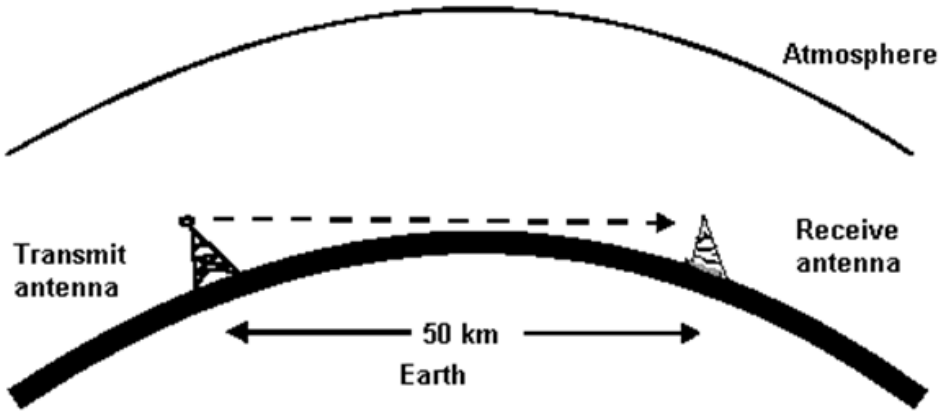


Fig. 7.9 Line-of-sight propagation (above 30 MHz)

7.5.1 Radio Waves

The radio waves are omnidirectional while the microwaves are directional. Radio waves are used for one sender and multiple receivers i.e. for multicasting property. Because of this multicast communication, it is used by radio, television, cordless phones and pagers. Radio wave is a general term to encompass frequencies in the range 3 KHz to 1 GHz. The microwaves range in frequencies between 1 to 300 GHz. Broadcast radio term may be informally used for the FM radio and VHF television and number of data networking applications.

As the radio waves are omnidirectional, they do not require dish-shaped antennas with precise alignment. Radio waves used omnidirectional antennas that send the signals in all directions. We can have different types of antennas based on wavelength and strength. Radio waves can penetrate walls. It can be considered an advantage. For example, AM radio can receive signals inside the building. It has the disadvantage that radio wave band is relatively narrow and if we subdivide it farther, it becomes narrower resulting in lower data rate for digital communications. Another problem with the radio waves is the multipath interference. Reflection from different sources can create multiple paths between antennas which is evident when TV reception displays multiple images. But because of longer wavelength, radio waves suffer less attenuation.

7.5.2 Infrared

Infrared signals are used for short range communication. It has the frequency range from 300 GHz to 400 THz. It has wavelength from 1 mm to 770 mm. Infrared communications is achieved by using the transmitters / receivers(transceivers) that modulate infrared light. Transceivers must be within the line of sight of each other directly or via reflection.

One disadvantage of infrared is that it does not penetrate walls. We cannot use infrared outside a building because the sun's rays contain infrared waves which can interfere with the communication.

When we use infrared remote control, we do not interfere with the use of the remote by our neighbours. It can be used for digital data transmission with high data rate.

As no licensing is required, there is no freuency distribution issue in case of infrared. The association that has setup standards for communication is called Infrared Data Association (IrDA). IrDA port is used to communicate a wireless keyboard to communicate with PC.

7.5.3 Microwaves

Microwaves use line of sight propogation. Microwaves frequency ranges from 1 to 300 GHz. Microwaves are used in cellular phones, satellite networks and Wireless LANs. Microwaves are unidirectional. It is useful in unicasting i.e. one to one communication. To travel longer distances, the antennas must be taller. There should not be any obstacle in between . Usually mountain tops are the preferred positions for the antennas to avoid obstacles. Microwaves can travel for one direction at a time. For two way communication two frequencies need to be allocated.At both ends a transceiver for operating two different frequencies. Repeaters are used along with antennas to regenerate the signal. Microwave is inexpensive and it allows you to communicate from anywhere.

Microwave is commonly used for both voice and television transmission.

There are two types of antennas used for microwave communications: (1) the parabolic dish and (2) the horn.

The parabolic dish is 3 m in diameter and its shape is like a parabola. It focuses a narrow beam to achieve line of sight transmission. The parabolic dish acts as a funnel, catching wide

range of waves and directing to a common point.

In horn antenna, the outgoing transmission is broadcast up a stem and deflected outward in a series of narrow parallel beams by the curved head. The scooped shape of the horn collects the received transmissions and deflected down into the stem.



Fig. 7.10 : Microwave Antenna

Typical Digital Microwave Performance

7.5.4 Satellite

A communication satellite is, in fact, a microwave relay station. It is used to link two or more ground based microwave transmitter/ receivers, known as earth stations, or ground stations. The satellite receives transmissions on one frequency band (uplink), amplifies or repeat the signal, and transmit it on another frequency (downlink). A single orbiting satellite will operate on a number of frequency bands, called **transponder channels,** or simply **transponders**.

Thus, the satellites are transponders (units that receive on one frequency and retransmit on another) that are set in geostationary orbits directly over the equator. These geostationary orbits are 36,000 km from the Earth's surface. At this point, the gravitational pull of the Earth and the centrifugal force of Earth's rotation are balanced and thus cancel out each other. Centrifugal force is the rotational force placed on the satellite that wants to fling it out into space. The following figure shows a general way of communication through the satellite.

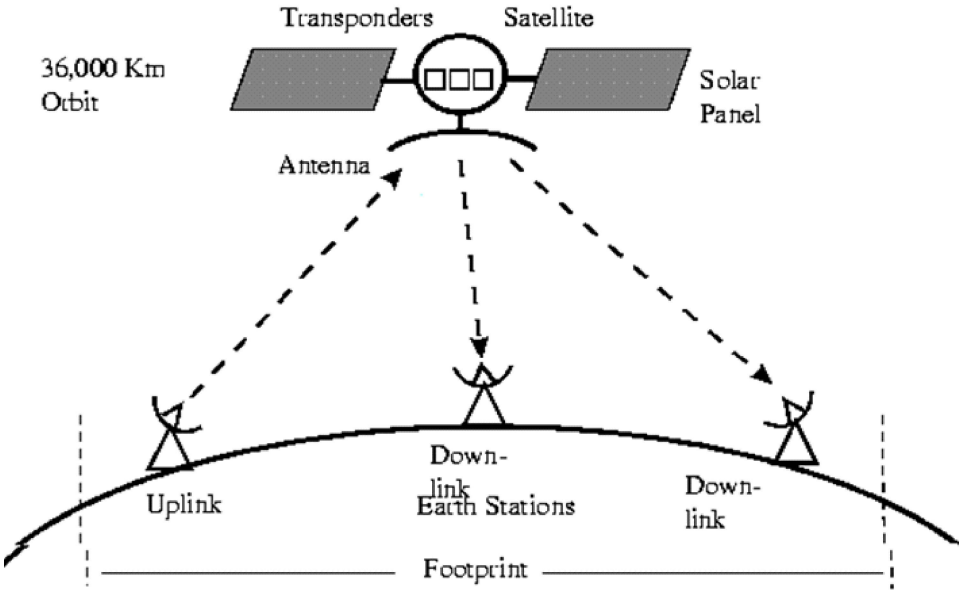


Fig. 7.11 : Satellite Communication

7.6 WIRELESS LANS (IEEE 802.11)

Now-a-days Wireless LAN becomes one of the fastest growing technologies and everyone likes to connect without using cables. One can find wireless LAN in office building, colleges and public areas. At home, a wireless LAN can be used by connecting roaming devices to Internet.

Wireless LANs are generally catogorized according to the transmission techniques. There are two types of categories:

Infrared (IR) LANs : As the infrared technology can not penetrate walls, an individual cell of an IR LAN is limited to a single room.

Spread spectrum LANs : The technology that uses this type LANs is spread spectrum transmission technologies. They operate in the ISM(industrial,scientific and medical) microwave bands in most of the cases. They operate in these bands so that there will be no licensing issue from Federal Communications Commission (FCC).

IEEE 802.11 : IEEE formed a new specification IEEE 802.11, specially for the development of wireless LANs, MAC protocol and physical medium specification. The following lines define the terminologies used in the IEEE 802.11 standard.

IEEE 802.11 terminology :

Access Point(AP) : Any entity that has station functionality and provides access to the distribution system via wireless medium for associated stations.

Basic service set (BSS) : A set of stations controlled by a single coordination function.

Coordination function : The logical function that determines when a station operating within a BSS is permitted to transmit and may be able to receive PDUs

Distribution System DS : A system used to interconnect a set of BSSs and integrated LANs to create an ESS.

Extended service set : A set of one or more interconnected BSSs and integrated LANs that appear as a single BSS to the LLC layer at any station associated with one of these BSSs.



CHECK YOUR PROGRESS

Q.7. What is transmission media?

.....

.....

Q.8. What are the three types of guided media?

.....

.....

Q.9. Write two advantages and disadvantages of fiber optics cable.

.....

.....

Q.10. Name the three ways for wireless data to be propagated.

.....

.....



7.7 LET US SUM UP

- Transmission media can be classified two main categories: guided media and unguided media.
- Guided media uses physical cable for data transmission, whereas in case of unguided media, air is the main transmission medium.
- The three main types of guided media are twisted pair cable, coaxial cable and fiber optics cable.
- Fiber optics cable is the latest, best and most expensive.
- Unguided media can be classified as radio waves, microwaves and infrared waves.
- Wireless data are transmitted through ground propagation, sky propagation and line-of-sight propogation.
- Radio waves are omnidirectional.

- Microwaves are unidirectional; propagation is line-of-sight.
- Infrared waves are used for short-range communication such as between a PC and a peripheral device.
- IEEE 802.11 standard defines for Wireless LANs.



7.8 FURTHER READING

- Data and Computer Communications by William Stallings, Pearson Education.
- Data Communications and networking by Behrouz A. Forouzan, Tata McGrawHill.



7.9 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : The different types of cables used in LANs are : Coaxial cables, shielded and unshielded twisted-pair cables and fiber optic cables.

Ans. to Q. No. 2 : A “vampire tap” is inserted into the thick coax to which the AUI is connected.

Ans. to Q. No. 3 : 75 Ω

Ans. to Q. No. 4 : 600 Mbps

Ans. to Q. No. 5 : Twisting of wires reduces the effect of electromagnetic interference which may be generated from nearby electric motors, fluorescent tube lights etc.

Ans. to Q. No. 6 : In optical fiber light signals travel by following the principle of total internal reflection.

Ans. to Q. No. 7 : Transmission media are the physical infrastructure components that carry data from one computer to another computer. They are the basis of data communication.

Ans. to Q. No. 8 : Guided media can be subdivided into three main categories.

1)Twisted Pair, (2) Coaxial cable and (3) Fiber Optics.

Ans. to Q. No. 9 : The two advantages of fiber optics cable are :
(i) Huge bandwidth, (ii) Resistance to noise.
The two disadvantages of fiber optics cable are :
(i) Cost, (ii) Maintenance overhead

Ans. to Q. No. 10 : Wireless data are transmitted through ground propagation, sky propagation and line-of-sight propagation.



7.10 MODEL QUESTIONS

Q.1. How does guided media differ from unguided media.

Q.2. Write about a use of each class of guided media.

Q.3. What are the three major classes of guided media.

Q.4. Name the advantages of optical fiber over twisted pair and coaxial cable.

Q.5. Name the three ways for wireless data to be propagated.

Q.6. Write what you know about radio waves and microwaves.

****

UNIT 8 : INTERNET

UNIT STRUCTURE

8.1 Learning Objectives

8.2 Introduction

8.3 Internet - A Definition

8.4 Internet Architecture

- 8.4.1 Peer-to-Peer
- 8.4.2 Client-Server

8.5 Accessing Internet

- 8.5.1 Dial-Up Connections
- 8.5.2 ADSL Connections
- 8.5.3 Cable Connections

8.6 Internet Service Providers

8.7 Organization of Internet

- 8.7.1 Internet Protocol Suite
- 8.7.2 IP Address
- 8.7.3 Domain Name System
- 8.7.4 Uniform Resource Locators

8.8 Application of Internet

8.9 Let Us Sum Up

8.10 Answers to Check Your Progress

8.11 Further Reading

8.12 Model Questions

8.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- define Internet and describe the growth of Internet
- describe the organization of Internet
- describe web servers, web browsers and search engines
- learn about server and client side technology

8.2 INTRODUCTION

Ever since the advent of computers, researchers have indulged in a continuous quest to bring the world closer through computers. This effort gave rise to the Internet. It is through the Internet that today millions of people communicate and share information, regardless of their location. This unit will focus on the basics of the Internet and also shed light upon how the Internet works along with its applications.

8.3 INTERNET – A DEFINITION

The Internet can be defined as a network of globally connected computers that is decentralized by design. This definition can be broken down into three parts. Let us understand each part of the definition in isolation.

It is a network. A network is a collection of computers. The Internet can also be referred to as a network because it is a collection of millions of computers.

Globally connected computers. This means that you can be connected to the Internet, regardless of your location. The Internet has brought people in the world closer by connecting computers located in the remotest of locations.

Decentralized design. The Internet has a decentralized design. That is, there is no centralized body that controls the way in which the Internet functions. The Internet does provide online services that are centrally administered, but as a whole, it would not be incorrect to say that the Internet has a decentralized design. Each computer connected to the Internet is called a *host*. The operator/ user of a particular host can choose from the millions of available Internet services and can also make services available through the Internet.

You can consider Internet to have the following characteristics :

- *A complex network* - with simplified definition as a 'network of network'
- *Disorganized* - Internet can be cumbersome and confusing, even for experienced users

- *A decentralized system* - millions of individual networks and over 200 million individual computers connected through the world
- *Composed of many billions of files*(web pages).
- *Dynamic* - changing every minute of every day. On an average, a new network is connected to the Internet every 30 minutes.
- *Expanding exponentially* - the Internet is growing at the rate not less than 15% per month.

8.4 INTERNET ARCHITECTURE

It is important to understand what the term “architecture” means. The notion of network architecture was introduced during the Internet research phase by the research community that had developed the ARPAnet protocols. Network architecture is a set of high-level design principles that guides the technical design of the network, especially the engineering of its protocols and algorithms. There are two most commonly used architecture in Internet technology : *peer-to-peer* and *client server architecture*.

8.4.1 Peer-to-Peer

Peer-to-peer is a communication model in which each and every node is capable of sharing information and can initiate a communication session. On the Internet, peer-to-peer (referred to as P2P) is a type of transient Internet network that allows a group of computer users with the same networking program to connect with each other and directly access files from one another's hard drives. **Napster** and **Gnutella** are examples of this kind of peer-to-peer software.

8.4.2 Client-Server

The Client-Server Architecture is based on the principle where the client computer requests for some data and the data are sent by the server computer through the network. The concept of *client/ server* computing has particular importance on the Internet because

most of the programmes are built using this design. Servers are powerful computers or processes dedicated to managing disk drives (file servers), printers (print servers), or network traffic (network servers). Clients are PCs or workstations on which users run applications. Clients rely on servers for resources, such as files, devices, and even processing power.

The following figure shows the two architectures :

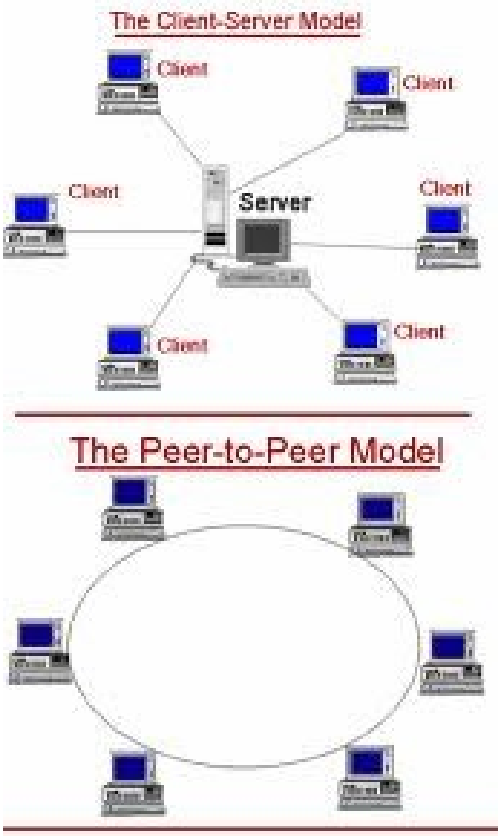


Fig. 8.1 : Client Server and peer-to-peer architecture

8.5 ACCESSING INTERNET

In order to access the Internet, one requires the following:

- A client computer with necessary hardware and client software
- The connectivity through Internet Service Provider (ISP) for flow of data
- Host computer(s) which is also called Web server hosting the desired data.

The necessary hardware devices that must be had on the client computer are :

- Processor of Pentium (Intel based) or AMD or Macintosh
- Network Interface Card (NIC)
- Modem, it can be an external or internal (fitted inside the computer)

The necessary softwares that must be found on the client computer are :

- Operaring system like WIndows, UNIX, LINUX or OS/2 etc.
- Web browser like Netscape, Internet explorer, FireFox etc. (it must be compitable with the operating system)

Common methods of Internet access in homes include **dial-up, landline broadband(ADSL)** (over coaxial cable, fibre optic or copper wires), **Wi-Fi, satellite and 3G technology cell phones**. We will concentrate basically on the following methods

- dial-up connections
- ADSL connections and
- Cable Connections

8.5.1 Dial-Up Connections

Dial-up connections is a type of Internet connectivity that operates through a standard telephone line. Before a person can subscribe to a dial-up service, he or she must have a computer and dial-up modem. A telephone line feeds into the modem. The modem is controlled by software in the computer; for example, the Network Connections utility that comes with Microsoft Windows operating systems. Here you can setup a profile for the ISP (Internet Service Provider, like BSNL), which will tell the modem what phone number to call and how to communicate with the dial-up service. The ISP itself provides this information.

Upon joining a dial-up service, the subscriber chooses a username and password. Once the modem calls the phone number and makes a connection, a "*handshake*" takes place in which information is exchanged between the computer modem and the remote server. The username and password are supplied by the

modem. This grants the user an access through the dial-up gateway to the Internet. Dial-up connections can be very economic and are widely available, the cost per minute is comparable to that of a local phone call, or priced as a monthly plan which will include a certain amount of time. As these connections use a standard modem the hardware costs are minimal. Dial-up connections are very slow compared to other connection types. When connected to the internet the same phone line cannot be used for phone calls, so if anyone calls you when you are connected, they will get the busy signal.

Dial-up connections transfer data over an analogue line, so before the data is sent it has to be converted from digital to analogue, Likewise, when data is received it has to be converted from analogue to digital (this is what the modem does), This adds a performance overhead which affects the speed of the connection.

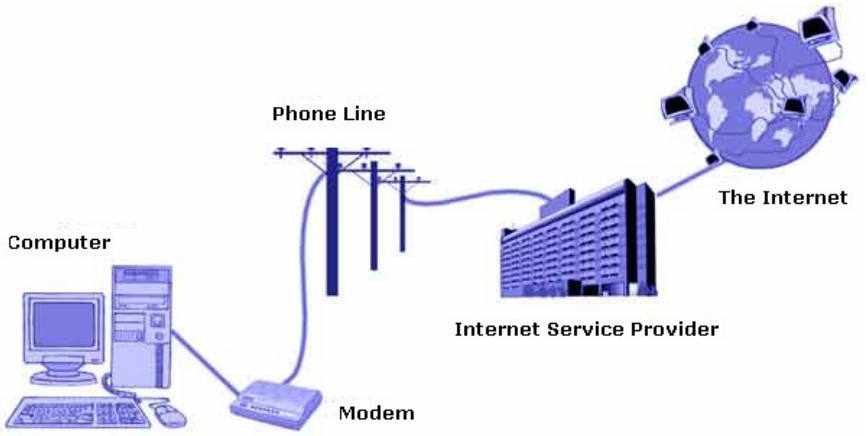


Fig. 8.2 : Dial-up Connection

8.5.2 ADSL Connections

ADSL (Asymmetric Digital Subscribers Line) connections are becoming more and more widely available and can provide an excellent Internet connection. The connections work by splitting your phone line into two separate channels, one for data (Internet) and other voice (phone calls), which means you can talk on the phone and be connected to the Internet at the same time.

You will often see ADSL connection services advertised as having different speed specifications. Given below are some common configurations:

256Kbps/128Kbps or 512Kbps/128Kbps

1Mbps/256Kbps or 2Mbps/512Kbps

8Mbps/1024Kbps

Notice that there are two values to each configuration. The first figure states the download speed and the second figure is the maximum upload speed. As an example let us take the second configuration 512Kbps/128Kbps. This means that you can potentially download data at a speed of 512Kbps and upload data at 128Kbps.

Advantage of ADSL connection : ADSL technology eliminates the need for a second phone line by allowing voice and data transfer at the same time (you can use the phone as normal while connected to the Internet). Because ADSL transfers data digitally, it eliminates the usual performance overhead associated with standard dial-up connections. In other words, ADSL doesn't need to convert the data from digital to analogue and back again.

Disadvantage of ADSL connections : ADSL connections are not available to everyone. You should always ensure that you have ADSL coverage in your area. The hardware costs can be quite significant as you will need a special ADSL modem and ADSL filters to use the service. Most ISPs allow you to hire these items which can reduce the initial cost. Because ADSL connections are always on you will need a firewall to protect your PC.

