

## **2<sup>nd</sup> Semester MCA02005 Internet and Web Programming**

**L-T-P 3-0-0 3 CREDITS**

### **Module I (8 Periods)**

Internet Architecture: Internet overview, evolution of internet. Internet components: Local Area Networks, Access Networks, Core Networks, Routers, Transmission infrastructure, ISPs. TCP/IP model, TCP/IP vs OSI model. HTML: HTML Overview, Structure of HTML Documents, Document Types, HTML Elements and attributes. Anchor Attributes, Image Tag and its attributes, Image and Anchors, Table.

### **Module II (8 Periods)**

Image Map: Attributes, Client Side Image Maps and Server Side Maps.

HTML Layout: Background, colors and text, Tables, Frames, Layers, Page content Division <Div>, <SPAN>. CSS: Style Sheet Basic, Properties, Positioning with Style Sheet.

Forms: <FORM> Elements, Form controls. Dynamic HTML.

### **Module III (8 Periods)**

Java Script: Introduction, Client-Side JavaScript, Server-Side JavaScript, JavaScript Objects, JavaScript Security. Operators: Assignment Operators, Comparison Operators, Arithmetic Operators, Increment, Decrement, Unary Negation, Logical Operators, String Operators, Special Operators, Conditional operator, Comma operator, delete, new, this, void.

Statements: Break, comment, continue, delete, do ... while, export, for, for...in, function, if...else, import, labelled, return, switch, var, while.

### **Module IV (8 Periods)**

JavaScript (Properties and Methods of Each) :Array, Boolean, Date, Function, Math, Number, Object, String, RegExp. Document and its associated objects, document, Link, Area, Anchor, Image, Applet, Layer.

Events and Event Handlers: General Information about Events, Defining Event Handlers, event.

### **Module V (8 Periods)**

Server Side Programming: Common Gateway Interface (CGI), Active Server Pages.

Internet applications: FTP, Telnet, Email