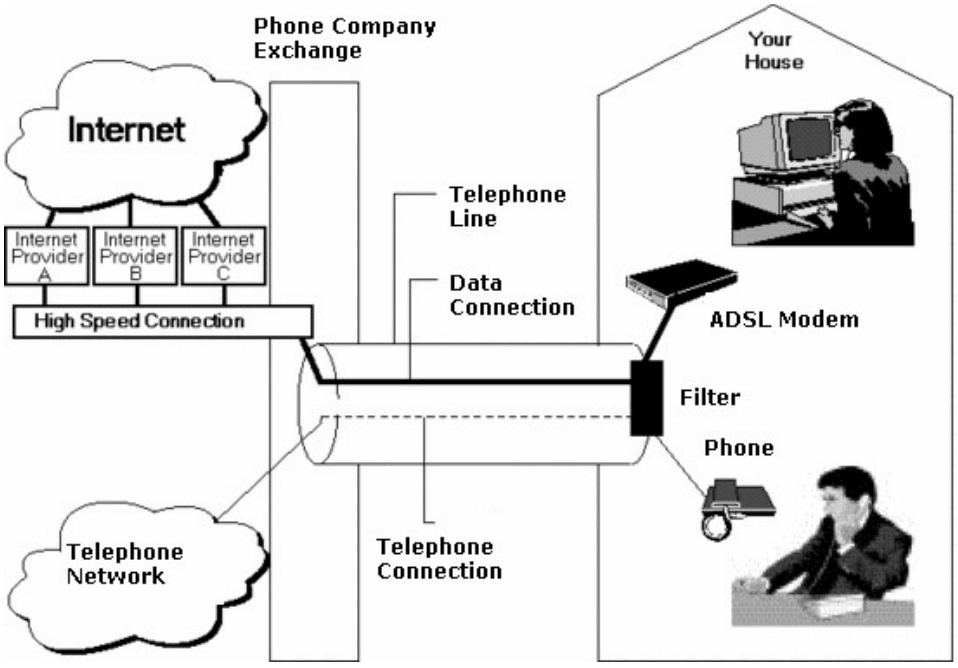


Fig. 8.3 : ADSL Connection

8.5.3 Cable Connections

Cable connections are considered one of the best types of Internet connection available to the home user because they offer very fast and reliable connections with a fixed monthly fee. Cable companies usually offer different packages to suit different Internet subscribers.

The different packages will offer different speed specifications and bandwidth limits. Because a cable connection uses a totally separate medium to transfer data it does not affect your ability to make/receive phone calls.

Advantage of Cable connection : Speed is a major reason for having a cable connection. Like ADSL connections, cable connections transfer data digitally, eliminating any digital/analogue conversion overhead. Cable connections are always on, eliminating long waits to make a connection.

Disadvantage of Cable connection : Cable connections are not available in every area. You will need to contact the cable company of your choice to ensure that you have coverage.

8.6 INTERNET SERVICE PROVIDER (ISP)

Internet service providers (ISPs) are companies that help the users to connect to the Internet for a monthly fee. In return, they provide a username, a password, and telephone number. The username and the password are used to authenticate the user on the Internet. The telephone number is used to establish connection with the dial-up server of the ISP. However, ISPs also support other forms of Internet access like DSL and cable connections and other dedicated technologies. In addition to providing Internet services to individual users, ISPs also provide Internet services to large companies in return for an annual fee. When such a company subscribes to an ISP, the company provides Internet access to all the computers on the corporate network. Examples of ISP in india are Bharti Airtel Ltd, Reliance Communication, BSNL etc.



CHECK YOUR PROGRESS

Q.1. Choose the correct answer

a) In a peer-to-peer communication model a group of computers use

i) different network program

ii) same network program

iii) none of the above

b) Internet is

i) complex system ii) decentralized system

iii) dynamic system iv) all of the above

c) Dial-up connections are

i) very fast ii) fast

iii) very slow iv) none of these

d) Dial-up connections transfer data over

i) analogue line ii) digital line

iii) both analoogue and digital

iv) none of these

e) In ADSL connection data and voice

i) can be sent at the same time

ii) can the send one at a time

iii) cannot be sent iv) all the above

8.7 ORGANIZATION OF INTERNET

The Internet system consists of a number of interconnected packet networks supporting communication among host computers using the ***Internet protocols (IP)***. A computer linked on the Internet is known as the host computer. The data to be moved from one host to another host is broken into small pieces called ***packets***. Each packet has a header with the address of destination host. The packets of different sizes move on in various networks before reaching the destination. Various packets of one file may take different routes to reach a destination. The different networks on the Internet are connected with special purpose devices called **routers,** a term which is already familiar to us. These routers look for destination address given on each packet and direct the packet to take the best route to the destination. Routers take their decisions based on information that is constantly reaching them from all over the Internet. They also hear from other routers about the links that are down or congested/slow, or about routers that are no longer accepting packets for certain destinations. Each packet's destination and proposed route are evaluated individually, in the blink of an eye, and sent off along the best route for that particular packet at that particular moment. The following diagram(8.4) shows the function of routers.

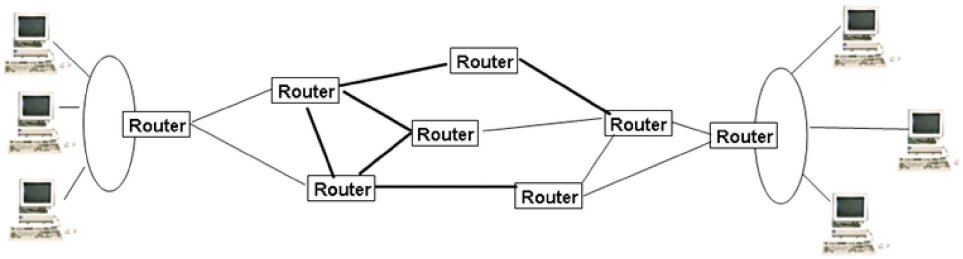


Fig. 8.4 : Routers connected on the Internet

8.7.1 Internet Protocol Suite

The Internet protocol suite is the set of communications protocols that implement the *protocol stack* on which the Internet runs. A protocol stack is a complete set of protocol layers that work together to provide networking capabilities.

A protocol is a mutually agreed-upon format for doing something. With regard to computers, it most commonly refers to a set of rules (i.e., a standard) that enables computers to connect and transmit data to one another; this is also called a ***communications protocol***. The Internet protocol suite is sometimes called the ***TCP/IP protocol suite***, which refers to the two most important protocols in it : the *Transmission Control Protocol (TCP)* and the *Internet Protocol (IP)*. A protocol can be implemented by hardware, software, or a combination of the two. It is called a protocol stack because it is typically designed as a hierarchy of layers as shown in the figure below, each supporting the one above it and using those below it. Each layer solves a specific set of problems involving the transmission of data and provides well-defined services to the layers above it. The application layer is logically closer to the the application programs and deal with more abstract data, and it relies on lower

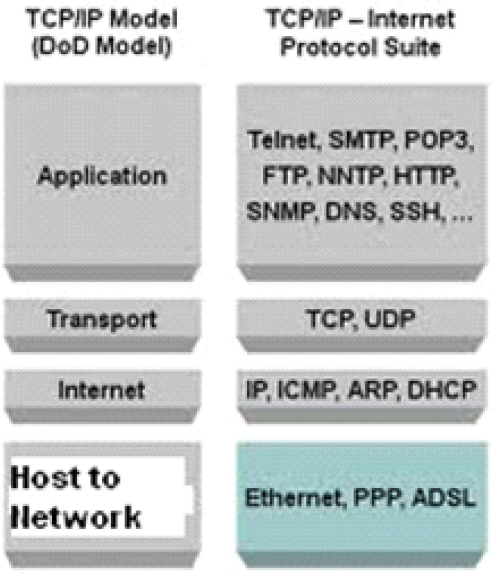


Fig. 8.5 : TCP/IP protocol suite

layers to convert their data into forms that can be physically manipulated for transmission.

Let us see the functions of a few protocols of application layer :

Telnet : Telnet is a popular client-server application program used for terminal services. Telnet is an acronym for Terminal Network. Telnet protocol is used to establish an on-line connection to a remote machine, so that the client can access the utility or application program from the remote machine.

FTP : The FTP (File Transfer Protocol) is a standard network protocol used to copy a file from one host to another over a TCP/IP-based network, such as the Internet. Using FTP, you can also update (delete, rename, move, and copy) files at a server. You need to **log on** to an FTP server.

SMTP : The SMTP (Simple Mail Transfer Protocol) protocol is used by the Mail Transfer Agent (MTA) to deliver your eMail to the recipient's mail server in the Internet. The SMTP protocol can only be used to send emails, not to receive them.

POP 3 : The POP (Post Office Protocol 3) protocol provides a simple, standardized way for users to access mailboxes and download messages to their computers. When using the POP protocol all your eMail messages will be downloaded from the mail server to your local computer.

IMAP : The IMAP (Internet Message Access Protocol) is one of the most prevalent Internet standard protocols for e-mail retrieval.

HTTP : The HTTP (Hypertext Transfer Protocol) is the set of rules for exchanging files (text, graphic images, sound, video, and other multimedia files) on the **World Wide Web.** Relative to the **TCP/IP** suite of protocols (which are the basis for information exchange on the Internet), HTTP is an application **protocol.**

DNS : The DNS (Domain Name System or Service or Server), is an Internet service that translates domain names like www.google.com or www.kkhsou.org etc. into IP addresses.

8.7.2 IP Address

So far we have discussed the fundamental concepts of Internet. Our discussion will be incomplete if we donot mention about the Internet address system. Let us see what is Internet address briefly.

The number of nodes on the Internet are increasing day-by-day. Always remember that each host computer or node on the Internet has its own unique address through which communication is possible with every one. To identify a host on the Internet, three addressing systems have been evolved : (a) A numerical system called I*P addressing*, (b) a hierarchical naming system called the *Domain Name System*, and (c) an addressing system called URLs, which are used for identifying sites on the web.

An ***Internet Protocol address (IP address)*** is a numerical label that is assigned to any device connecting to the Internet and that uses the Internet Protocol for communication between its nodes. Each IP address has specific components and follows a basic format. Each IP address is 32-bit logical address that is divided into two main parts: the *network number* and the *host number*. The network number identifies a network and must be assigned by the Internet Network Information Center (InterNIC) if the network is to be part of the Internet. The host number identifies a host on a network and is assigned by the local network administrator.

The 32-bit IP address is grouped eight bits at a time, separated by dots, and represented in decimal format (known as *dotted decimal notation*). Each bit in the octet has a binary weight (128, 64, 32, 16, 8, 4, 2, 1). The minimum value for an octet is 0, and the maximum value for an octet is 255. The following figure illustrates the basic format of an IP address 198.168.1.133 (written in decimal).

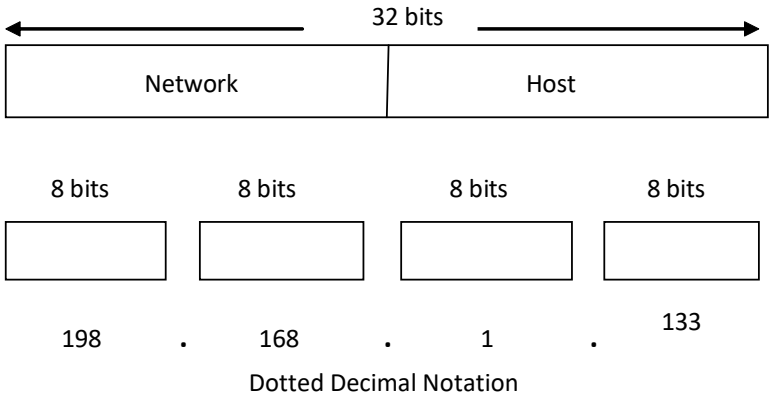


Fig. 8.6 : Format o f an 32-bit IP address

We will not discuss IP addressing in more details now but you should know that IP addressing is again divided into five different classes which are A, B,C, D, and E. The classes A, B, and C are available for commercial use only. In the following classes A, B and C are shown :

	1st octet	2nd octet	3rd octet	4th octet
Class A	Network	Host	Host	Host
Class B	Network	Network	Host	Host
Class C	Network	Network	Network	Host

Fig. 8.7 : IP Address Class

You have seen that IP addresses are numeric and it may be quite difficult to remember such long numeric data. So there is a system to translate it into simple English names. The Domain Name System (DNS) serves as a directory programme for IP addresses and it takes care of translating IP addresses to simple English names.

8.7.3 Domain Name System

DNS refers to Domain Name System and represents a powerful Internet technology for converting domain names to IP addresses. Behind every site, there is an IP address. But, while it is easy to remember the name of a website, it is quite hard to remember the exact IP address. For example, everybody knows about Google.com, but if you had to remember "74.125.45.100", things would have been much harder.

How does a DNS program works? - Every time a domain name is typed in a browser it is automatically passed on to a DNS server, which translates the name into its corresponding IP address.

This system offers :

- an **hierarchical namespace** where each name is unique
- a system of **distribution servers** enabling namespace to be made available.
- a **client system** which helps to find out the IP address corresponding to a name.

DNS Hierarchy : The structure of the DNS system relies on a tree structure. The top-most node of the tree is the DNS root domain (**.**), under which there are subdomains, such as .com, .edu, .gov, and .mil, which are called ***top level domains (TLDs)***. Each node of the tree is called a **domain name**. The end of a branch is called the **host**, and corresponds to a machine or entity on the network. The

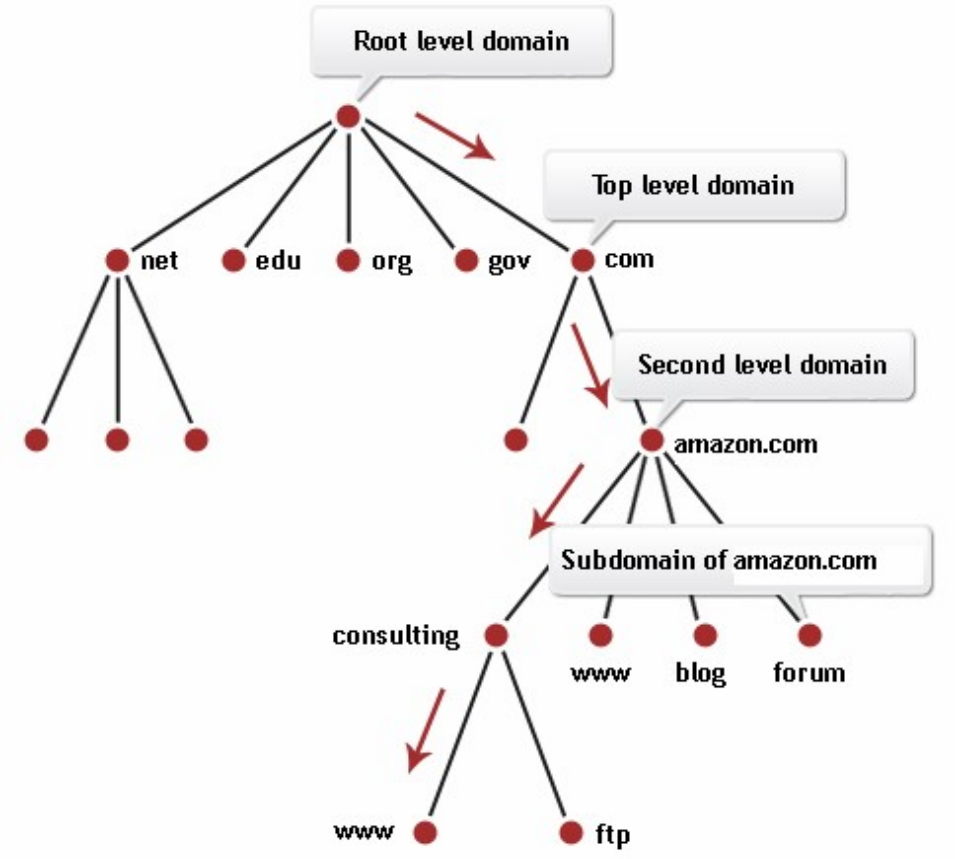


Fig. 8.8 : DNS Hierarchy

host name given to it must be unique in the respective domain. In the following figure DNS hierarchy is shown.

The root DNS servers (root name servers) keep track of all the Top Level Domain (TLD) name servers. The client queries the root name servers to resolve a request for the given domain name. In response the root name server provides the address of the TLD name server for the given query in which the domain name ends with. For example, if a client requests for google.com, the root name server will address the client to the com DNS server so as to solve his query.

The top level name server holds the list of authoritative name server in their respective domain. For example, the ***com*** domain holds the address of yahoo.com, google.com etc. Every TLD includes many second-level domains; for example, amazon.com as shown. Every second level TLD may include number of third level domains. In the figure you have noticed that TLD divides the Internet domain name space into several domains. Most commonly used domains are :

com – Usually used by commercial organization. Eg. Yahoo (yahoo.com)

edu – Usually used by educational institutes. Eg. San Jose State University (sjsu.edu)

org – Used by non profit organizations. Eg. IEEE (ieee.org)

mil – used by military organizations. Eg. US army (army.mil)

net – In earlier days it was used to represent the network infrastructure. Now a days it is open public for any commercial organization.

gov – used to represents government organization. Eg. assam. gov.in

The top level domain names also use two-letter country designations which is known as *geographic domain*. For example,

.us is used for the United States,

.ca for Canada (not California),

.uk or .gb for the United Kingdom or Great Britain,

.il for Israel and

.in for India etc.

8.7.4 Uniform Resource Locators (URL)

A URL (Uniform Resource Locator) is the the global address of documents and other resources on the World Wide Web. The type of resource depends on the Internet application protocol. Using the World Wide Web's protocol (HTTP), the resource can be an HTML page, an image file, a Java applet or any other file supported by HTTP. The URL contains the the following parts

- name of the protocol required to access the resource
- a domain name that identifies a specific computer on the Internet where the resource(file that we want to access) is located,
- path name of the file on the host computer.

The protocol identifier and the resource name are separated by a colon and two forward slashes. In the following an example is shown :

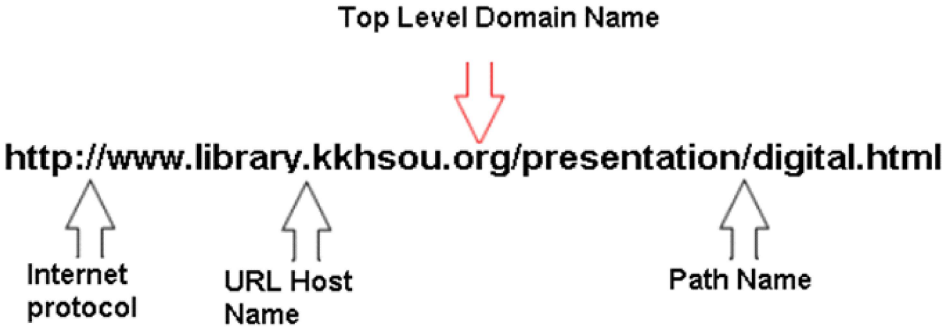


Fig. 8.9 : URL example



CHECK YOUR PROGRESS

Q.2. Choose the correct answer

a) For connection of the Internet, you will need

i.an IP address ii. a TCP/IP protocol

iii.an ISP iv. All the above

b) A host on the Internet finds another host by its

i. postal address ii. IP address

iii. electronic address iv. none of above

c) User can get files from another computer on the Internet by using

i. FTP ii. HTTP iii. UTP iv. i & ii both

d) IP addresses are translated into simple English names by the following system -

i. HTTP ii. SMTP iii. DNS iv. FTP

e) An IP address contains

i. 64 bits ii.16 bits iii. 8 bits iv. 32 bits

Q.3. Define IP address, Domain name and Host.

.....

.....

.....

.....

.....

.....

8.8 APPLICATION OF INTERNET

The Internet is a global system of interconnected computer networks that use the standard Internet Protocol Suite (TCP/IP) to serve billions of users worldwide. It is a network of networks that consists of millions of private, public, academic, business, and government networks, of local to

global scope, that are linked by a broad array of electronic and optical networking technologies. Now-a-days the application of Internet touches the every field of modern society. Let us briefly focus some of them.

Email : Email is now an essential communication tool in business. It is also excellent for keeping in touch with family and friends. The advantages to email is that it is free (no charge per use) when compared to telephone, fax and postal services.

Information : There is a huge amount of information available on the Internet for just about every subject known to man, ranging from government law and services, trade fairs and conferences, market information, new ideas and technical support.

Services : Many services are now provided on the Internet such as online banking, job seeking and applications, and hotel reservations. Often these services are not available off-line or cost more.

E-Commerce : The Internet is a very effective way to buy and sell products all over the world. Business houses use the Internet to provide product information, online support service, etc. Companies carry out online trading including advertising, selling, buying, distributing products, and providing after-sales services through the Internet.

Communities : Communities of all types have sprung up on the Internet. Institutions use the Internet for voice and video conferencing and other forms of communication that allow people to telecommute, or work from a distance. It is a great way to meet up with people of similar interest and discuss common issues. Scientists and scholars use the Internet to communicate with colleagues, to perform research, to distribute lecture notes and course materials to students, and to publish papers and articles.

Entertainment : Entertainment is another popular reason why many people prefer to surf the Internet. In fact, media of Internet has become quite successful in trapping multifaceted entertainment factor. Downloading games, visiting chat rooms, online gaming or just surfing the Web are some of the uses people have discovered. There are numerous games that may be downloaded from the Internet for free.

a very large amount of space if they all had to be stored back on a web site's server.

Security : If a cookie was stored on a web site, then it can be accessed by anyone with access to that web site. However, when the cookie is stored on your computer, then it can't be accessed by hackers that break into the web site.



CHECK YOUR PROGRESS

Q.4. a) Which of the following protocols is used by the Internet mail?

i. HTTP ii. TCP/IP iii. SMTP iv. none of these

b) Find the odd item from the following -

i. Yahoo! ii. HotBot iii. Lycos iv. Win2000

c) Computers on the Internet owned and operated by education institution form part of the

i. com domain ii. edu domain

iii. mil domaiin iv. gov domain

d) For a small website, one needs to buy space from the

i. Network administrator ii. Telephone exchange

iii. ISP iv. None of these

e) Find the odd item from the following :

i. FTP ii. SMTP iii. JAVA iv. HTTP

Q.5. Fill in the blanks :

i. A _____ is a program that is executed by the web server in response to a request.

ii. The client-side scripts are executed by the _____.

iii. Web _____ allow a Web site to store information about you as well as to track your visits to it over a prolonged period of time.

Q.6. Choose the appropriate option:

i) Which of the following protocols is responsible for Internet mail?

a) TCP/IP b) HTTP c) FTP

d) None of these

ii) A user can get files from another computer on the Internet by using

a) FTP b) HTTP c) UTP

d) None of these

iii) Take the odd item out

a) Internet Explorer b) Mozilla Firefox

c) Google d) Netscape navigator

iv) Take the odd item out

a) PHP b) JSP c) HTML d) XML

v) FTP is used for

a) Uploading files only b) Downloading files only

c) Both (a) and (b) d) None of these



LET US KNOW

An ***Intranet*** is a Web-based architecture used for managing internal information of an office, company, university or college or even a large library that runs on IP protocols. It means that an intranet is a private network that is contained within an enterprise.. It uses Internet protocols such as **TCP/IP** (Transmission Control Protocol/Internet Protocol), **HTTP** (Hypertext Transfer Protocol), **HTML** (Hyper Text Markup Language) **SMTP** (Small Message Transfer Protocol), **FTP** (File Transfer Protocol), and **AAA** (Access, Authorization, and Authentication). It may consist of many interlinked Local Area Networks and also use leased lines in the Wide Area Network. The main purpose of an intranet is to share information within the organization and computing

resources among employees. An intranet can also be used to facilitate working in groups and for teleconferences.

An ***Extranet***, or extended Intranet, can be defined as a private network of linking branch offices or several cooperating organizations located outside the walls of any organisation. An Extranet service uses existing Intranet interactive infrastructure, including standard servers, e-mail clients and Web browsers. This makes Extranet far more economical than the creation and maintenance of a proprietary network. It enables trading partners, suppliers and customers with common interests to form a tight business relationship and a strong communication bond.



8.9 LET US SUM UP

The Internet is a network of networks. It connects millions of computers and thousands of computer networks throughout the world. The Internet has revolutionised our society, our economy and our technological systems. This unit has discussed specifically the technology associated with the Internet such as Internet architecture, organization of Internet (Internet protocol and Internet addressing). It is important to know the various information resources and services available in Internet or World Wide Web (www). It is equally important to know about search engines, selecting and searching from a particular search engine. Client side technologies includes HTML, DHTML, VB Script, Java Script,ECMAScript, JScript, Java applets and Flash Animations etc. Server side technologies includes PHP, JSP, CGI/ Perl, XML/XSL, MYSQL, PostgreSQL etc.



8.10 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : a) ii, b) iv, c) iii, d) i, e) i.

Ans. to Q. No. 2 : a) iv, b) ii, c) i, d) iii, e) iv.

Ans. to Q. No. 3 : IP address is the Internet Protocol (IP) address given to every computer connected to the Internet. An IP address is needed to route information much like a street address or PO box is needed to receive regular mail. Example: 177.198.48.157

Domain name is a text name which a computer network registers. The domain name is used to give computers text names rather than using the numeric IP addresses. This is like getting a vanity phone number that spells out a word to make it easy to remember.

Computer (host) names are names given to individual computers. Each host name corresponds to an IP address. Host names and domain names are optional and everything will work fine using just IP addresses.

Ans. to Q. No. 4 : a) iii, b) iv, c) ii, d) iii, e) iii

Ans. to Q. No. 5 : i) script, ii) browser, iii) cookies

Ans. to Q. No. 6 : i) a, ii) a, iii) c, iv) c, v) c



8.11 FURTHER READING

- Data and Computer Communications by William Stallings, Pearson Education.
- Data Communications and networking by Behrouz A. Forouzan, Tata McGrawHill.
- *"*Web technologies: a computer science perspective*", Jeffrey C. Jack son*, Pearson Prentice Hall.
- *"*Web Technology : A Developer's Perspective*", Gopalan, Gopalan/ akilandeswari*, Pearson Prentice Hall.



8.12 MODEL QUESTIONS

Q.1. What is Internet? Explain why Internet is called a 'network of networks'?

Q.2. What do you mean by the Internet architecture? Describe the most commonly used Internet architecture.

Q.3. Differentiate between the dial-up and ADSL Internet connection.

Q.4. Differentiate between the ADSL Internet connection and cablle Internet connection.

Q.5. What is Internet protocol suite? List out the functions of five Internet protocols.

Q.6. What is an IP address? Discuss the function of domain name system.

Q.7. What is an URL? Using an example identify the components of an URL.

Q.8. Write short notes on the following :

a) Dial-up connections b) ADSL connections

c) Internet Service Providers d) IP Address

e) Domain name system f) Uniform resource locator

****

UNIT 9 : WORLD WIDE WEB

UNIT STRUCTURE

9.1 Learning Objectives

9.2 Introduction

9.3 Web Pages

9.4 Web Browsers

9.5 Web Server

9.6 Web Search Engines

9.7 Web Cookies

9.8 Technologies in WWW

9.9 Let Us Sum Up

9.10 Answers to Check Your Progress

9.11 Further Reading

9.12 Model Questions

9.1 LEARNING OBJECTIVES

After going through this unit, you will be able to :

- define Internet and elaborate on its growth
- describe the organization of Internet
- illustrate the world wide web
- describe web servers, web browsers and search engines
- know the server and client side technology

9.2 INTRODUCTION

The World Wide Web- known as WWW or W3 or simply, the Web - is one of several Internet resource discovery tools developed to help people publish, organize and provide access to information on the Internet. The Web was first developed by Berners Lec in 1989 while working at CERN, European Particle Physics Laboratory in Switzerland, and has now become the most powerful, and popular, resource discovery toll on the Internet. The

WWW can be defined as a hypertext, multimedia, distributed information system that provides links to hypertext documents, as well as to many other Internet tools and databases.

The most important things required to understand the underlying mechanism of the Web are the client-server architecture, the Hypertext transfer protocol (HTTP), Universal Resource Locators (URLs) which we have already seen very briefly and the another important resource is Hypertext Markup Language (HTML). The most common method of creating a web document is through the use of Markup Languages. Most of these are created by adding a set of formatting code to ASCII text to show fonts, justifications, links, etc. on the web. A markup language is a language that uses tags to indicate a change in presentation style or a change in content type.

By now, you can imagine that the web ***documents*** are referring to each other by ***links***. This simple view is known as the ***hypertext*** paradigm. The reader sees on the screen a document with sensitive parts of text representing the links. A link is followed by mere pointing and clicking.

9.3 WEB PAGES

A web page is nothing but a hypertext document which is suitable for the World Wide Web and can be accessed through a web browser and displayed on a monitor or mobile device. Web pages are usually written in HTML or XHTML and may contain static contents or dynamic contents. Web pages provide navigation to other web pages via ***hypertext links***. Hypertext is a system of organizing, navigating, distributing and publishing information electronically.

Contents of a Web Page :

A web page may contain various types of information, which can be divided into two main groups - *perceived information* (visible to the website visitor) and *hidden information* (hidden from the visitor's eye).

Depending on the purpose and target audience of a website, its perceived information could be textual, non-textual and interactive.

The non-textual information includes static images (e.g. GIF, JPEG, PNG or TIFF), animated images (e.g. animated GIF, Flash, Shockwave, Java Applet), vector formats (e.g. Flash, SVG), audio file formats (MIDI, WAV, MP3, Java Applets), video files (WMV, RM , FLV, MPG, MOV) etc. Interactive content on web pages could be displayed via DHTML, interactive illustrations, DHTML based buttons. *Hyperlink* or *link* is a navigational element in a web page through which navigation between the content on separate pages is possible.

The hidden information on web pages includes comments, metadata, charset details, CSS visual specifications, scripts (e.g. the interactivity focused JavaScript) etc.

Depending on the type of information contained in a web page, web pages can be divided into three categories :

- ***dynamic web page***
- ***static web page***
- ***active web page***

A ***dynamic web page*** is a kind of web page that has been prepared with fresh information (content and/or layout), for each individual viewing. It is not static because it changes with the time (ex. a news content), the user (ex. preferences in a login session), the user interaction (ex. web page game), the context (parametric customization), or any combination of the foregoing.

A web page is **static**, if it does not change its behaviour in response to external actions. A static web page remains the same, i.e. static, for all its life, unless and until someone manually changes its contents. Examples of some static web pages are some home pages, page specifying the contact details, etc., which do not change that often.

An **active web page** is a web page that executes a program on the client machine, i.e. web browser. It means the web page shown by a web browser is the program currently executed by the client and hence that page is called an active web page.

Creating a Web Page :

A web page can be created in many different ways. There are a lot of web pages created by simply using HTML code which are simple and not very interactive. Advanced web pages can be created by using a programming language which is also known as scripting languages, such as ***JSP, PHP, Python*** or ***Perl***.

Working of Web Pages:

The information on a web page is displayed online with the help of a web browser, which connects with the server where the website's contents are hosted through the Hypertext Transfer Protocol (HTTP). For instance, if you look at the URL of the web page you are on at the moment, you could notice the prefix 'http://', which tells the browser what protocol to use to execute the particular URL request.

Each web page's contents are usually presented in HTML or XHTML format, which allows for the information to be easily structured and then quickly read by the client's web browser. With the help of CSS (Cascading Style Sheets), designers can precisely control the web page's look and feel as far as the layout typographic elements, color scheme and navigation are concerned. CSS instructions can be either embedded within the HTML web page (valid for that particular page) or can be included in a separate external file (valid for the whole site).

9.4 WEB BROWSERS

In a network such as the World Wide Web, all the computers are divided into two categories : *client* and *server*. A ***web client*** is a software that accesses a web server by sending requests and processing the resulting response.

The Figure 9.1 shows how a web server and a web client is connected to a network and what components are actually held by them for communicating each other.

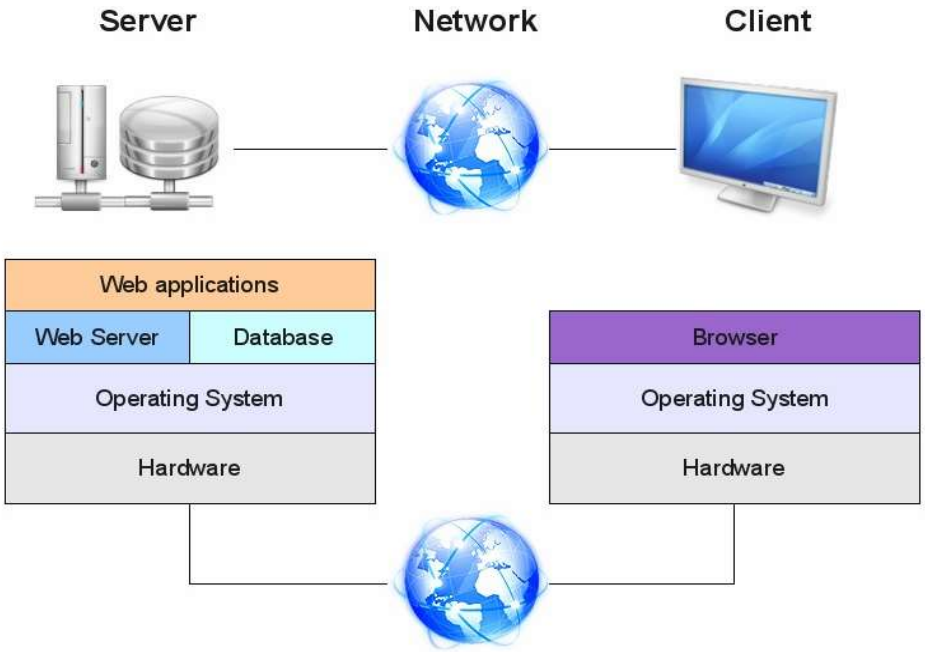


Figure 9.1 : A Web Server and a Client

The web server is the computer on the Internet that stores the web pages, images and other web resources. When we click a hyperlink or type a URL into the web browser, a request for the web page is made by the web client to the web server. The server then receives the request and responds by sending the HTML page to the browser. The browser then interprets and displays the page.

The client-side processing is carried out by the **Web browser**. The browser consists of an interface and processing software. ***Web browsers*** are programs running on the client. They are called browsers because they allow the users to browse the resources available on the servers. We can search for and view different kinds of information on the Web with browsers. The main purpose of a web browser is to make the information resources in the Web available to the users. A browser allows you to view the documents written in HTML (Hyper Text Markup Language). These documents are called *web pages* which we have alrady discussed.

9.5 WEB SERVERS

Server software runs exclusively on server machines, handling the storage and transmission of documents. In contrast, client software such as Netscape, Internet Explorer, etc. runs on client computer that access, translates and displays documents.

A web server is a software package that processes HTML documents for viewing by Web browsers that runs on client computers. Web servers can be run from any hardware platform. There are servers that are specifically designed for Macintosh computers, PCs, Silicon Graphics, and various other platforms, including the Amiga. It can also run under several operating systems, including MS Windows, Windows NT, Unix, Linux etc.

The web server is responsible for document storage and retrieval. It sends the document requested (or an error message) back to the requesting client. The client interprets and presents the document. The client is responsible for document presentation. The language that web clients and servers use to communicate with each other is called the Hypertext Transfer Protocol (HTTP). All web clients and servers must be able to speak HTTP in order to send and receive hypermedia documents. For this reason, web servers are often called HTTP servers, or HTTP Daemons (HTTPD).

9.6 SEARCH ENGINES

The Web is quickly becoming a vast information space that needs search tools to find information efficiently. The search engines can be defined as the tool created for finding, classifying and storing information about various Websites on the Internet. These can help in locating information of relevance on a particular subject by using various search methods. A search engine is a service that indexes, organizes, and often rates and reviews Web sites. Search engines are online utilities that quickly search thousands of Web documents

for an entered word or phrase. Search engines are usually accessed through Web browser software. Each search engine provides different searching options and has its own look.



Figure 9.2 : A list of Search Engines

You may wonder as to how the search engines find the answers to a query so quickly. Actually it is a four-step process and the steps are :

- ***Crawling*** the Web, following links to find pages.
- ***Indexing*** the pages to create an index from every word to every place it occurs.
- ***Ranking*** the pages so that the best ones show up first.
- ***Displaying*** the results in a way that is easy for the user to understand.

Given below is a list of important search engines which you can use for searching the web.

Name	**URL :**
Altavista	http://www.altavista.com
Excite	http://www.excite.com
Hotbot	http://www.hotbot.com
Infoseek	http://www.infoseek.com
Lycos	http://www.lycos.com

Northern Light	http://www.northernlight.com
Webcrawler	http://www.webcrawler.com
Yahoo	http://www.yahoo.com
Google	http://www.google.com

9.7 WEB COOKIES

Basically, the cookies stores tracking data about your browsing. Cookies is a message given to a Web browser by a Web server. The browser stores the message in a text file in the client machine and uses each time the browser requests a page from the server. Web cookies were also called "magic cookies" when they were first introduced. When you enter a Web site using cookies, you may be asked to fill out a form providing such information as your name and interest.

If you use Microsoft Internet Explorer, your cookies are kept as individual files in a folder named "Cookies", often found in the "Documents and Settings" folder. If you use Mozilla Firefox, your cookies are stored in a text file named "cookies", often found in the "Firefox/Profiles" folder.

Some of the reasons why web sites use cookies are described below:

Customization : Some sites use cookies to record your surfing patterns, and then optimize the information the site subsequently presents. For example, a search site may present you with advertising that reflects your interests based on the keywords you search for.

Distribution : One of the key reasons for which web sites use cookies is to distribute the information storage. A cookie takes up a small amount of space on an individual computer, but would take up a very large amount of space if they all had to be stored back on a web site's server.

Security : If a cookie was stored on a web site, then it can be accessed by anyone with access to that web site. However, when the cookie is stored on your computer, then it can't be accessed by hackers that break into the web site.



CHECK YOUR PROGRESS

Q.1. Distinguish between web server and web browser.

.....

.....

.....

.....

Q.2. Distinguish between Internet and WWW.

.....

.....

9.8 TECHNOLOGIES IN WWW

The Web uses the HTTP protocol, which governs the dialog between the client and the server. Now, if we consider the technologies used in WWW then we can divide it into two categories like : *technologies in client side* and *technologies in the server side*. Client-side technologies are the programs that operate in the browser. There is no need to interact with the server. These programs are created by using the languages known as ***client side scripting language*** which are generally very easy to use, and you can try them on your own computer. **HTML** is a basic mark-up language that you will use to create the structure of your web page so HTML is one of the client side language. Remember that HTML is not a scripting language which is a basic building block of static web page.

A ***script*** is a program that is executed by the web server in response to a request. The SCRIPT element includes a script in the HTML document. Scripts allow greater interactivity in a document by responding to the user events. A scripting language is a lightweight programming language. *VBScript, Java Script* are examples of scripting languages.

The client-side scripts are executed by the browser or the client. Their work is confined only to the browser. They donot have to wait for the server to process and return data. This makes the web pages more responsive. The browser should also support the scripts that the web page wants to display.

The following applications are used to form server-side technology.

PHP : PHP is a fast, server-side scripting language that is used to create interactive, dynamic web sites.

JSP : Java Server Pages or JSP for short is Sun's solution for developing dynamic web sites. JSP provides excellent server side scripting support for creating database driven web applications.

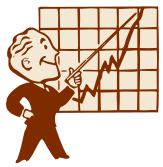
CGI/Perl : Perl is a programming language that can handle input and output from a Web Server, usually through the Common Gateway Interface.

XML/XSL : XML is a software and hardware independent markup language designed for describing and transmitting information. XSL is a language for defining, transforming and formatting XML documents.

MySQL : MySQL is a fast, open-source Relational Database Management System that uses the popular Structured Query Language (SQL).

PostgreSQL : PostgreSQL is a more sophisticated open-source Relational Database Management System. It is well-suited to sites that require robust database functionality, such as e-commerce sites.

Linux/Apache : Linux is a popular open-source operating system, and Apache is the most widely-used web server on the Internet. Together they provide a fast, extremely reliable, secure platform for web sites.



CHECK YOUR PROGRESS

Q.3. Fill in the blanks :

i. A _____ is a program that is executed by the web server in response to a request.

ii. The client-side scripts are executed by the _____.

iii. Web _____ allow a Web site to store information about you as well as to track your visits to it over a prolonged period of time.

Q.4. Choose the appropriate option:

i) Which of the following protocols is responsible for Internet mail?

a) TCP/IP b) HTTP c) FTP

d) None of these

ii) A user can get files from another computer on the Internet by using

a) FTP b) HTTP c) UTP

d) None of these

iii) Take the odd item out

a) Internet Explorer b) Mozilla Firefox

c) Google d) Netscape navigator

iv) Take the odd item out

a) PHP b) JSP c) HTML d) XML

v) FTP is used for

a) Uploading files only b) Downloading files only

c) Both (a) and (b) d) None of these



9.9 LET US SUM UP

It is important to know the various information resources and services available in Internet or World Wide Web (www). It is equally important to know about search engines, selecting and searching from a particular search engine. Client side technologies includes HTML, DHTML, VB Script, Java Script,ECMAScript, JScript, Java applets and Flash Animations etc. Server side technologies includes PHP, JSP, CGI/Perl, XML/XSL, MYSQL, PostgreSQL etc.



9.10ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : A *Web server is a computer* that delivers (serves up) *Web pages.* Every Web server has an *IP address* and possibly a *domain name.* For example, if you enter the *URL http:/www.pcwebopedia.com/index.html* in your browser, this sends a request to the server whose domain name is *pcwebopedia.com.* The server then fetches the page named index.html and sends it to your browser. *Web browser,* a *software application* used to locate and display Web pages which is found on a client computer. The main function of the Web browser is to make the information resource available to the users. The two most popular browsers are *Netscape Navigator* and Microsoft *Internet Explorer.* Both of these are graphical browsers, which means that they can display graphics as well as text.

Ans. to Q. No. 2 : The terms Internet and World Wide Web are often used in everyday speech without much distinction. However, the Internet and the World Wide Web are not one and the same. The Internet is a global data communications system. It is a hardware and software infrastructure that provides connectivity between computers. In contrast, the Web is one of the services communicated

via the Internet. It is a collection of interconnected documents and other resources, linked by hyperlinks and URLs.

Ans. to Q. No. 3 : i) script, ii) browser, iii) cookies

Ans. to Q. No. 4 : i) a, ii) a, iii) c, iv) c, v) c



9.11 FURTHER READING

- Data and Computer Communications by William Stallings, Pearson Education.
- Data Communications and networking by Behrouz A. Forouzan, Tata McGrawHill.
- *"*Web technologies: a computer science perspective*", Jeffrey C. Jack son*, Pearson Prentice Hall.
- *"*Web Technology : A Developer's Perspective*", Gopalan, Gopalan/ akilandeswari*, Pearson Prentice Hall.



9.12 MODEL QUESTIONS

Q.1. What is WWW? How does it function? How is it different from Internet?

Q.2. Define the term WWW ? What are the various features of WWW ?

Q.3. Define the terms Web site, Web page, Web server, URL and home age.

Q.4. Discuss the functioning of a web server and web browser. Also give two examples of a web server and web browser.

Q.5. What is Web server application ? How is it different from a Web browser?

Q.6. Describe how the Web browser communicates with the web server.

Q.15. Write short notes on the following :

a) WWW b) Web servers

c) Web browsers d) Search Engines

****

UNIT 10 : INTRODUCTION TO HTML

UNIT STRUCTURE

10.1 Learning Objectives

10.2 Introduction

10.3 HTML Basics

- 10.3.1 Syntax of HTML
- 10.3.2 Elements and tags in HTML
- 10.3.3 Attributes in HTML
- 10.3.4 HTML editors

10.4 Basic and Formatting Tags

- 10.4.1 Basic tags
- 10.4.2 Formatting text
- 10.4.3 Changing font size and color

10.5 Hyperlink and Image Tags

- 10.5.1 Creating hyperlinks
- 10.5.2 Adding images to web pages

10.6 Frame Tags

- 10.6.1 Creating frames using HTML tags

10.7 Let Us Sum Up

10.8 Further Reading

10.9 Answers to Check Your Progress

10.10 Model Questions

10.1 LEARNING OBJECTIVES

After going through this unit, learners will be able to

- understand the syntax of HTML, including the basic structure and elements of HTML code.
- differentiate between HTML elements and tags, and understand their significance in creating web pages.
- utilize basic HTML tags such as headings, paragraphs, and line breaks to structure and format text on web pages.

- format text using HTML tags such as bold, italic, and underline, and modify font size and color.
- create ordered and unordered lists, and create nested lists using HTML tags.
- create hyperlinks to other web pages or resources using the appropriate HTML tags.
- understand the concept of frames in HTML and create frames using HTML tags.

10.2 INTRODUCTION

HTML (Hypertext Markup Language) is the foundation of web development, serving as the standard markup language for creating web pages. In this context, it is essential to understand the basic syntax of HTML, including its elements, tags, and attributes. HTML editors are tools that facilitate the creation of web pages by providing an interface for writing and designing HTML code. The basic and formatting tags, such as headings, paragraphs, and text formatting, help structure and style the content of web pages. Additionally, list tags, hyperlink, image, and table tags, as well as frames and form tags, are crucial in creating engaging and interactive web pages. Understanding the various form elements, such as input fields, buttons, and dropdown lists, is essential in designing web forms for user input. Overall, HTML is a fundamental building block in web development, and having a good understanding of its syntax and various tags is crucial in creating effective and visually appealing web pages.

10.3 HTML BASICS

The syntax of HTML (Hypertext Markup Language) consists of a set of rules and conventions for defining the structure and content of web pages. It is based on the use of various elements and tags that are used to enclose content and provide additional information about it.

10.3.1 Syntax of HTML

At its most basic level, an HTML document is enclosed within the <html> tag, with the document divided into two main sections: the <head> section, which contains information about the document such as the page title and links to external stylesheets and scripts, and the <body> section, which contains the visible content of the web page.

A basic HTML document consists of several elements that provide information about the document and define the structure and content of the web page. Here's an example of a simple HTML document:

```
<!DOCTYPE html>
<html>
<head>
<title>My Web Page</title>
</head>
<body>
<h1>Welcome to KKHSOU</h1>
<p>This is a paragraph of text.</p>
</body>
</html>
```

Let's break down the structure of this document and explain the purpose of each element:

<!DOCTYPE html>: This is the document type declaration and tells the browser that this is an HTML document. It is not an HTML tag, and it doesn't have a closing tag.

<html>: This is the root element of the HTML document and contains all the other elements. It has an opening tag <html> and a closing tag </html>.

<head>: This element contains information about the document, such as the page title, metadata, and links to stylesheets and scripts. It has an opening tag <head> and a closing tag </head>.

<title>: This element sets the title of the document, which appears in the browser's title bar and in search engine results. It has an opening tag <title> and a closing tag </title>.

<body>: This element contains the visible content of the web page, such as headings, paragraphs, images, and links. It has an opening tag <body> and a closing tag </body>.

<h1>: Welcome to KKHSOU indicates the main heading of the web page. It has an opening tag <h1> and a closing tag </h1>.

: This is a paragraph element and contains text that is displayed as a block of text on the web page. It has an opening tag and a closing tag .

Overall, this example HTML code provides a basic structure for a web page, with a document type declaration, an HTML root element, and separate head and body sections. Within the body section, we have two elements that define the content of the web page: a main heading and a paragraph of text.

In the next sections of the unit, we will discuss the meaning of elements, tags and attributes.

10.3.2 Elements and tags in HTML

HTML (Hypertext Markup Language) uses a combination of elements and tags to define the structure and content of a web page.

An element is a component of an HTML document that defines the content, structure, or formatting of the page. Elements are represented by tags, which are enclosed in angle brackets (<>). A tag typically has an opening tag and a closing tag, with the content to be enclosed in between.

For example, the tag is used to define a paragraph element. The opening tag is and the closing tag is . The paragraph text is placed between the opening and closing tags like this:

This is a paragraph.

10.3.3 Attributes in HTML

HTML tags can also have attributes, which provide additional information about the element. Attributes are placed inside the opening tag, and they are written in the form of name-value pairs, separated by an equals sign.

For example, the <img> tag is used to embed an image in an HTML document. It has an attribute called src, which specifies the URL of the image file. Here's an example:

<imgsrc="image.jpg" alt="A beautiful landscape">

In this example, the src attribute is set to "image.jpg" and the alt attribute is set to "A beautiful landscape".

Overall, HTML elements and tags provide a structured way to define the content and layout of a web page, allowing web developers to create rich, interactive, and accessible websites.

10.3.4 HTML editors

An HTML editor is a software program or tool that allows you to create, edit, and manage HTML code for creating web pages. There are various types of HTML editors available, ranging from basic text editors to advanced graphical editors with drag-and-drop interfaces. Here are some examples of HTML editors:

- Notepad (Windows): A simple text editor that comes with every Windows computer. It is basic, but can be used for creating and editing HTML code.
- Adobe Dreamweaver: A comprehensive web design software that allows you to create, edit, and manage HTML, CSS, and JavaScript code visually. It also provides a variety of pre-designed templates and tools for web development.
- WYSIWYG editors: These are graphical editors that allow you to create web pages visually, without having to write any code. Some popular WYSIWYG editors include Adobe Muse, Weebly, and Squarespace.

10.4 BASIC AND FORMATTING TAGS

10.4.1 Basic tags

Here are some basic HTML tags:

<html>: This tag is used to define the start and end of an HTML document.

<head>: This tag is used to define the head section of an HTML document, which contains information about the document such as the title, meta data, and links to stylesheets.

<title>: This tag is used to define the title of the document, which appears in the browser's title bar and as the page's name in search engine results.

<body>: This tag is used to define the body section of an HTML document, which contains the content that is displayed in the browser.

10.4.2 Formatting text

Here are some common HTML formatting tags:

** Tag:**

The HTML tag is used to define bold text in a document. When you apply the tag to text, it will make the text appear in a bold font. Here is an example of how to use the tag in HTML:

This is a boldtext.

Output:

This is a **bold** text.

In the example above, the word "bold" will be displayed in a bold font when the HTML is rendered in a web browser. The tag is often used for emphasis or to highlight important words or phrases in a document.

<i> Tag:

The HTML <i> tag is used to define a section of text that is to be displayed in italic format. The <i> tag is a part of the HTML formatting

tags that can be used to add emphasis or importance to certain words or phrases in a document.

Here is an example of how to use the tag in HTML:

This is a paragraph of text.

In the example above, the word “paragraph” will be displayed in italic format because it is wrapped in the tag.

<u> Tag:

The <u> tag is used to underline text in HTML. Here is an example:

This text should be <u>underlined</u>.

Output:

This text should be underlined.

 and : This tag is used to emphasize text.

** Tag:**

The tag is an HTML tag used to indicate strong importance or emphasis of text content. It typically renders the text in a bold font. It is commonly used to highlight important keywords, phrases, or sentences within a paragraph or section of content, and can also be used to indicate headings or titles.

Here is an example of how the tag might be used:

Welcome to KKHSOU, the premier Open University in Northeast India.

Output:

Welcome to **KKHSOU**, the premier Open University in Northeast India.

In this example, the text “KKHSOU” is enclosed within the tag, which is used to indicate that the text should be displayed in a bold font. The tag is used to define a paragraph, and the text “Welcome to” and “, the premier Open University in Northeast India.” are also part of the same paragraph.

<sup> Tag:

The <sup> tag is an HTML element used to denote superscript text, which is text that is smaller and raised above the surrounding text. This tag is often used to indicate footnotes, mathematical exponents,

and other types of annotations or references. The <sup> tag is a paired tag, meaning that it has both an opening and a closing tag. Any text between the opening and closing tags will be displayed as superscript. Here is an example of how the <sup> tag can be used:

The equation for the area of a circle is A = πr².

Output:

The equation for the area of a circle is $A = \pi r^2$.

This will display the equation for the area of a circle with the exponent "2" in superscript.

<sub> Tag:

The <sub> tag is an HTML tag used to represent subscript text. When you surround a piece of text with _{and} tags, it will be displayed as a smaller font size and lowered below the normal text baseline.

Here's an example of the <sub> tag in HTML:

The chemical formula for water is H₂O.

Output:

The chemical formula for water is H_2O

The <sub> tag is often used to represent chemical formulas, mathematical expressions, and footnotes in web pages.

<strike> and </strike>: This tag is used to strike through text.

<H> tag:

In HTML, the "h" tags are used to create headings for your web page. They are important for structuring the content of your page and making it more readable and understandable for users.

There are six different heading tags in HTML, from h1 to h6, with h1 being the most important and largest and h6 being the smallest. Here are some examples of how to use these tags:

```
<!DOCTYPE html>
<html>
<head>
    <title>My Web Page</title>
</head>
```

```
<body>
    <h1>This is a Level 1 Heading</h1>
    <h2>This is a Level 2 Heading</h2>
    <h3>This is a Level 3 Heading</h3>
    <h4>This is a Level 4 Heading</h4>
    <h5>This is a Level 5 Heading</h5>
    <h6>This is a Level 6 Heading</h6>
</html>
```

In this example, we have used all six heading tags to create headings of different sizes and levels of importance. It's important to note that the choice of which heading tag to use depends on the hierarchy and importance of the content on your page.

Using heading tags also has benefits for search engine optimization (SEO) as search engines use them to understand the structure and content of your page. It's recommended to use only one h1 tag per page, as it indicates the main heading and title of the page. The rest of the heading tags can be used multiple times on the page to create subheadings and organize the content.

<P> Tag:

The <p> tag in HTML is used to define a paragraph of text. It is a block-level element, which means that it creates a new line before and after the paragraph. Here is an example of how to use the <p> tag with the text "Welcome to KKHSOU":

```
<p>Welcome to KKHSOU</p>
```

In this example, the text "Welcome to KKHSOU" is enclosed in the <p> tags. When the HTML document is rendered in a web browser, the text will be displayed as a paragraph, with a line break before and after the text.

**
 Tag:**

The <br> tag is an HTML element used to create a line break or a new line in the text. Here is an example of how to use the <br> tag:

```
<p>This is the first line.<br>
This is the second line.</p>
```

In the example above, the <br> tag is used to create a line break between “This is the first line.” and “This is the second line.”, resulting in the two lines being displayed on separate lines when rendered in a web browser.

Output:

This is the first line.

This is the second line.

<hr> Tag:

The <hr> tag is an HTML element that is used to create a horizontal line on a web page. Here is an example of how to use it:

```
<p>This is some text.</p>
<hr>
<p>This is some more text.</p>
```

In the example above, the <hr> tag creates a horizontal line between the two paragraphs of text. The line extends the full width of the container element, creating a visual separation between the two sections of text.

Output:

This is some text.

This is some more text.

<blockquote>Tag:

The <blockquote> tag is an HTML element that is used to indicate a block of quoted text. It is often used to highlight a quote or excerpt from another source within a web page. The <blockquote> tag is typically used in combination with the <cite> tag to indicate the source of the quote.

Here’s an example of how the <blockquote> tag can be used in HTML:

```
<blockquote>
“The greatest glory in living lies not in never falling, but in
rising every time we fall.”
</blockquote>
<cite>-Nelson Mandela</cite>
```

This would display the following on the web page:

“The greatest glory in living lies not in never falling, but in rising every time we fall.”

-Nelson Mandela

<code> Tag:

The <code> tag is an HTML element used to indicate a section of code or computer program within a document. It can be used to distinguish code snippets from other text on a web page and to help with readability and organization.

Here’s an example of how the <code> tag can be used in HTML:

```
<p>To calculate the sum of two numbers in Python, you can use the following code:</p>
<code>
x = 5
y = 7
sum = x + y
print(“The sum of”, x, “and”, y, “is”, sum)
</code>
```

This would display the following on the web page:

To calculate the sum of two numbers in Python, you can use the following code:

```
x = 5
y = 7
sum = x + y
print("The sum of", x, "and", y, "is", sum)
```

In this example, the <code> tag is used to indicate the section of code that calculates the sum of two numbers in Python. The text inside the <code> tag is typically displayed in a monospace font and may be styled differently from the rest of the text on the web page to help distinguish it as code.

<pre> Tag:

The <pre> tag in HTML is used to define preformatted text. This means that any text within the <pre> tag will be displayed exactly as it appears in the HTML code, including whitespace and line breaks.

Here's an example of how to use the <pre> tag:

```
<!DOCTYPE html>
<html>
<head>
    <title>Preformatted Text Example</title>
</head>
<body>
    <pre>
        This is an example of preformatted text.
        Welcome to Krishna KantaHandiqui State Open
        University.
    </pre>
</body>
</html>
```

When this code is rendered in a web browser, the text within the <pre> tag will be displayed as:

```
This is an example of preformatted text.

Welcome to Krishna Kanta Handiqui State Open University.
```



CHECK YOUR PROGRESS

1. Fiil in the blanks:

(a) The HTML _____ tag is used to define the title of a web page.

(b) The HTML _____ tag is used to create a hyperlink to another web page.

(c) The HTML _____ tag is used to create a line break in text.

(d) The HTML _____ tag is used to create a heading in a web page.

(e) The HTML _____ tag is used to display an image on a web page.

(f) The HTML _____ tag is used to create a numbered list.

(g) The HTML _____ tag is used to create an unordered list.

(h) The HTML _____ tag is used to define a paragraph of text.

10.4.3 Changing font size and color

The <font> tag in HTML is used to define the font, size, and color of text. However, this tag is now obsolete and has been deprecated in HTML5. The recommended way to style text in HTML is to use CSS.

Here's an example of how to use the <font> tag:

```
<!DOCTYPE html>
<html>
<head>
    <title>Font Example</title>
</head>
<body>
    <font color="red" size="4">This text is red and size 4.</font>
    <font face="Arial">This text is in Arial font.</font>
</body>
</html>
```

When this code is rendered in a web browser, the first line of text within the <font> tag will be displayed in red and size 4, while the second line of text within the <font> tag will be displayed in Arial font.

Again, it is important to note that the <font> tag is no longer recommended for styling text in HTML, and instead, CSS should be used for this purpose.

Here's an example of how to use CSS to style text:

```
<!DOCTYPE html>
<html>
<head>
    <title>CSS Text Styling Example</title>
    <style>
        /* Define a class for the styled text */
        .styled-text {
            font-size: 18px;
            font-family: Arial, sans-serif;
```

```
                color: #333;
                font-weight: bold;
                text-decoration: underline;
            }
        </style>
</head>
<body>
        <p>This is some normal text.</p>
        <p class="styled-text">This is some styled text.</p>
</body>
</html>
```

In this example, a class called "styled-text" is defined in the CSS style block. The class specifies a font size of 18 pixels, a font family of Arial or a sans-serif font, a color of #333 (a shade of gray), bold font weight, and underlined text decoration.

Then, in the HTML body, a paragraph element is defined with the class "styled-text". This paragraph will have the styles defined in the CSS class applied to it, resulting in the text being displayed in the specified font size, font family, color, and with bold and underlined styles.

10.5 HYPERLINK AND IMAGE TAGS

10.5.1 Creating hyperlinks

The <a> tag in HTML stands for "anchor" and is used to create hyperlinks on a webpage. When a user clicks on a hyperlink created with the <a> tag, they are taken to the linked URL.

Here is an example of how to use the <a> tag to create a hyperlink with the text "KKHSOU Website":

```
<a href="http://www.kkhsou.in/">KKHSOU Website</a>
```

The HTML code KKHSOU Website creates a hyperlink to the KKHSOU (Krishna KantaHandiqui State Open University) website.

Here is a breakdown of the code:

- <a> is the HTML tag used to create a hyperlink.
- href="http://www.kkhsou.in/" is an attribute of the <a> tag, which specifies the destination URL of the hyperlink. In this case, the destination URL is the KKHSOU website, which is http://www.kkhsou.in/.
- KKHSOU Website is the visible text of the hyperlink. When this code is rendered in a web browser, the visible text will appear as a clickable link.
- is the closing tag for the <a> tag.

When a user clicks on the link, their web browser will navigate to the KKHSOU website specified in the href attribute.

10.5.2 Adding images to web pages

The HTML <img> tag is used to insert an image into a web page. Here's an example of how to use the <img> tag:

<imgsrc="kamakhya.jpg" alt="Kamakhya Temple">

In this example, the <img> tag has two important attributes:

- src: This attribute specifies the URL of the image file. In this case, the image file is named "image.jpg". You can specify a relative or absolute URL.
- alt: This attribute specifies an alternative text description of the image. This is important for accessibility purposes; as visually impaired users may use screen readers to read the content of a webpage. The text "Kamakhya Temple" is used as the alternative text for this example.

When this code is rendered in a web browser, the image specified in the src attribute will be displayed. If the image cannot be loaded for any reason, the text specified in the alt attribute will be displayed instead.

Note that the <img> tag is a self-closing tag and does not require a closing tag. If you need to specify additional attributes, you can add them before the closing angle bracket, like this:

```
<imgsrc=”kamakhya.jpg” alt=”A beautiful image” width=”500"
height=”300">
```

In this example, two additional attributes are added: width and height. These specify the width and height of the image in pixels.

10.6 FRAME TAGS

10.6.1 Creating frames using HTML tags

The <frame> tag is an HTML element that is used to define a single frame within a frameset. Frames allow multiple web pages to be displayed within a single browser window, with each page being contained within its own frame.

Here is an example of how to use the <frame> tag:

```
<!DOCTYPE html>
<html>
<head>
<title>Frameset Example</title>
</head>
<frameset cols=”25%, 75%”>
<frame src=”menu.html” name=”menu”>
<frame src=”content.html” name=”content”>
</frameset>
<body>
<h1>Welcome to my Frameset Example</h1>
</body>
</html>
```

In this example, the <frameset> element is used to define a frameset that contains two frames, one for the menu and one for the content. The cols attribute is used to specify the size of each frame, where the first frame will take up 25% of the width and the second frame will take up 75% of the width.

The <frame> tag is used to define each frame, where the src attribute specifies the URL of the web page to be displayed within the frame,

and the name attribute provides a name for the frame that can be used to target links or forms within that frame.

The <body> element is included as a fallback for browsers that do not support frames, and will be displayed instead of the frameset in these cases.

Note that the <frame> tag is deprecated in HTML5 and should not be used in new web development projects. Instead, modern web development techniques such as CSS and JavaScript are used to achieve similar effects.



CHECK YOUR PROGRESS

2. Choose the correct option:

(a) What is the recommended way to style text in HTML?

i) <font> tag ii) CSS iii) <style> tag iv) <p> tag

(b) Which attribute of the <img> tag specifies the URL of the image file?

i) src ii) alt iii) href iv) target

(c) What is the purpose of the <a> tag in HTML?

i) To insert an image into a web page

ii) To create a hyperlink on a webpage

iii) To define a single frame within a frameset

iv) To style text in HTML

(d) Which tag is used to create frames within a frameset in HTML?

i) <frame> ii) <img> iii) <a> iv) <style>

(e) What is the purpose of the “alt” attribute in the <img> tag?

i) To specify the destination URL of the hyperlink

ii) To define the font family for text

iii) To specify an alternative text description of the image

iv) To define the width and height of the image



10.7 LET US SUM UP

- HTML (Hypertext Markup Language) is used to define the structure and content of web pages.
- An HTML document has a basic syntax with elements and tags enclosed within the <html> tag.
- The <head> section contains information about the document, while the <body> section contains the visible content of the web page.
- Elements are represented by tags enclosed in angle brackets (<>), and tags typically have an opening and a closing tag with content in between.
- HTML tags can have attributes, which provide additional information about the element, and are written as name-value pairs.
- HTML editors are software programs or tools that allow creating, editing, and managing HTML code for creating web pages.
- Basic HTML tags include <html>, <head>, <title>, <body>, <p>, <h1>, <img>, while formatting tags include <b>, <i>, <u>, , and .
- HTML provides various tags to format and structure web content, such as heading tags, paragraph tags, and list tags.
- Styling text and adding visual elements like images and hyperlinks to web pages is also possible in HTML using tags like <img>, <a>, and <style>.
- The <frame> tag is used to define frames within a frameset, allowing multiple web pages to be displayed within a single browser window. However, the use of frames is now considered outdated and not recommended.



10.8 FURTHER READING

1) https://www.w3schools.com/html/default.asp

2) "HTML5 and CSS3: Develop with Tomorrow's Standards Today" by Brian P. Hogan

3) "HTML & CSS: The Complete Reference" by Thomas Powell and Chuck White



10.9 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q.No.1:

(a) title

(b) a

(c) br

(d) h1, h2, h3, h4, h5, or h6 (any of the heading tags)

(e) img

(f) ol

(g) ul

(h) p

Ans. to Q.No.2:

(a) ii) (b) i) (c) ii) (d) i) (e) iii)



10.10 MODEL QUESTIONS

Q.1: What is HTML, and why is it essential in web development?

Q.2: What are the basic syntax and structure of an HTML document?

Q.3: What are the different types of tags in HTML, and how are they used to structure and format web pages?

Q.4: How do attributes provide additional information about HTML elements, and what are some examples of commonly used attributes?

Q.5: How do HTML editors help web developers create and manage HTML code?

Q.6: What are some of the key elements and tags used in creating engaging and interactive web pages?

Q.7: How do HTML lists, hyperlinks, and images contribute to the content and layout of a web page?

Q.8: Why is understanding HTML syntax and various tags crucial in creating effective and visually appealing web pages?

Q.9: What is the recommended way to style text in HTML, and why is the <font> tag deprecated in HTML5?

Q.10: How do you create a hyperlink in HTML using the <a> tag? What attributes are used to specify the destination URL and visible text of the link?

Q.11: What is the purpose of the <img> tag in HTML, and what attributes can be used to specify the image file, alternative text, and dimensions of the image?

Q.12: What is a frameset in HTML, and how can you use the <frame> tag to create multiple frames within a single browser window? What attributes can be used to specify the size, name, and source of each frame?

****

UNIT 11 : HTML LIST AND TABLES

UNIT STRUCTURE

11.1 Learning Objectives

11.2 Introduction

11.3 HTML List

11.4 Unordered List

11.5 Ordered Lists

11.6 HTML Tables

11.7 Let Us Sum Up

11.8 Further Reading

11.9 Answers to Check Your Progress

11.10 Model Questions

11.1 LEARNING OBJECTIVES

After going through this unit, learners will be able to

- understand the purpose and structure of HTML lists.
- identify the differences between unordered and ordered lists in HTML.
- create an unordered list using the <ul> tag and its attributes.
- create a nested unordered list in HTML.
- create an ordered list using the <ol> tag and its attributes.
- use different numbering styles in an ordered list in HTML.
- create a nested ordered list in HTML.
- understand the purpose and structure of HTML tables.
- create an HTML table using the <table> tag and its attributes.

11.2 INTRODUCTION

Lists and tables are some of the most commonly used elements in HTML for organizing and presenting information on web pages. Lists allow for the grouping of related items, whether ordered or unordered. Tables, on the other hand, allow for the display of tabular data in a structured and organized manner. Understanding how to create and use lists and tables in

HTML is essential for creating well-structured and easy-to-read web pages. This unit will cover the basics of creating unordered and ordered lists, as well as creating and formatting tables in HTML.

11.3 HTML LIST

In HTML, a list is a set of items that are presented in a specific order or without any order. Lists are used to group and organize related pieces of information. HTML provides two types of lists: unordered lists and ordered lists. Unordered lists use bullet points or other symbols to indicate each item, while ordered lists use numbers or letters to indicate the order of each item. Lists are an important element in web design as they help to structure and organize information, making it easier for users to read and understand.Lists are important in web design for several reasons:

- Organization: Lists help to organize related items, making it easier for users to understand the structure of the content.
- Readability: By breaking down content into smaller chunks, lists can improve the readability of text, making it easier to scan and digest.
- Clarity: Lists can provide clarity to complex information by visually separating each item and making it easier to understand.
- Accessibility: Lists can be beneficial for users with disabilities, such as those who use screen readers, by providing a clear and structured format for information.
- Flexibility: Lists can be used in a variety of ways, from simple bullet-point lists to complex nested lists, making them a flexible tool for web designers to structure content on a webpage.

11.4 UNORDERED LIST

The <ul> tag is used to create an unordered list in HTML. Here is the syntax of the <ul> tag:

```
<ul>
<li>Item 1</li>
```

```
<li>Item 2</li>
<li>Item 3</li>
</ul>
```

The <ul> tag must be opened with <ul> and closed with . Each item in the list is represented by the <li> tag. The <li> tag must be nested inside the <ul> tag and closed with .

The <ul> tag also has some optional attributes that can be used to modify the behavior or appearance of the list, including:

- type: Specifies the type of bullet point to use for the list items. Valid values include “disc”, “circle”, and “square”. For example: <ul type="circle”>
- start: Specifies the starting number for the list items. For example: <ul start="3">
- class, id, style, title, lang, dir: These attributes can be used to add CSS styling or other metadata to the list. For example: <ul class=”my-list” id=”list-1" style=”color:red;”>

Note that the use of CSS is typically preferred for styling, rather than using HTML attributes.

The <ul> tag is used to create an unordered list in HTML. The type attribute of the <ul> tag can be used to specify the type of bullet point to use for the list items. Here are some examples of different types of bullet points and their real-life use cases:

(a) Disc bullet points (type="disc”): These are the default bullet points used in unordered lists. They are often used for simple lists of items, such as a grocery list or a list of to-do items.

```
<ul type="disc”>
        <li>Bread</li>
        <li>Milk</li>
        <li>Eggs</li>
</ul>
```

Output:

- Bread
- Milk
- Eggs

(b) Circle bullet points (type="circle"): These bullet points are similar to disc bullet points, but they are hollow instead of filled. They can be used to create a visual distinction between different lists.

```
<ul type="circle">
    <li>Apples</li>
    <li>Oranges</li>
    <li>Bananas</li>
</ul>
```

Output:

- Apples
- Oranges
- Bananas

(c) Square bullet points (type="square"): These bullet points are square in shape and can be used to create a more structured and organized look for a list. They are commonly used for technical lists or product features.

```
<ul type="square">
    <li>High-speed processor</li>
    <li>Large storage capacity</li>
    <li>Long battery life</li>
</ul>
```

Output:

- High-speed processor
- Large storage capacity
- Long battery life

Nested List: Nesting unordered lists in HTML means creating a hierarchy of lists by placing one unordered list inside another unordered list. This is a useful technique for organizing and presenting information in a structured way.

Here's an example of how to nest unordered lists in HTML:

```
<ul>
    <li>Home</li>
    <li>Menu
    <ul>
    <li>Breakfast</li>
    <li>Lunch</li>
    <li>Dinner</li>
    </ul>
    </li>
    <li>About Us</li>
    <li>Contact Us</li>
</ul>
```

This is a common pattern used in website navigation menus, where submenus are nested within main menu items. For example, a website for a restaurant might have a navigation menu like this.

In this menu, "Menu" is a main menu item that contains three subitems ("Breakfast", "Lunch", and "Dinner").

Output:

- Home
- Menu
  - Breakfast
  - Lunch
  - Dinner
- About Us
- Contact Us

11.5 ORDERED LIST

The <ol> tag in HTML stands for "ordered list" and is used to create a numbered list on a web page.

The syntax for the <ol> tag is as follows:

```
<ol>
<li>First item</li>
```

```
<li>Second item</li>
<li>Third item</li>
</ol>
```

The <ol> tag must be opened with <ol> and closed with . Inside the opening and closing <ol> tags, you will need to include one or more <li> (list item) tags to specify each item in the ordered list.

Attributes that can be used with the <ol> tag include:

- type: Specifies the type of numbering used in the list. The possible values are “1” (decimal), “A” (uppercase letters), “a” (lowercase letters), “I” (uppercase Roman numerals), and “i” (lowercase Roman numerals).
- start: Specifies the starting number of the first item in the list.
- reversed: Specifies that the list items should be numbered in reverse order (from high to low).

The <ol> tag in HTML allows you to specify different numbering styles for your ordered list. Here are some examples of how you can use different numbering styles:

(a) Decimal numbering (default):

```
<ol>
<li>Item 1</li>
<li>Item 2</li>
<li>Item 3</li>
</ol>
```

Output:

1. Item 1
2. Item 2
3. Item 3

(b) Uppercase letters:

```
<ol type=”A”>
<li>Item 1</li>
<li>Item 2</li>
<li>Item 3</li>
</ol>
```

Output:

A. Item 1

B. Item 2

C. Item 3

(c) Lowercase letters:

```
<ol type=”a”>
<li>Item 1</li>
<li>Item 2</li>
<li>Item 3</li>
</ol>
```

Output:

a. Item 1

b. Item 2

c. Item 3

(d) Uppercase Roman numerals:

```
<ol type=”I”>
<li>Item 1</li>
<li>Item 2</li>
<li>Item 3</li>
</ol>
```

Output:

I. Item 1

II. Item 2

III. Item 3

(e) Lowercase Roman numerals:

```
<ol type=”i”>
<li>Item 1</li>
<li>Item 2</li>
<li>Item 3</li>
</ol>
```

Output:

i. Item 1

ii. Item 2

iii. Item 3

You can also combine the different numbering styles with the start attribute to begin the list at a specific number or letter. For example:

```
<ol start=”5" type=”A”>
<li>Item 1</li>
<li>Item 2</li>
<li>Item 3</li>
</ol>
```

Output:

E. Item 1
F. Item 2
G. Item 3

Nested List: To create a nested ordered list, you simply place an <ol> element inside an <li> element.

Here’s an example of a nested ordered list in HTML:

```
<ol>
<li>Item 1</li>
<li>Item 2
<ol>
<li>Item 2.1</li>
<li>Item 2.2</li>
</ol>
</li>
<li>Item 3</li>
</ol>
```

In this example, the second item in the list (“Item 2”) contains a nested ordered list with two items (“Item 2.1” and “Item 2.2”).

Output:

1. Item 1
2. Item 2
    1. Item 2.1
    2. Item 2.2
3. Item 3

As you can see, the nested ordered list is indented to the right to indicate that it is a sub-list of the item above it.



CHECK YOUR PROGRESS

1. Fill in the blanks:
   (a) HTML provides two types of lists: _____ lists and _____ lists.
   (b) The _____ tag is used to create an unordered list in HTML.
   (c) The <li> tag must be nested inside the _____ tag when creating an unordered list.
   (d) _____ bullet points are the default bullet points used in unordered lists.
   (e) The <ol> tag in HTML stands for _____ list and is used to create a numbered list on a web page.

11.6 HTML TABLES

The HTML <table> tag is used to create a table on a webpage. It allows you to organize and display data in a structured way. A table is made up of rows and columns, and the <table> tag defines the entire table.

Here is an example of how the <table> tag is used:

```
<table border="1">
<tr>
<th>Header 1</th>
<th>Header 2</th>
```

```
<th>Header 3</th>
</tr>
<tr>
<td>Data 1</td>
<td>Data 2</td>
<td>Data 3</td>
</tr>
<tr>
<td>Data 4</td>
<td>Data 5</td>
<td>Data 6</td>
</tr>
</table>
```

In this example, we have a simple table with two rows and three columns. The first row (<tr>) contains three table headers (<th>) which will appear at the top of each column. The second and third rows contain data cells (<td>) that correspond to each header. The first cell of each row contains data for the first column, the second cell contains data for the second column, and so on.

When rendered in a web browser, this code will display the following table:

Header 1	Header 2	Header 3
Data 1	Data 2	Data 3
Data 4	Data 5	Data 6



11.7 LET US SUM UP

- Lists are important in web design as they help to structure and organize information, making it easier for users to read and understand.
- HTML provides two types of lists: unordered lists and ordered lists.

- Unordered lists use bullet points or other symbols to indicate each item, while ordered lists use numbers or letters to indicate the order of each item.
- The <ul> tag is used to create an unordered list in HTML, and the <ol> tag is used to create an ordered list.
- The <li> tag represents each item in the list, and must be nested inside the <ul> or <ol> tags.
- The <ul> tag has optional attributes like type, start, class, id, style, title, lang, and dir that can be used to modify the behavior or appearance of the list.
- CSS is typically preferred for styling, rather than using HTML attributes.
- Different types of bullet points can be used in unordered lists, such as disc, circle, and square bullet points, to create visual distinctions between lists.
- Nested lists can be created in HTML by placing one unordered list inside another unordered list, which is a useful technique for organizing and presenting information in a structured way.
- Ordered lists are commonly used for lists that need to be presented in a specific order, such as instructions or steps in a process.



11.8 FURTHER READING

1) https://www.w3schools.com/html/default.asp
2) “HTML5 and CSS3: Develop with Tomorrow’s Standards Today” by Brian P. Hogan
3) “HTML & CSS: The Complete Reference” by Thomas Powell and Chuck White

11.9 ANSWERS TO CHECK YOUR PROGRESS

Ans.to.Q.No.1:

(a) unordered, ordered (b) (c) (d) Disc (e) ordered

11.10 MODEL QUESTIONS

Q.1: What is the purpose of using lists in HTML?

Q.2: What are the two types of lists available in HTML, and how do they differ from each other?

Q.3: What is the syntax for creating an unordered list in HTML, and what are some optional attributes that can be used with the <ul> tag?

Q.4: Can you provide examples of different types of bullet points that can be used in an unordered list, and when they are commonly used?

Q.5: How do you nest unordered lists in HTML, and what is an example use case for this technique?

Q.6: What is an ordered list in HTML, and what is the syntax for creating one?

Q.7: What are the components of a table in HTML?

****

UNIT 12 : ADVANCED HTML TAGS

UNIT STRUCTURE

12.1 Learning Objectives

12.2 Introduction

12.3 HTML Forms

12.4 Form Elements and Attributes

12.5 Advanced HTML Tags

- 12.5.1 The <canvas> Tag
- 12.5.2 The <video> and <audio> Tags
- 12.5.3 The <svg> Tag

12.6 Let Us Sum Up

12.7 Answers to Check Your Progress

12.8 Further Reading

12.9 Model Questions

12.1 LEARNING OBJECTIVES

After going through this unit, learners will be able to

- Identify the purpose and function of HTML forms and explain how they are used to collect user input and send it to the server.
- Describe the different types of form elements available in HTML, including <input>, <select>, <textarea>, and <button>, and explain how they are used to create various types of form controls.
- Explain the use of attributes in HTML forms, including the action, method, and enctype attributes, and describe how they affect the behavior of the form.
- Define the <canvas> tag and describe its purpose in HTML, including its use in creating dynamic and interactive graphics.
- Explain the use of the <video> and <audio> tags in HTML, and describe how they are used to embed video and audio content on a web page.

- Define the <svg> tag and explain its purpose in HTML, including its use in creating scalable vector graphics.

12.2 INTRODUCTION

HTML (Hypertext Markup Language) is the standard markup language used to create and design web pages. One of the key features of HTML is the ability to create forms that allow users to input data and submit it to a server for processing. In this unit, we will explore the different form elements and attributes available in HTML, including <input>, <select>, and <textarea>, and how they can be used to create a variety of form controls. We will also examine advanced HTML tags such as <canvas>, <video>, <audio>, and <svg>, and how they can be used to create dynamic and interactive graphics and multimedia content on a web page. By the end of this unit, you will have a solid understanding of how to create and style HTML forms and use advanced HTML tags to enhance the user experience on your web pages.

12.3 HTML FORMS

HTML forms are a crucial component of web development that allow users to input data and interact with web pages. They consist of various form elements such as text fields, checkboxes, radio buttons, dropdown menus, and buttons, which can be used to create different types of form controls. These form elements are defined using HTML tags, such as <input>, <select>, <textarea>, and <button>.

HTML forms also use various attributes to define the behavior of the form, such as the method attribute to define the HTTP method for submitting the form, the action attribute to specify the URL where the form data is sent, and the enctype attribute to specify the encoding type for the form data.

When a user submits a form, the data entered by the user is sent to the server for processing. This allows web developers to create dynamic web applications that can collect user input and provide a customized user experience based on that input.

HTML forms can be styled using CSS to enhance their appearance and make them more user-friendly. Additionally, JavaScript can be used to add functionality to forms, such as form validation, autocomplete, and dynamic form elements that change based on user input.

12.4 FORM ELEMENTS AND ATTRIBUTES

Here are the most commonly used elements within the HTML <form> tag:

<label>: The <label> tag in HTML is used to associate a label with an input element. The purpose of this tag is to provide a description or a caption for an input element so that users can easily understand the purpose of the input field. Here's an example of how to use the <label> tag:

```
<label for="username">Username:</label>
<input type="text" id="username" name="username">
```

In this example, we have created a label for an input field that accepts a username. The for attribute in the <label> tag is used to associate the label with the input element. The value of the for attribute should match the id attribute of the input element.

<input>: This element is used to create various types of form elements such as text fields, radio buttons, checkboxes, buttons, and more.

(a) Text Input:

The following example creates a text input field for users to enter their name:

```
<label for="name">Name:</label>
<input type="text" id="name" name="name">
```

In this example, the type attribute is set to text to create a text input field. The id and name attributes are used to identify the field for styling and server-side processing.

(b) Checkbox:

The following example creates a checkbox for users to select their preferred color:

```
<label for="color">Favorite Color:</label>
<input type="checkbox" id="color" name="color" value="red"> Red
```

```
<input type="checkbox" id="color" name="color" value="green">
Green
<input type="checkbox" id="color" name="color" value="blue"> Blue
```

In this example, the type attribute is set to checkbox to create a checkbox input field. The id and name attributes are used to identify the field, and the value attribute is used to set the value of each checkbox.

(c) Radio Buttons:

The following example creates a group of radio buttons for users to select their preferred gender:

```
<label for="gender">Gender:</label>
<input type="radio" id="gender" name="gender" value="male"> Male
<input type="radio" id="gender" name="gender" value="female">
Female
```

In this example, the type attribute is set to radio to create radio buttons. The name attribute is used to group the radio buttons together, so only one option can be selected at a time.

(d) Submit Button:

The following example creates a submit button that the user can click to submit the form:

```
<input type="submit" value="Submit">
```

In this example, the type attribute is set to submit to create a submit button. The value attribute is used to set the text that appears on the button.

(e) Reset Button:

The <input type="reset"> tag in HTML is used to create a reset button in a form. When a user clicks on this button, all the form fields are reset to their default values, which is typically empty.

Here's an example of how to use the <input type="reset"> tag:

```
<input type="reset" value="Reset">
```

(f) File Upload:

The following example creates a file upload field for users to upload a file:

```
<label for="file">Upload File:</label>
<input type="file" id="file" name="file">
```

In this example, the type attribute is set to file to create a file upload field. The id and name attributes are used to identify the field.

These are just some examples of how to use the <input> tag to create different types of form elements. There are many other attributes and values that can be used to customize the appearance and behavior of form elements, depending on your needs.

(g) <textarea>: The <textarea> tag is used to create a multi-line text input field in HTML. It is commonly used in forms where users are expected to enter more than one line of text, such as a comment or a message.

Here's an example of how to use the <textarea> tag:

```
<form>
<label for="message">Message:</label>
<textarea id="message" name="message" rows="5" cols="30"></
textarea>
</form>
```

In this example, we have created a form with a label and a <textarea> input field. The for attribute in the label is used to associate the label with the <textarea> input field. The id attribute in the <textarea> tag is used to uniquely identify the input field, while the name attribute specifies the name of the input field when the form is submitted.

The rows and cols attributes in the <textarea> tag specify the number of rows and columns, respectively, that should be visible in the input field. In this example, we have specified rows="5" and cols="30", which means that the input field will be 5 rows high and 30 characters wide.

Here's another example that shows how to pre-fill the input field with some text:

```
<form>
<label for="message">Message:</label>
<textarea id="message" name="message" rows="5"
cols="30">Enter your message here...</textarea>
</form>
```

In this example, we have added some default text inside the <textarea> input field using the content between the opening and closing <textarea> tags. This text will be automatically selected when the user clicks inside the input field, so they can start typing right away without having to delete the default text first.

Overall, the <textarea> tag is a simple but powerful HTML element that makes it easy to create multi-line text input fields in web forms.

(h) <select>: This element is used to create a drop-down list of options that the user can choose from.

The <select> tag in HTML is used to create a drop-down list of options that users can select from. This is often used in forms where users are required to choose from a pre-defined list of options, such as selecting a country, a language, or a category.

Here's an example of how to use the <select> tag:

```
<label for="color">Select your favorite color:</label>
<select id="color" name="color">
<option value="red">Red</option>
<option value="green">Green</option>
<option value="blue">Blue</option>
<option value="yellow">Yellow</option>
</select>
```

In this example, we have created a drop-down list of colors using the <select> tag. The id attribute is used to uniquely identify the drop-down list, while the name attribute specifies the name of the input field when the form is submitted.

The <option> tag is used to create each individual option in the drop-down list. The value attribute in the <option> tag specifies the value that will be submitted when the form is submitted, while the content between the opening and closing <option> tags specifies the text that will be displayed in the drop-down list.

Here's another example that shows how to pre-select an option in the drop-down list:

```
<label for="fruit">Select your favorite fruit:</label>
<select id="fruit" name="fruit">
    <option value="apple">Apple</option>
    <option value="banana" selected>Banana</option>
    <option value="orange">Orange</option>
    <option value="strawberry">Strawberry</option>
</select>
```

In this example, we have used the selected attribute in the <option> tag to pre-select the "Banana" option in the drop-down list. This means that when the page loads, the "Banana" option will be displayed as the default selection in the drop-down list.

Overall, the <select> tag is a simple but powerful HTML element that makes it easy to create drop-down lists of options in web forms. It is a best practice to always use the <select> tag to provide users with a clear and concise set of options to choose from in your web forms.

(i) <button>: This element is used to create a clickable button that can be used to submit a form or perform some other action.

Here's an example of how to use the <button> tag:

```
<button type="submit" name="submit">Submit</button>
```

In this example, we have created a button that will submit the form when clicked. The type attribute is set to "submit" to indicate that this button should be used for submitting the form. The name attribute is used to specify the name of the input field when the form is submitted.

(j) <legend> and <fieldset>: The <legend> and <fieldset> tags in HTML are used to group related form elements and provide a descriptive caption for the group. This is useful for creating more organized and accessible forms, especially when there are multiple form fields that are related to each other.

The <fieldset> tag is used to group related form elements together. Here's an example of how to use the <fieldset> tag:

```
<fieldset>
    <legend>Personal Information</legend>
    <label for=”name”>Name:</label>
    <input type=”text” id=”name” name=”name”><br>
    <label for=”email”>Email:</label>
    <input type=”email” id=”email” name=”email”><br>
    <label for=”phone”>Phone:</label>
    <input type=”tel” id=”phone” name=”phone”><br>
</fieldset>
```

In this example, we have created a fieldset that groups together three form fields for the user's name, email, and phone number. The <legend> tag is used to provide a descriptive caption for the group, in this case "Personal Information". The fieldset creates a border around the related form fields, making it clear to the user that these fields are related.

The <legend> tag is used to provide a caption or title for the group of related form elements. Here's an example of how to use the <legend> tag:

```
<fieldset>
    <legend>Shipping Address</legend>
    <label for=”address”>Address:</label>
    <input type=”text” id=”address” name=”address”><br>
    <label for=”city”>City:</label>
    <input type=”text” id=”city” name=”city”><br>
    <label for=”state”>State:</label>
    <input type=”text” id=”state” name=”state”><br>
    <label for=”zip”>Zip Code:</label>
    <input type=”text” id=”zip” name=”zip”><br>
</fieldset>
```

In this example, we have used the <legend> tag to provide a descriptive caption for a group of related form fields that are used to collect the user's shipping address.

Overall, the <fieldset> and <legend> tags are useful for organizing and structuring forms in a way that makes it easier for users to understand the relationship between different form fields. They also help to improve the

accessibility of forms by providing clear and descriptive captions for groups of related form fields.

These are just some of the most commonly used elements within the <form> tag, but there are many others that can be used to create more complex forms with additional functionality.

A Complete HTML Form Example:

Here's an example of a complete HTML form that includes all the basic form elements:

```
<form action="/submit-form" method="POST">
<fieldset>
<legend>Contact Information</legend>
<label for="name">Name:</label>
<input type="text" id="name" name="name" required><br>
<label for="email">Email:</label>
<input type="email" id="email" name="email" required><br>
<label for="phone">Phone:</label>
<input type="tel" id="phone" name="phone"><br>
</fieldset>
<fieldset>
<legend>Order Information</legend>
<label for="product">Product:</label>
<select id="product" name="product" required>
<option value="">— Please select —</option>
<option value="product1">Product 1</option>
<option value="product2">Product 2</option>
<option value="product3">Product 3</option>
</select><br>
<label for="quantity">Quantity:</label>
<input type="number" id="quantity" name="quantity" min="1"
required><br>
<label for="delivery">Delivery date:</label>
<input type="date" id="delivery" name="delivery" required><br>
</fieldset>
```

```
<fieldset>
<legend>Comments</legend>
<textarea id=”comments” name=”comments” rows=”4"
cols=”50"></textarea><br>
</fieldset>
<fieldset>
<legend>Options</legend>
<label for=”gift-wrap”>Gift wrap:</label>
<input type=”checkbox” id=”gift-wrap” name=”gift-wrap”
value=”yes”><br>
<label for=”newsletter”>Subscribe to our newsletter:</label>
<input type=”checkbox” id=”newsletter” name=”newsletter”
value=”yes”><br>
</fieldset>
<input type=”submit” value=”Submit”>
<input type=”reset” value=”Reset”>
</form>
```

This is an HTML form with different fields to collect information from the user.

The form has four fieldsets, each with a legend to describe the group of fields they contain:

- Contact Information: This fieldset contains the user’s name, email address, and phone number.
- Order Information: This fieldset contains the product selection, quantity, and delivery date.
- Comments: This fieldset contains a textarea where the user can enter comments.
- Options: This fieldset contains checkboxes to indicate whether the user wants gift wrapping and/or wants to subscribe to the newsletter.

At the bottom of the form, there are two buttons: “Submit” and “Reset.” The “Submit” button will send the form data to the server specified in the form’s action attribute, using the HTTP POST method. The “Reset” button will clear all the form fields.

Output:

Contact Information
Name:
Email:
Phone:
Order Information
Product: -- Please select --
Quantity:
Delivery date: dd-mm-yyyy
Comments
Options
Gift wrap:
Subscribe to our newsletter:
Submit Reset



CHECK YOUR PROGRESS

Q.1. What is the purpose of HTML forms?

Q.2. What is the purpose of the <label> tag in HTML forms?

Q.3. What is the purpose of the <input type="submit"> tag in HTML forms?

Q.4. What is the purpose of the <textarea> tag in HTML forms?

12.5 ADVANCED HTML TAGS

12.5.1 The <canvas> Tag

The <canvas> tag is an HTML5 element used to create dynamic, interactive graphics and animations within a web page. It provides a drawing area for JavaScript code, which can be used to create 2D or 3D graphics, animations, and even games.

Here's an example of how to use the <canvas> tag:

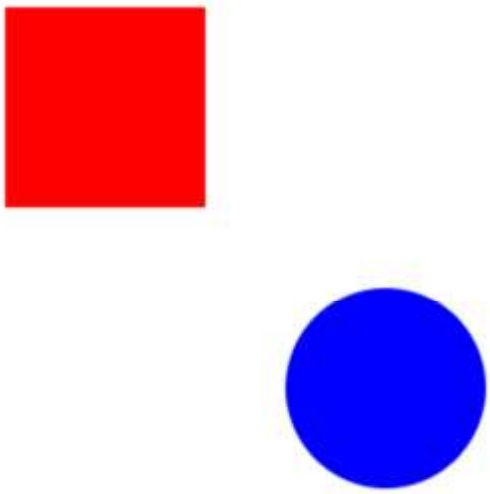
```
<!DOCTYPE html>
<html>
```

```
<head>
<title>Canvas Example</title>
</head>
<body>
<canvas id=”myCanvas” width=”400" height=”400"></canvas>
<script>
    var canvas = document.getElementById(“myCanvas”);
     var ctx = canvas.getContext(“2d”);
     // draw a red rectangle
     ctx.fillStyle = “red”;
     ctx.fillRect(10, 10, 100, 100);
     // draw a blue circle
     ctx.beginPath();
     ctx.fillStyle = “blue”;
     ctx.arc(200, 200, 50, 0, 2 * Math.PI);
     ctx.fill();
</script>
</body>
</html>
```

In this example, we create a canvas element with an id of “myCanvas” and a width and height of 400 pixels each. We then use JavaScript to get a reference to the canvas element and create a 2D drawing context. Finally, we use the context to draw a red rectangle and a blue circle on the canvas.

The <canvas> tag provides a powerful and flexible way to create dynamic graphics and animations on the web, and can be used in combination with other web technologies like CSS and JavaScript to create stunning visual effects.

Output:



12.5.2 The <video> and <audio> Tags

(a) <AUDIO> Tag: The HTML <audio> tag is used to embed audio content into a web page. Here's an example:

```
<audio controls>
<source src="music.mp3" type="audio/mpeg">
<source src="music.ogg" type="audio/ogg">
Your browser does not support the audio element.
</audio>
```

In this example, the <audio> tag includes the controls attribute, which adds controls for playing, pausing, and adjusting the volume of the audio file. The <source> tag is used to specify different audio file formats (mp3 and ogg) for compatibility with different web browsers. The text "Your browser does not support the audio element." will be displayed if the browser does not support the audio tag.You would replace "music.mp3" and "music.ogg" with the actual file names and file paths of your audio file(s).

(b) <VIDEO> Tag:

The HTML <video> tag is used to embed video content into a web page. Here's an example:

```
<video controls width="640" height="360">
<source src="video.mp4" type="video/mp4">
<source src="video.webm" type="video/webm">
Your browser does not support the video tag.
</video>
```

In this example, the <video> tag includes the controls attribute, which adds controls for playing, pausing, and adjusting the volume of the video file. The width and height attributes are used to set the dimensions of the video player. The <source> tag is used to specify different video file formats (mp4 and webm) for compatibility with different web browsers. The text "Your browser does not support the video tag." will be displayed if the browser does not support the video tag.

You would replace "video.mp4" and "video.webm" with the actual file names and file paths of your video file(s).

12.5.3 The <svg> Tag

The <svg> tag is an HTML element used for embedding scalable vector graphics in a web page. SVG is a language for describing two-dimensional graphics in XML format. It allows you to create drawings, diagrams, and other types of visualizations that can be scaled up or down without losing quality.

Here's an example of how to use the <svg> tag:

```
<!DOCTYPE html>
<html>
<head>
<title>SVG Example</title>
</head>
<body>
```

```
<h1>SVG Example</h1>
<svg width=”400" height=”200">
<rect x=”50" y=”50" width=”300" height=”100" fill=”blue” />
<circle cx=”200" cy=”100" r=”50" fill=”yellow” />
<text x=”200" y=”125" font-size=”30" fill=”red”>SVG</text>
</svg>
</body>
</html>
```

In this example, we have used the <svg> tag to create a rectangular shape with a blue fill, a circle with a yellow fill, and text with a red fill. The width and height attributes define the size of the SVG canvas. The x and y attributes determine the position of the rectangle, and the cx and cy attributes determine the center of the circle. The r attribute specifies the radius of the circle. The text element is used to add text to the SVG, and the font-size attribute sets the size of the text.



12.6 LET US SUM UP

- HTML forms are used to allow users to input data and interact with web pages.
- Various form elements, such as text fields, checkboxes, radio buttons, dropdown menus, and buttons, can be used to create different types of form controls.
- HTML forms use various attributes to define the behavior of the form, such as the method attribute to define the HTTP method for submitting the form.
- The action attribute is used to specify the URL where the form data is sent, and the enctype attribute is used to specify the encoding type for the form data.
- When a user submits a form, the data entered by the user is sent to the server for processing.

- HTML forms can be styled using CSS to enhance their appearance and make them more user-friendly.
- JavaScript can be used to add functionality to forms, such as form validation, autocomplete, and dynamic form elements that change based on user input.
- The <label> tag in HTML is used to associate a label with an input element.
- The <input> tag is used to create various types of form elements such as text fields, radio buttons, checkboxes, buttons, and more.
- The <textarea> tag is used to create a multi-line text input field in HTML.
- The <canvas> tag allows for the creation of dynamic and interactive graphics and animations within a web page.
- The <audio> and <video> tags are used to embed audio and video content into a web page.
- The <svg> tag is an HTML element used for embedding scalable vector graphics in a web page.



12.7 FURTHER READING

1) https://www.w3schools.com/html/default.asp
2) "HTML5 and CSS3: Develop with Tomorrow's Standards Today" by Brian P. Hogan
3) "HTML & CSS: The Complete Reference" by Thomas Powell and Chuck White



12.8 ANSWERS TO CHECK YOUR PROGRESS

Ans.to.Q.No.1: HTML forms allow users to input data and interact with web pages, enabling web developers to create dynamic web applications that can collect user input and provide a customized user experience based on that input.

Ans.to.Q.No.2: The <label> tag in HTML is used to associate a label with an input element to provide a description or a caption for an input field so that users can easily understand the purpose of the input field.

Ans.to.Q.No.3: The <input type="submit”> tag in HTML is used to create a submit button that the user can click to submit the form.

Ans.to.Q.No.4: The <textarea> tag in HTML is used to create a multi-line text input field in forms where users are expected to enter more than one line of text, such as a comment or a message.



12.9 MODEL QUESTIONS

Q.1: What is the purpose of HTML forms in web development?

Q.2: What are some common form elements used in HTML?

Q.3: How do you create a label in HTML for an input field?

Q.4: What attributes are used to define the behavior of an HTML form?

Q.5: How do you create a text input field in HTML?

Q.6: How do you create a checkbox in HTML?

Q.7: How do you create a group of radio buttons in HTML?

Q.8: How do you create a submit button in HTML?

Q.9: What is the purpose of the <textarea> tag in HTML forms?

Q.10: What is the purpose of the <audio> tag in HTML, and how can it be used to embed audio content on a web page?

Q.11: How does the <video> tag work in HTML, and what attributes can be used to customize the video player?

Q.12: What is SVG, and how can the <svg> tag be used to embed scalable vector graphics on a web page?

Q.13: Can you explain the syntax of the <svg> tag, and provide an example of how it can be used to create visualizations with different shapes and colors?

****

UNIT 13 : INTRODUCTION TO XML

UNIT STRUCTURE

13.1 Learning Objectives

13.2 Introduction

13.3 Overview of XML

13.4 XML Basics

13.5 Introduction to DTD

- 13.5.1 Internal DTD
- 13.5.2 External DTD

13.6 Let Us Sum Up

13.7 Further Reading

13.8 Answers to Check Your Progress

13.9 Model Questions

13.1 LEARNING OBJECTIVES

After going through this unit, learners will be able to

- define what XML is and explain its purpose
- compare and contrast HTML and XML and determine when to use each one
- explain the syntax rules of XML, including elements, attributes, and values
- understand the concept of XML namespaces and how they are used in XML documents
- define what DTD is and explain its purpose in XML
- understand the syntax rules of DTD and how it is used to validate XML documents
- create simple XML documents and XML documents with DTDs.

13.2 INTRODUCTION

XML (eXtensible Markup Language) is a versatile markup language used to store and transport data. It is designed to be self-descriptive, making it an

ideal choice for exchanging data between different systems and platforms. In this unit, we will provide an overview of XML, compare it with HTML, explore its syntax rules, and learn about its namespaces. We will also introduce DTD (Document Type Definition), which is used to validate XML documents. By the end of this unit, learners will gain a thorough understanding of XML, how it is used, and how it can be optimized for performance and efficiency.

13.3 OVERVIEW OF XML

XML stands for "eXtensible Markup Language." It is a markup language used for encoding documents in a format that can be read by both humans and machines. XML was designed to be both human-readable and machine-readable, making it a versatile and widely used language for data interchange and storage.

XML documents are made up of elements, which consist of a start tag, content, and an end tag. Elements can also have attributes, which provide additional information about the element. XML documents can also include comments and processing instructions.

XML is often used in web development, database management, and data exchange between different systems. It is a flexible and extensible language, which means that it can be customized to suit specific needs by creating new tags and attributes.

Purpose of Using XML: The primary purpose of XML is to provide a standard format for exchanging data between different systems, platforms, and programming languages. XML is a flexible and extensible markup language that can be used to create custom tags and attributes to represent any kind of data, making it a popular choice for data exchange and storage. Some of the key purposes of XML include:

(a) Data exchange: XML is commonly used for exchanging data between different systems, such as web services, APIs, and databases. By using a standardized format for the data, systems can communicate and exchange information seamlessly.

(b) Data storage: XML can be used to store data in a structured format that is easy to parse and manipulate. This makes it a popular choice for storing configuration files, metadata, and other types of structured data.

(c) Web development: XML is often used in web development to represent data in a standardized format that can be easily understood by both humans and machines. For example, XML is used in RSS feeds, which allow websites to syndicate content and share it with other websites and services.

(d) Document markup: XML can be used to markup documents in a structured way, making it easier to search, index, and process the content. This is particularly useful for large documents such as technical specifications, legal documents, and scientific papers.

Similarities of HTML and XML

Here are some similarities and differences between HTML and XML:

- Both HTML and XML use tags to define elements and attributes to describe them.
- Both HTML and XML allow for nesting of elements, allowing complex data structures to be represented.
- Both HTML and XML can be used to create structured documents that can be read by machines as well as humans.
- Both HTML and XML are text-based and can be viewed and edited in any text editor.

Differences of HTML and XML

- HTML is designed for displaying data on the web, while XML is designed for carrying data. HTML is mainly used for creating web pages and web applications, while XML is mainly used for exchanging data between different systems.
- HTML has a predefined set of tags that are used to define elements such as headings, paragraphs, lists, and links, whereas XML does not have predefined tags, and users can define their own tags and attributes.

- HTML is less strict than XML in terms of syntax rules. HTML allows for unclosed tags and non-strict syntax, whereas XML requires all elements to be properly nested and closed.
- HTML is typically used to display visual content, while XML is often used for data storage, data exchange, and configuration files.
- HTML can be parsed by a web browser, while XML requires a separate program or parser to interpret the data.

In summary, while both HTML and XML use tags and attributes to define elements, they have different purposes and syntax rules. HTML is mainly used for creating web pages and displaying content, while XML is used for data storage and exchange between different systems.

13.4 XML BASICS

XML (Extensible Markup Language) has a set of syntax rules that must be followed in order to create valid XML documents. Here are some of the key syntax rules of XML:

1. XML documents must have a root element that contains all other elements.
2. All XML elements must have a closing tag. Empty elements can be written as self-closing tags, for example:
3. XML element names must follow certain naming rules:
   - Names must start with a letter or underscore.
   - Names cannot start with the letters “xml” (or any variation of capitalization).
   - Names can contain letters, digits, hyphens, underscores, and periods.
4. Attribute values must be enclosed in quotes (either single or double quotes).
5. XML documents must be well-formed, which means they must follow certain structural rules, such as:
   - All elements must be properly nested within each other (no overlapping elements).

- All elements must have an opening and closing tag (except for empty elements).
- There can only be one root element in a document.

6. XML documents can also be valid, which means they follow the rules of a particular XML schema or Document Type Definition (DTD).
7. XML is case-sensitive, so element and attribute names must be spelled exactly as they are defined.

Following these syntax rules is important for ensuring that XML documents can be parsed correctly and used by other programs and systems. If a document does not follow these rules, it may be considered invalid and may not be usable by other programs or systems.

A Sample XML Document:

In XML, elements, attributes, and values are used to define and describe data. Here is an example of how these components are used in an XML document:

```
<?xml version=”1.0" encoding=”UTF-8"?>
<Assam>
<District name=”Dibrugarh”>
<Population>1,327,748</Population>
<Area unit=”km²”>3,381</Area>
</District>
<District name=”Guwahati”>
<Population>2,930,000</Population>
<Area unit=”km²”>328</Area>
</District>
<District name=”Jorhat”>
<Population>1,091,295</Population>
<Area unit=”km²”>2,851</Area>
</District>
</Assam>
```

In this example, the XML document represents the state of Assam and contains information about its districts. Here are some of the key elements, attributes, and values used in the document:

Elements:

- The root element is <Assam>, which contains all the other elements.
- Each district is represented by an <District> element, with the name of the district as an attribute.
- Within each <District> element, there are two child elements: <Population> and <Area>.

Attributes:

- The name attribute is used to specify the name of each district.
- The unit attribute is used to specify the unit of measurement for the area of each district.

Values:

- The <Population> element contains the population of each district.
- The <Area> element contains the area of each district. Together, elements, attributes, and values are used to create structured and meaningful data in XML documents. This example shows how they can be used to describe books in a bookstore.

This example shows how elements, attributes, and values can be used in XML to represent structured data. By following the rules of XML syntax, this data can be easily parsed and used by other programs and systems.

Using Namespace

XML namespaces are used to avoid naming conflicts between elements and attributes in an XML document.

Example XML document with namespaces:

```
<?xml version=”1.0" encoding=”UTF-8"?>
<assam xmlns:g=”http://www.geo.org/”>
<g:District name=”Dibrugarh”>
<g:Population>1,327,748</g:Population>
<g:Area unit=”km²”>3,381</g:Area>
</g:District>
```

```
<g:District name=”Guwahati”>
<g:Population>2,930,000</g:Population>
<g:Area unit=”km²”>328</g:Area>
</g:District>
<g:District name=”Jorhat”>
<g:Population>1,091,295</g:Population>
<g:Area unit=”km²”>2,851</g:Area>
</g:District>
</assam>
```

In this example, the XML document represents the state of Assam and contains information about its districts. Here’s how namespaces are used in this document:

(a) Namespace declaration:

The xmlns:g="http://www.geo.org/” declaration at the start of the document defines a new namespace called “g”.

The namespace is associated with the URI “http://www.geo.org/”.

(b) Namespace prefix:

The prefix “g:” is used before the element and attribute names that belong to the “http://www.geo.org/” namespace.

(c) Elements and attributes:

The <g:District> element and its child elements <g:Population> and <g:Area> belong to the “http://www.geo.org/” namespace.

The name attribute belongs to the “http://www.geo.org/” namespace.

By using namespaces, this XML document can contain elements and attributes with the same name as elements and attributes in other XML documents without causing naming conflicts. This helps to ensure that the data is well-structured and easy to parse by other programs and systems.



CHECK YOUR PROGRESS

1. State whether the following statements are true or false:

(a) XML stands for "Extensible Markup Language."

(b) XML documents can only be read by machines, not humans.

(c) HTML and XML are used for the same purpose.

(d) All XML elements must have a closing tag.

(e) XML syntax rules do not need to be followed in order to create valid XML documents.

13.5 INTRODUCTION TO DTD

DTD (Document Type Definition) is a schema language used in XML to define the structure and constraints of an XML document. DTD can be used to define the elements, attributes, and content of an XML document.

Example DTD for an XML document representing the state of Assam:

```
<!DOCTYPE assam [
<!ELEMENT assam (District*)>
<!ELEMENT District (Population, Area)>
<!ATTLIST District name CDATA #REQUIRED>
<!ELEMENT Population (#PCDATA)>
<!ELEMENT Area (#PCDATA)>
<!ATTLIST Area unit CDATA #IMPLIED>
]>
```

In this example, the DTD defines the structure and constraints of an XML document representing the state of Assam. Here's how the DTD works:

1. The <!DOCTYPE> declaration specifies the root element of the XML document as assam.
2. The <!ELEMENT> declarations define the structure of the XML document.
   - The assam element can contain any number of District elements.
   - Each District element must contain exactly one Population element and one Area element.

- The Population and Area elements can contain character data (i.e., text).

3. The <!ATTLIST> declarations define the attributes of the XML document.
   - The name attribute is required and has a data type of CDATA (character data).
   - The unit attribute is optional and has a data type of CDATA.

By defining the structure and constraints of an XML document using DTD, you can ensure that the document conforms to a specific format and contains only valid data. This can help to improve the quality and consistency of your data, and make it easier to parse and process by other programs and systems.

In XML, there are two types of DTDs (Document Type Definitions): internal DTD and external DTD. Let us discuss the use of both of them with examples.

13.5.1 Internal DTD

An internal DTD is defined within the XML document itself using a <!DOCTYPE> declaration. The DTD is enclosed in square brackets [ ] and appears before the root element of the document. Internal DTDs are useful for small XML documents or for documents that are not reused frequently.

Example of an Internal DTD:

```
<?xml version=”1.0"?>
<!DOCTYPE assam [
<!ELEMENT assam (District*)>
<!ELEMENT District (Population, Area)>
<!ATTLIST District name CDATA #REQUIRED>
<!ELEMENT Population (#PCDATA)>
<!ELEMENT Area (#PCDATA)>
<!ATTLIST Area unit CDATA #IMPLIED>
]>
<assam>
<District name=”Kamrup”>
```

```
<Population>1517543</Population>
<Area unit=”square km”>4343</Area>
</District>
<District name=”Tinsukia”>
<Population>1323509</Population>
<Area unit=”square km”>3790</Area>
</District>
<District name=”Dibrugarh”>
<Population>1530817</Population>
<Area unit=”square km”>3381</Area>
</District>
<District name=”Nagaon”>
<Population>2823685</Population>
<Area unit=”square km”>3951</Area>
</District>
</assam>
```

In this example, the internal DTD is defined within the <!DOCTYPE> declaration at the beginning of the XML document. The DTD specifies that the assam element can contain any number of District elements, each of which must contain exactly one Population element and one Area element. The District element has a required name attribute, while the Area element has an optional unit attribute.

The XML document itself contains four District elements, each with a different name, population, and area. By adhering to the structure and constraints defined in the DTD, this XML document is considered valid and can be parsed and processed by other programs and systems that understand the DTD.

13.5.2 External DTD

Here’s an example of the same XML document with an external DTD:

- *XML Document (assam.xml):*

```
<?xml version=”1.0"?>
<!DOCTYPE assam SYSTEM “assam.dtd”>
<assam>
<District name=”Kamrup”>
<Population>1517543</Population>
<Area unit=”square km”>4343</Area>
</District>
<District name=”Tinsukia”>
<Population>1323509</Population>
<Area unit=”square km”>3790</Area>
</District>
<District name=”Dibrugarh”>
<Population>1530817</Population>
<Area unit=”square km”>3381</Area>
</District>
<District name=”Nagaon”>
<Population>2823685</Population>
<Area unit=”square km”>3951</Area>
</District>
</assam>
```

- *External DTD (assam.dtd):*

```
<!ELEMENT assam (District*)>
<!ELEMENT District (Population, Area)>
<!ATTLIST District name CDATA #REQUIRED>
<!ELEMENT Population (#PCDATA)>
<!ELEMENT Area (#PCDATA)>
<!ATTLIST Area unit CDATA #IMPLIED>
```

In this example, the XML document refers to an external DTD file called assam.dtd using a <!DOCTYPE> declaration at the beginning of the document. The SYSTEM keyword is used to specify the location of the DTD file.

The external DTD file contains the same DTD definitions as the internal DTD example. However, in the external DTD, the <!DOCTYPE> declaration and the DTD definitions themselves are separated into two files. This can be useful when multiple XML documents need to adhere to the same DTD structure and constraints.

When processing the XML document, a parser will read the <!DOCTYPE> declaration and retrieve the DTD definitions from the external DTD file. The parser will then use these definitions to validate the XML document and ensure that it adheres to the specified structure and constraints.

Internal Vs External DTD

Whether to use an internal or external DTD depends on the specific requirements of your project and how you plan to manage your XML documents and their associated DTDs.

Internal DTDs are defined within the XML document itself, which means that the entire document is self-contained and can be easily distributed as a single file. This can be convenient for smaller projects or when you don't need to share the DTD with other documents or systems.

External DTDs, on the other hand, are defined in a separate file that is referenced by the XML document using a <!DOCTYPE> declaration. This allows the same DTD to be used by multiple XML documents, which can be helpful for larger projects or when you want to standardize the structure and constraints across multiple documents.

Here are some reasons why you might choose one approach over the other:

Use an internal DTD if:

- You only have a single XML document to validate
- You want to keep the document self-contained and don't want to rely on an external file

- You don't need to share the DTD with other documents or systems
- You want to keep the DTD definition close to the document content for easy reference

Use an external DTD if:

- You have multiple XML documents that need to adhere to the same DTD structure and constraints
- You want to standardize the DTD definition across multiple documents
- You want to simplify the maintenance of the DTD by keeping it in a separate file
- You want to reuse the same DTD across different projects or systems

Ultimately, the choice between an internal or external DTD comes down to your specific project requirements and preferences. Both approaches have their advantages and disadvantages, and the best choice depends on your situation.



CHECK YOUR PROGRESS

2. State whether the following statements are true or false:

(a) XML stands for "Extensible Markup Language."

(b) XML documents can only be read by machines, not humans.

(c) HTML and XML are used for the same purpose.

(d) All XML elements must have a closing tag.

(e) XML syntax rules do not need to be followed in order to create valid XML documents.



13.6 LET US SUM UP

- XML stands for eXtensible Markup Language and is used for encoding documents in a format that can be read by both humans and machines.

- XML documents are made up of elements that consist of a start tag, content, and an end tag. Elements can also have attributes that provide additional information about the element.
- XML is primarily used for exchanging data between different systems, data storage, web development, and document markup.
- HTML and XML both use tags and attributes to define elements and allow for nesting of elements, but they have different purposes and syntax rules.
- Key syntax rules for creating valid XML documents include having a root element, properly nested elements, and enclosed attribute values.
- XML documents can be well-formed and valid, following specific XML schemas or Document Type Definitions (DTD).
- XML is case-sensitive, meaning element and attribute names must be spelled exactly as defined.
- Following syntax rules is important for ensuring that XML documents can be parsed correctly and used by other programs and systems.
- DTD (Document Type Definition) is a schema language used in XML to define the structure and constraints of an XML document.
- DTD can be used to define the elements, attributes, and content of an XML document. By defining the structure and constraints of an XML document using DTD, you can ensure that the document conforms to a specific format and contains only valid data.
- There are two types of DTDs in XML: internal DTD and external DTD. Internal DTD is defined within the XML document itself, while external DTD is stored in a separate file and referenced by the XML document.
- DTD can help to improve the quality and consistency of your data, and make it easier to parse and process by other programs and systems.



13.7 FURTHER READING

1) https://www.w3schools.com/xml/



13.8 ANSWERS TO CHECK YOUR PROGRESS

Ans.to.Q.No.1: (a)True (b)False (c)False (d)True (e)False

Ans.to.Q.No.2: (a) Document Type Definition

Ans.to.Q.No.2: (b) <!DOCTYPE>

Ans.to.Q.No.2: (c) structure

Ans.to.Q.No.2: (d) name

Ans.to.Q.No.2: (e) DTD file (or .dtd file)



13.9 MODEL QUESTIONS

Q.1: What is XML, and what is its primary purpose?

Q.2: How are XML documents structured, and what components do they consist of?

Q.3: What are some common uses of XML in web development and data exchange?

Q.4: What are the similarities and differences between HTML and XML?

Q.5: What are some syntax rules that must be followed when creating valid XML documents?

Q.6: How is data defined and described in an XML document, and what are some examples of elements and attributes?

Q.7: Why is it important to follow syntax rules when creating XML documents, and what happens if a document is not well-formed or valid?

Q.8: What is DTD and how is it used in XML?

Q.9: What is the purpose of defining the structure and constraints of an XML document using DTD?

Q.10: What are the two types of DTDs in XML and how are they different from each other?

Q.11: What is an internal DTD and when is it typically used in XML?

Q.12: What is an external DTD and what are the advantages of using it in XML?

****

UNIT 14 : INTRODUCTION TO CSS

UNIT STRUCTURE

14.1 Learning Objectives

14.2 Introduction

14.3 Overview of CSS

14.4 Types of CSS

- 14.4.1 Inline Style Sheet
- 14.4.2 Embedded Style Sheet
- 14.4.3 External Style Sheet
- 14.4.4 Imported Style Sheet

14.5 Selectors

- 14.5.1 Element Selector
- 14.5.2 Class Selector
- 14.5.3 ID Selector
- 14.5.4 Attribute selector
- 14.4.5 Pseudo-class Selector

14.6 Let Us Sum Up

14.7 Further Reading

14.8 Answers to Check Your Progress

14.9 Model Questions

14.1 LEARNING OBJECTIVES

After going through this unit, learners will be able to

- understand the purpose and role of CSS in web development.
- identify the different types of CSS and their respective usage, including inline style, embedded style, external style sheets, and imported style sheets.
- analyze the various CSS selectors and how they are used to target specific HTML elements.
- evaluate the different CSS properties and their impact on web design, including background, text, and border properties.

- apply background properties in CSS to create visually appealing backgrounds.
- utilize text properties in CSS to customize the appearance of text on a web page.
- implement border properties in CSS to add borders around HTML elements and create separation between different sections of a web page.

14.2 INTRODUCTION

Cascading Style Sheets (CSS) is a fundamental component of web development that allows web designers to control the layout, formatting, and presentation of HTML documents. In this unit, we will explore the different types of CSS, including inline style, embedded style, external style sheets, and imported style sheets. We will also cover CSS selectors, which are used to target specific HTML elements for styling, as well as CSS properties, including background, text, and border properties, which allow designers to create visually appealing and user-friendly web pages. By the end of this unit, you will have a solid understanding of CSS and its applications in web development, as well as the skills necessary to implement various CSS techniques in your own projects.

14.3 OVERVIEW OF CSS

CSS stands for Cascading Style Sheets, which is a language used for describing the presentation of web pages, including layout, colors, fonts, and other visual elements. CSS allows designers to separate the presentation of a web page from its content, which provides several advantages such as making it easier to maintain the layout and visual style of a website across multiple pages, and simplifying the creation of responsive web designs that can adapt to different devices and screen sizes.

CSS works by selecting HTML elements in a web page using selectors and then applying styles to them using properties and values. For example, a CSS selector might target all the paragraph elements in a web page,

and the associated properties might include font-size, color, and line-height, while the values might specify a particular size, color, or other formatting option.

There are many advantages of using CSS for web design, including:

- Separation of presentation and content: CSS allows designers to separate the visual presentation of a web page from its underlying HTML structure. This makes it easier to maintain the layout and visual style of a website across multiple pages, as changes to the CSS file can be made without affecting the HTML code.
- Consistent design: CSS allows designers to create a consistent look and feel across multiple pages on a website. By defining styles in a single CSS file, designers can ensure that all pages on a website have a consistent layout, typography, and color scheme.
- Faster page loading times: By separating the presentation of a web page from its content, CSS can help to reduce the size of the HTML files, making web pages load faster. This is because CSS files can be cached by the browser, allowing them to be loaded more quickly on subsequent visits to a website.
- Flexibility and responsiveness: CSS provides a flexible and responsive design approach that allows web pages to adapt to different devices and screen sizes. By using CSS media queries, designers can create layouts that adjust automatically to the size of the screen, making web pages more user-friendly and accessible on a range of devices.
- Easy to learn and use: CSS is relatively easy to learn and use, making it accessible to both beginners and experienced web designers. The language has a simple syntax and a large community of developers who share knowledge and provide support.

14.4 TYPES OF CSS

There are several types of CSS that can be used in web design, includingInline Styles, Embedded Styles, External Style Sheets, and

Imported Style Sheets. In the following sections, we will discuss how these CSS styles works.

14.4.1 Inline Style Sheet

Inline CSS refers to CSS styles that are applied directly to individual HTML elements using the “style” attribute. This method of applying styles can be useful for making small style changes to specific elements on a web page, but is generally not recommended for larger projects, as it can make the HTML code difficult to read and maintain.

To apply inline CSS, simply add the “style” attribute to an HTML element and define the styles within the attribute using CSS syntax. For example, to change the color of a heading to red, you could use the following code:

```
<h1 style=”color: red;”>Welcome to KKHSOU</h1>
```

In this example, the “style” attribute is used to apply the color red to the heading element.

While inline CSS can be useful for making quick and simple style changes, it can be difficult to maintain and update, especially on larger projects. For this reason, most web designers prefer to use external or embedded stylesheets to define styles for their web pages.

14.4.2 Embedded Style Sheet

Embedded CSS is a method of adding CSS styling directly within the HTML document, using the <style> tag. This is useful for small-scale projects, or for styles that only apply to a specific page, rather than globally to an entire website.

```
<!DOCTYPE html>
<html>
<head>
        <title>Embedded CSS Example</title>
        <style>
```

```
            body {
                background-color: #f2f2f2;
                font-family: Arial, sans-serif;
                font-size: 16px;
                color: #333333;
            }
            h1 {
                font-size: 24px;
                font-weight: bold;
                color: #008000;
                margin-bottom: 20px;
            }
            p {
                line-height: 1.5;
                margin-bottom: 10px;
            }
        </style>
</head>
<body>
        <h1>Welcome to my website!</h1>
        <p>This is an example of embedded CSS. It allows you to add CSS styles directly to your HTML document.</p>
        <p>You can use the <code>&lt;style&gt;</code> tag to define styles for different HTML elements, such as <code>&lt;p&gt;</code>, <code>&lt;h1&gt;</code>, and <code>&lt;body&gt;</code>.</p>
</body>
</html>
```

Explanation:

The above code is an example of HTML code that includes embedded CSS styles. It begins by declaring the HTML document type as HTML5, followed by opening the HTML document and starting the head section.

Within the head section, a title is set for the document using the <title> tag. The document's title is set to "Embedded CSS Example". The CSS styles are defined within the <style> tags. The styles defined in this example apply to the body, h1, and p HTML elements. For the body element, the background color is set to #f2f2f2, the font family is set to Arial or any sans-serif font, the font size is set to 16px, and the text color is set to #333333.

For the h1 element, the font size is set to 24px, the font weight is set to bold, the color is set to #008000, and the margin bottom is set to 20px.

For the p element, the line height is set to 1.5 and the margin bottom is set to 10px.

Finally, the HTML document is closed with the </html> tag.

14.4.3 External Style Sheet

External CSS is a way of separating the presentation of an HTML document from its content by using a separate CSS file. With external CSS, the styles that determine the layout, color, font, and other visual aspects of the HTML document are defined in a separate file with a .css file extension, rather than being included within the HTML file itself.

To use external CSS, the HTML document must include a <link> element within the <head> section, which specifies the location of the external CSS file. The <link> element includes a href attribute, which is set to the URL of the CSS file.

Here's an example of how the above code (included in the Embedded Style) could be modified to use an external style sheet:

HTML file (example.html):

```
<!DOCTYPE html>
<html>
<head>
        <title>Embedded CSS Example</title>
        <link rel="stylesheet" href="styles.css">
```

```
</head>
<body>
    <h1>Welcome to my website!</h1>
    <p>This is an example of embedded CSS. It allows you to add CSS styles directly to your HTML document.</p>
    <p>You can use the <code>&lt;style&gt;</code> tag to define styles for different HTML elements, such as <code>&lt;p&gt;</code>, <code>&lt;h1&gt;</code>, and <code>&lt;body&gt;</code>.</p>
</body>
</html>
```

CSS file (styles.css):

```
body {
    background-color: #f2f2f2;
    font-family: Arial, sans-serif;
    font-size: 16px;
    color: #333333;
}
h1 {
    font-size: 24px;
    font-weight: bold;
    color: #008000;
    margin-bottom: 20px;
}
p {
    line-height: 1.5;
    margin-bottom: 10px;
}
```

Explanation:

In this example, we've removed the <style> tags from the HTML file and added a <link> tag to reference the external style sheet. The href attribute of the <link> tag specifies the location of the CSS file.

In this case, the CSS file is named “styles.css” and is located in the same directory as the HTML file.

The CSS file contains the same styles that were previously defined within the <style> tags. By moving these styles to an external file, we can keep our HTML file cleaner and easier to read, and also make it easier to maintain and update the styles across multiple pages.

14.4.4 Embedded Style Sheet

An embedded style sheet is a type of CSS (Cascading Style Sheets) that is placed within the HTML document itself, instead of being stored in a separate CSS file. This means that all the styling rules are defined within the HTML file itself, rather than in a separate file, making it easier to manage and modify the styles for a single page.

To use an embedded style sheet, you would typically include a <style> tag in the <head> section of your HTML document. Within this tag, you can write CSS rules just like you would in an external CSS file. For example:

```
<head>
<title>My Website</title>
<style>
   body {
    background-color: #f1f1f1;
    font-family: Arial, sans-serif;
   }
   h1 {
    color: #333;
    font-size: 32px;
    font-weight: bold;
   }
</style>
</head>
```

In the above example, the <style> tag contains two CSS rules that define the background color and font family for the entire page, as well as the color, font size, and font weight for any <h1> elements within the page.

Using an embedded style sheet can be a convenient way to apply styles to a single page, without having to create a separate CSS file. However, it may become unwieldy for larger websites with many pages, as each page would need to have its own embedded style sheet. In such cases, an external CSS file would be a better option.



CHECK YOUR PROGRESS

1. State whether the following statements are true or false:

(a) Inline CSS is recommended for larger web design projects.

(b) External Style Sheets are defined within the <style> tags.

(c) CSS can help to reduce the size of HTML files, which can result in faster page loading times.

(d) CSS is a complex programming language that is difficult to learn and use.

2. Answer the following questions:

(a) What is the advantage of using CSS for web design?

(b) What is an example of inline CSS?

(c) What is an external style sheet in CSS?

14.5 SELECTORS

CSS (Cascading Style Sheets) selectors are used to target and apply styles to specific HTML elements on a web page. Selectors can be based on the element type, class name, ID, attribute, and even the relationship between elements.

There are several types of selectors in CSS, which are explained in the following sections.

14.5.1 Element Selector

The element selector in CSS is used to select all elements of a particular type on a web page and apply a style to them. It targets an HTML element using its tag name.

For example, if you want to style all paragraphs on a web page, you can use the element selector like this:

```
p {
  color: red;
  font-size: 16px;
}
```

In this example, the CSS selector p selects all <p> elements on the page and applies the specified styles. The color property sets the color of the text to red, and the font-size property sets the size of the font to 16 pixels.

Here is an example of using the element selector in an external CSS file:

index.html:

```
<!DOCTYPE html>
<html>
<head>
<link rel=”stylesheet” type=”text/css” href=”style.css”>
</head>
<body>
<p>Hello, World!</p>
<p>This is a paragraph.</p>
</body>
</html>
```

style.css:

```
p {
  color: red;
  font-size: 16px;
}
```

In this example, the CSS file is linked to the HTML file using the link tag in the head section. The p selector in the CSS file selects all <p> elements in the HTML file and applies the specified styles to them. When you open the HTML file in a web browser, you will see that both paragraphs are styled with red text and a font size of 16 pixels.

14.5.2 Class Selector

Class selectors in CSS are used to select elements with a specific class attribute. They are denoted by a period (.) followed by the class name.

Here's an example of how to use class selectors in external CSS:

HTML file (index.html):

```
<!DOCTYPE html>
<html>
<head>
        <title>Class Selector Example</title>
        <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
        <h1 class="heading">Hello World!</h1>
        <p class="paragraph">This is an example of a class selector.</p>
        <p>This is a paragraph without any class.</p>
</body>
</html>
```

CSS code (style.css):

```
.heading {
        color: blue;
        font-size: 32px;
}
.paragraph {
        color: red;
        font-size: 18px;
}
```

In this example, the class attribute is used to give specific elements a class name. The h1 element with the text “Hello World!” has the class name heading, while the p element with the text “This is an example of a class selector.” has the class name paragraph.

The CSS code then selects the elements with the heading and paragraph class names and applies styles to them. In this case, the heading class sets the color to blue and font size to 32 pixels, while the paragraph class sets the color to red and font size to 18 pixels.

Note that you can apply the same class name to multiple elements in your HTML code, and all of those elements will receive the same styles.

Class selectors are useful when you want to apply the same styles to multiple elements that don’t necessarily belong to the same tag.

14.5.3 ID Selector

The ID selector in CSS is used to select an element based on its unique ID attribute value. ID selectors are denoted by the hash symbol (#) followed by the ID value.

Here is an example of how to use ID selectors in an external CSS file:

HTML file (index.html):

```
<!DOCTYPE html>
<html>
<head>
        <title>ID Selector Example</title>
        <link rel=”stylesheet” type=”text/css” href=”style.css”>
</head>
<body>
        <h1 id=”header”>Welcome to my Website</h1>
        <p>This is an example of using ID selector in CSS.</p>
</body>
</html>
```

CSS file (style.css):

```
#header {
   color: red;
   font-size: 24px;
}
```

In the above code, the element <h1> is assigned an ID value of “header”. In the CSS file, the ID selector #header is used to apply the styles color: red; and font-size: 24px; to the element with ID “header”.

When the HTML file is opened in a web browser, the element <h1> will be displayed with red font color and font size of 24 pixels, as specified in the CSS file.

14.5.4 Attribute selector

Attribute selectors in CSS allow you to select elements based on the presence or value of an attribute. There are several types of attribute selectors in CSS, including [attribute] to select elements with a specific attribute, [attribute=value] to select elements with a specific attribute value, and [attribute^=value] to select elements with an attribute that starts with a certain value.

Here is an example of how to use attribute selector in an HTML file and a CSS file:

HTML file (example.html):

```
<!DOCTYPE html>
<html>
<head>
        <title>Attribute Selector Example</title>
        <link rel=”stylesheet” type=”text/css” href=”example.css”>
</head>
<body>
        <h1 data-color=”red”>Red Heading</h1>
        <p data-color=”green”>Green Paragraph</p>
        <h2>Blue Heading</h2>
```

```
        <p>Regular Paragraph</p>
</body>
</html>
```

In this HTML file, we have four elements: two headings with custom data attributes (data-color), and two regular headings and paragraphs.

CSS file (example.css):

```
h1[data-color=”red”] {
        color: red;
}
p[data-color=”green”] {
        color: green;
}
h2[class^=”blue”] {
        color: blue;
}
```

In this CSS file, we have three attribute selectors. The first selector targets the h1 element with data-color=”red” and sets the color to red. The second selector targets the p element with data-color=”green” and sets the color to green. The third selector targets the h2 element with a class that starts with “blue” and sets the color to blue.

When you open the HTML file in a web browser, you should see that the “Red Heading” is displayed in red, the “Green Paragraph” is displayed in green, and the “Blue Heading” is displayed in blue.

Note that attribute selectors can be very powerful, but they should be used judiciously to avoid making your CSS code overly complex.

14.4.5 Pseudo-class Selector

Pseudo-class selectors are used to select and style elements based on their state or position within the document. They begin with a colon (:) followed by the name of the pseudo-class.

For example, let's say you have a web page with links, and you want to style the links differently when they are being hovered over by the mouse cursor. You can use the :hover pseudo-class selector for this.

Here is an example HTML code:*(index.html)*

```
<!DOCTYPE html>
<html>
<head>
        <title>Example</title>
        <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
        <a href="#">Link</a>
</body>
</html>
```

Here is the corresponding CSS code:*(style.css)*

```
a:hover {
        color: red;
}
```

In this example, we are selecting all anchor (a) elements that are being hovered over (:hover) and changing the color of their text to red.

The :hover pseudo-class is just one example of many pseudo-classes available in CSS, such as :active, :focus, :first-child, :nth-child(), and many more. These selectors allow you to apply styles to elements based on their position, relationship to other elements, or user interaction with the page.



CHECK YOUR PROGRESS

3. State whether the following statements are true or false:

(a) The element selector targets HTML elements based on their ID.

(b) Class selectors in CSS are denoted by a period (.)

(c) The ID selector in CSS is used to select an element based on its unique class attribute value.

(d) Attribute selectors in CSS allow you to select elements based on the presence or value of an attribute.

(e) Class selectors are useful when you want to apply the same styles to multiple elements that don't necessarily belong to the same tag.



14.6 LET US SUM UP

- CSS is a language used for describing the presentation of web pages, including layout, colors, fonts, and other visual elements.
- CSS separates the presentation of a web page from its content, making it easier to maintain and design responsive web pages.
- CSS works by selecting HTML elements and applying styles to them using properties and values.
- Advantages of using CSS include separation of presentation and content, consistent design, faster page loading times, flexibility and responsiveness, and ease of use.
- Inline CSS applies styles directly to individual HTML elements using the "style" attribute, but is not recommended for larger projects.
- Embedded CSS adds styles directly within the HTML document using the <style> tag and is useful for small-scale projects or styles that apply to a specific page.
- External style sheets are separate files containing CSS styles and can be applied to multiple web pages, allowing for consistent design across a website.
- To use external style sheets, the CSS code is saved in a separate file with a .css extension and linked to the HTML document using the <link> tag.

- CSS selectors are used to target and apply styles to specific HTML elements on a web page.
- Element selectors are used to select all elements of a particular type on a web page and apply a style to them.
- Class selectors are used to select elements with a specific class attribute.
- ID selectors are used to select an element based on its unique ID attribute value.
- Attribute selectors in CSS allow you to select elements based on the presence or value of an attribute.



14.7 FURTHER READING

1) https://www.w3schools.com/css/



14.8 ANSWERS TO CHECK YOUR PROGRESS

Ans.to.Q.No.1: (a) False (b) False (c) True (d) False

Ans.to.Q.No.2(a): CSS allows designers to separate the presentation of a web page from its content, making it easier to maintain the layout and visual style of a website across multiple pages, and simplifying the creation of responsive web designs that can adapt to different devices and screen sizes.

Ans.to.Q.No.2(b): An example of inline CSS is: <h1 style=”color: red;”>Welcome to KKHSOU</h1>. This code changes the color of the heading to red using the “style” attribute.

Ans.to.Q.No.2(c): An external style sheet is a separate CSS file that contains styles for one or more web pages and is linked to the HTML document using the <link> tag.

Ans.to.Q.No.3: (a) False (b) True (c) False (d) True (e) True



14.9 MODEL QUESTIONS

Q.1: What is CSS, and how is it used in web design?

Q.2: What are the advantages of using CSS in web design?

Q.3: What are the different types of CSS that can be used in web design?

Q.4: What is an inline style sheet, and when is it appropriate to use one?

Q.5: What is an embedded style sheet, and how is it different from an external style sheet?

Q.6: How can CSS help to improve page loading times on a website?

Q.7: What is the syntax for defining CSS styles, and how are they applied to HTML elements?

Q.8: How can CSS be used to create a consistent design across multiple pages on a website?

Q.9: What are CSS selectors and how are they used in web development?

Q.10: What are the different types of CSS selectors?

Q.11: What is an element selector in CSS? How is the element selector used in an external CSS file?

Q.12: What is a class selector in CSS?How is the class selector used in an external CSS file?

Q.13: What is an ID selector in CSS?How is the ID selector used in an external CSS file?

Q.14: What is an attribute selector in CSS?What are the different types of attribute selectors in CSS?How are attribute selectors used in an HTML file and CSS file?

****

UNIT 15 : INTRODUCTION TO JAVASCRIPT

UNIT STRUCTURE

15.1 Learning Objectives

15.2 Introduction

15.3 Client Side Programming

- 15.3.1 Introduction to JavaScript
- 15.3.2 Advantages of JavaScript

15.4 Basic Programming Concepts

- 15.4.1 Variables
- 15.4.2 Data-Types
- 15.4.3 Operators

15.5 Control Structures in JavaScript

- 15.5.1 Branching
- 15.5.2 Looping

15.6 Array

- 15.6.1 Dense Array
- 15.6.2 Array Within Array
- 15.6.3 Methods of Array

15.7 Functions

- 15.7.1 User-Defined Function
- 15.7.2 Built-in Function
- 15.7.3 Dialog Box

15.8 Let Us Sum Up

15.9 Further Reading

15.10 Answers to Check Your Progress

15.11 ModelQuestions

15.1 LEARNING OBJECTIVES

After going through this unit, learners will be able to

- describe client side programming
- discuss the basic programming concepts of JavaScript

- identify the different control structures used in JavaScript for branching and looping
- explain the Array
- describe function and its types

15.2 INTRODUCTION

In the previous unit, you have learnt the advanced HTML tags and basic concept of XML. This unit deals with the concept of JavaScript. In this unit you will learn about the client side programming. This unit also describes the basic programmin g idea of JavaScript. In addition, you will come across the different control structures used in JavaScript for branching and looping. This unit seeks to provide you a with the concept of array and functions used in JavaScript and you will also be able to learn about the working with form object. Besides you will be acquianted with the basic idea of built-in Objects in JavaScript. Finally, the unit will discuss the use of user defined objects.

15.3 CLIENT SIDE PROGRAMMING

A HTML document is purely static, i.e., it can only present the documents. Interaction is needed to makes it dynamic. To make it interactive, communication is to take place between the client's browser and the web server, where the actual content is stored. In order to fulfil this requirement client side scripting languages are developed. So a client side scripting language is capable of making interaction with the client; it can capture data inserted by the user , and validate data as per the server specification and dispatch corrected data to the web server. The client side programming languages are JavaScript, vbScript etc.

15.3.1 Introduction to JavaScript

JavaScript is a scripting language of the Web. It is the most popular scripting language on the internet and it works in all major browsers, such as Internet Explorer, Firefox, Chrome, Opera, and Safari. JavaScript is used in millions of Web pages to add functionality to validate forms, detect browsers and much more.

Java and JavaScript are two completely different languages in both concept and design. Java (developed by Sun Microsystems) is a powerful and much more complex programming language - in the same category as C and C++.

A scripting language is a lightweight programming language.

- JavaScript was designed to add interactivity to HTML pages
- JavaScript is usually embedded directly into HTML pages
- JavaScript as an interpreted language means that scripts execute without preliminary compilation
- Everyone can use JavaScript without purchasing a license The HTML <SCRIPT> tag is used to insert a JavaScript into an HTML page. A JavaScript program code is included between the tag pair <SCRIPT> and </SCRIPT>, which is contained either in the header section or in the body section of the HTML page.

The **<SCRIPT> Tag :** It is a container tag and it is used to include JavaScript code into a HTML document.

Syntax :

```
<SCRIPT Language="JavaScript”>

 // JavaScript code

</SCRIPT>
```

15.3.2 Advantages of JavaScript

- JavaScript is an interpreted language, which is interpreted by a JavaScript enabled browser just like a HTML code.
- JavaScript does not require any other special editor for programming. Just simple editor is enough, where any HTML code can be created.
- It has comparatively fewer commands;so the programmer needs to know limited syntax.
- It requires minimum storage spaces which lead to better performance.

- JavaScript can properly execute the event handling like an event driven programming language. Example of events may be: press a button, click mouse button, move mouse over etc.
- Error detection-correction process as well as debugging is easy.



CHECK YOUR PROGRESS

Q.1. Fill in the blanks :

i. A HTML document is a purely _____.

ii. JavaScript is an _____.

iii. _____ Tag is a container tag and it is used to include JavaScript code into a HTML document.

15.4 BASIC JAVASCRIPT CONCEPTS

A few points on JavaScript :

- JavaScript is case sensitive, i.e. it can understand the difference between the capital letter and small letters.
- Every JavaScript statement is ended with a semicolon.
- JavaScript uses a **prompt ()** method to receive data from user in the client browser. Here contents are delimited by using comma.
- JavaScript uses **document.write()** methods to display message or content of variable into client browser. Here also contents are delimited by using comma. Here document is a JavaScript object and write is a method of it.
- In document.write() we can even put any HTML tag, delimited by comma.
- JavaScript uses an **alert()** method, which is used to display message or content of variable. Here contents are delimited by '+' operator.
- JavaScript variables are loosely typed. i.e. type of a variable can be decided by looking at its value. A variable can change its type according to the value stored in it. i.e. the declaration of a variable in JavaScript is not mandatory prior to its use.

15.4.1 Variables

Definition : A variable is a place in memory which can hold a data element. When a variable is defined a memory location is assigned internally to it. Value of a variable can vary during the execution of a program. A variable can be declared by using **var** keyword.

Syntax :

var <variable name>=value;

e.g.

var number=100; var name="Gagan";

Rules for giving variable name :

- A variable may start with a uppercase, lowercase, a underscore character (_) or a $ sign.
- Rest of the characters may be letters, underscore (_), doller ($) or digits (0…9).
- If multiple words are used as a variable name then the first character should be lowercase and first character of next words should be capital. e.g. interest Rate, discountAmount etc.

15.4.2 Data-Types

Data type means types of data that can hold by a variable. In JavaScript there are four different primitive data types such as: **Number, Boolean, String** and **Null.** JavaScript also supports complex types like: **array** and **object**.

Number : It can take integer, floating point number or NaN (stands for Not a Number) value. Integer can be decimal, hexadecimal or octal. But at the time of display others are converted into decimal.

Boolean : It is represented by **true** and **false**. These are converted to 1 and 0 automatically when used in numerical expression.

String : Strings are represented by putting alphabets or digits within single or double quotation mark.

Null : This type contains a single value i.e. null, which means empty or nonexistent reference.

15.4.3 Operators

Arithmetic Operators : The normal arithmetic operators can be used here also. Such as:

Operator	Name	Meaning
+	plus	for addition
–	minus	for subtraction
/	slash	for division
*	asterisk	for multiplication
%	modulus	for getting remainder of two integer number division
++	increment	increment by 1
—	decrement	decrement by 1

Relational Operators : these are used to compare two quantities. They are used to construct a condition. They produce Boolean values.

Operator	Meaning
==	Equal
>	Greater than
<	Less than
>=	Greater than and Equal
<=	Less than and Equal
!=	Not Equal
===	Strictly Equal
!==	Not Strictly Equal

Logical Operator : These are used to combine two or more condition. They produce Boolean values.

Operator	Meaning
&&	AND
‖	OR
!	NOT

String Operator : It is a "+" symbol used to perform concatenation.

Assignment Operator : It is used to assign a value or expression of right side of an = sign to a single variable to the left hand side. It has several variations as shown below:

Operator	Example	Result
=	a=5;	a is 5
+=	b=7;b+=a;	b is 12
-=	c=4; b-=c;	b is 8
*=	b*=c;	b is 32
/=	b/=8;	b is 4
%=	a%=3;	a is 2

Delete operator : It is used to delete property of an object or an array element.

e.g. delete array1[2];

New operator : It is used to create an instance of an object.

e.g. array1= new array1();

Example Program 1.

```
<HTML>
 <HEAD>
 <SCRIPT Language="JavaScript">
 var num=10;
 var name="TARAJAN";
</SCRIPT>
</HEAD>
<BODY >
<SCRIPT Language="JavaScript">
// use of document.write(). document.write("Number is =",num,"<br/>"); document.write("Name is =",name,"<br/>");
// use of prompt().
num=prompt("Enter a distance",num); name=prompt("Enter a Destination ",name); document.write("Destination is = ",name,"<br/>");
// use of alert().
```

alert("Distance is = "+num+" k.m. "); document.write("It is",num, " k.m. apart
");

// **use of conditional expression.** str1=(num>100)?"More than 100":"Below 100"; document.write("Num is ",str1,"<br/>");

</SCRIPT>

</BODY>

</HTML>



CHECK YOUR PROGRESS

Q.2. State True or False :

i. JavaScript is case sensitive, i.e. it can understand the difference between the capital letter and small letters.

ii. In document. write () we can put only limited HTML tag.

iii. Data type means types of data that can be hold by a variable

iv. JavaScript does not support complex types like array and object.

v. '–' symbol is used to perform concatenation.

15.5 CONTROL STRUCTURES IN JAVASCRIPT

Control structures like conditional statements are used in JavaScript to perform different actions based on different conditions.

15.5.1 Branching

If statement : it is similar to C language. Here a condition is tested if it is true then it will execute the true part i.e. statement set 1, if it is false then it will execute the false part i.e. statement set 2,

Syntax :

if (<condition>)

{

Statement set 1;

}

[else

```
{
        Statement set 2;
} ]
```

Switch statement : It is more suitable for the situation where we need to select one option from multiple options.

Syntax :

```
switch (choice)
{
 Case 1 :
        Statement set 1; break;
 Case 2 :
        Statement set 1; break;
 Case 3 :
        Statement set 1; break;
        default :
        Statement set 1;
}
```

Immediate if (conditional expression) : JavaScript has a conditional expression operator ? and : , using which immediate if is constructed as stated below.

Syntax :

condition ? value if true : value if false

e.g. b=100;

str1=a>b ? “More than 100” : “Below 100”;

15.5.2 Looping

Looping means to execute the same set of statement repeatedly for several times (zero or more times).

for loop : It is same as C language. It is more suitable when the number of iteration is known earlier.

Syntax :

for(initialization; termination; incrementation/decrementation)

```
{
  Body;
}
```

e.g.

```
for(i=0;i<100;i++)
{
  document.write(I,"<br/>");
}
```

while loop : It is also same as C language. It is executed until a condition is satisfied. Once the condition becomes false, the loop is terminated.

Syntax :

```
while(condition)
{
  Body;
}
```

e.g.

```
i=0;
while(i<100)
{
  document.write(I,"<br/>"); i++;
}
```

do … while loop : It is also same as while loop. The only difference is that here the condition is tested at the end of the loop, whereas in case of while the condition is tested at the beginning of the loop.

Syntax :

```
do
{
  Body;
}
while(condition);
```

e.g.

```
i=0;
do
  {
   document.write(I,"<br/>”); i++;
  }
while(i<100);
```

break and continue statement : These two statements are used in loops.

- **break** statement is used to come out of a loop canceling the rest of the iterations.
- **continue** statement is used to jump to the end of the loop ignoring the statements from its occurrence up to the end of the loop for that iteration only. The next iterations will be executed.

Example program 2 :

```
<html>
<Head>
<script
 language=”JavaScript”> var
 num1=0,num2=0,i=0;
</script>
</head>
<body >
<script
 language=”JavaScript”> num1=prompt
 (“Enter a small number “,num1);
num2=prompt(“Enter a large number “,num2);
 alert(“yoU have inserted ! “+num1+” and “+num2);
for(i=num1; i<=num2;i++)
 document.write(i,”<br/>”); alert(“Over! “);
document.write(“Over! “);
</script>
</body>
</html>
```

15.6 ARRAY

An **array** is a common name given to set of variables. It has an index number to be written within square brackets and started with the value 0. Elements of an array are not necessarily of same type, they can be of any type. The array must be declared before using it. Array is a JavaScript object. Array has several methods for the manipulation of its elements. Array is declared as follows:

Syntax :

arrayname = new Array(size);

Or arrayname = new Array();

Example of array declaration :

name = new Array(12);

Use of array :

name[0]=”Kamal”;

name[1]=”Amal”;

… name[11]=”Ratul”;

15.6.1 Dense Array

It is a JavaScript array where initialization is done at the time of declaration. i.e. Both declaration and initialization are taking place at the same place.

Syntax :

arrayname = new Array(value0, value1, value2, … valuen-1); e.g.

address= new Array(“Sankar”,"Ulubari”,"Guwahati”, 781007);

15.6.2 Array Within Array

An array can be an element of another array, thereby forming array within array. It can also be considered as multidimensional array. The inner array can be accessed by using double dimension.

e.g.

address= new Array(“Sankar”,”Guwahati”, new array(9861288771,9435165455),”Mridul”,”Jorhat”, new array(9976512123, 9707013422,9854329975));

i.e. the array elements for the above array are as follows:

address[0]=”Sankar”

address[1]=”Guwahati” address[2][0]=”9861288771"

address[2][1]=” 9435165455"

address[3]=”Mridul” address[4]=” Jorhat”

address[5][0]=” 9976512123"

address[5][1]=” 9707013422"

address[5][2]=” 9854329975"

15.6.3 Methods of Array

There are a few methods which are working on array. Such as: join(), reverse().

Join() : This method is used to join all the elements of the array like a single string and return the joint string. We can use the delimiter as argument of join(), that we want between the elements in the joint string. If we do not put any delimiter the default is comma and followed by a space.

e.g.

jname = address.join(); document.write(jname,”
”);

Reverse() : Using this method the elements of an array can be reversed.

e.g.

```
1. address.reverse();                    // it will reverse the array
2. rname1=address.reverse();             // it will reverse the array
                                         //address and stored them
                                         //into another array rname1
                                         document.write(rname,”<br/
                                         >”); // display array
3. rname2=address.reverse().join();      // it will reverse as well as
                                         // join the array and store
                                         //them to a string rname2
                                         document.write(rname,”<br/
                                         >”);
```

In the above example 1 the original array 'address' is reversed. In example 2, array is again reversed and store the reverse array into a new array rarray1. In example 3, only a reverse string is produced i.e. rname2.

Length property : In an array we can use the length property to find the size of the array.

e.g. n=address.length;



CHECK YOUR PROGRESS

Q3. i. What is an array?

.....

ii. Write down the syntax for dense array.

.....

15.7 FUNCTIONS

Functions are considered to be a subprogram or blocks of code, which performs a subtask. A function can return a value. Function is used to eliminate repetition of coding. It turns the program into a structured one which increases readability.

Functions are of two types :

- User-defined function.
- Built-in function

15.7.1 User-Defined Function

User defined functions are those function which are developed by the programmer itself in the coding. The user defined functions are to be declared first before using it. Once we declare the function we can call it by giving its name. A function is declared by using the keyword function.

Syntax :

function fun_name(parameter list delimited by comma)

```
{
 Body
}
```

Though a function can be declared anywhere within the HTML document yet it is better to place it in the header section. It is because, the header section is executed before the body section.

e.g.

```
<html><head><title>Example of Function:: Display Addition
chart
</title>
<script language="JavaScript”> var a=0;
function addchart(num)
   {
      var n=0,i=0; for(i=1;i<=10;i++)
      document.write(num,” + “,i,” = “,num+i,” <br/>”);
   }
  </script>
 </head>
 <body>
      <script language=”JavaScript”> a=prompt
      (“Enter a number”,a); a=parseIns(a);
      addchart(a);
  </script>
 </body>
</html>
```

15.7.2 Built-in Function

Built-in functions are those functions which are already developed by the language developer. They are the standard functions, which can be accessed at any time within the <script> tag. In JavaScript there are several such built-in functions. Such as:

eval() : It is used to convert a string value to a numeric value.

parseInt() : It is also used to convert a string value to a numeric value.

parseFloat() : It is also used to convert a string value to a float value.

15.7.3 Dialog Box

In JavaScript there are a few dialog boxes for displaying messages and content of variables. These dialog boxes are very effective as well as attractive for interacting with the user at run time. They appear as a separate window. e.g. the dialog boxes are: prompt, alert and confirmed.

prompt() : It is used to receive data from user at run time. It also displays a message.

Syntax :

prompt("message", "Default value");

alert() : It is used to display a small message in a separate window.

Syntax :

alart("message");

Confirm Dialog Box : It is used for the confirmation of the user. A confirmed dialog box may contain **ok** and **cancel** button.

Syntax :

confirm("message");



CHECK YOUR PROGRESS

Q.4. Fill in the blanks :

i. A function can return a _____.

ii. The user defined functions are to be _____ before using it.

iii. A function is declared by using the _____ function.



15.8 LET US SUM UP

- A HTML document is purely static. i.e. It can only present the documents.
- JavaScript is a high level object oriented language.
- JavaScript is case sensitive, i.e. it can understand the difference between the capital letter and small letters.
- When a variable is defined a memory location is assigned internally to it.
- Data type means types of data that can be held by a variable. In JavaScript there are four different primitive data types such as: Number, Boolean, String and Null.
- Looping means to execute the same set of statement repeatedly for several times (zero or more times).
- An array is a common name given to set of variables. It has an index number to be written within square brackets and started with the value 0.
- Functions are considered to be a subprogram or blocks of code, which performs a subtask. A function can return a value.
- Built-in functions are those which are already developed by the language developer. They are the standard functions, which can be accessed at any time within the <script> tag.



15.9 FURTHER READING

- **Web enabled commercial Application Development using HTML, Javascript, DHTML and PHP** - *Ivan Bayross*

15.10 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1 : i. static, ii. interpreted language, iii. <SCRIPT>

Ans. to Q. No. 2 : i. True, ii. False, iii. True, iv. False, v. False

Ans. to Q. No. 3 : i. An array is a common name given to set of variables
ii. arrayname = new Array(value0, value1, value2, … valuen-1)

Ans. to Q. No. 4 : i. Value, ii. declared first, iii. keyword

15.11 MODEL QUESTIONS

Q.1: What is JavaScript? Describe in brief.

Q.2: Write short note on :

a. Variable

b. Data type

c. Operators

Q.3: Write in short about the control structure in JavaScript.

Q.4: What is an array? What is its use?

Q.5: Write in short about Function.

Q.6: Write a function to find sum of two numbers using javascript

Q.7: Write a function to reverse the digits of a number using javascript (example 357 in reverese 753)

****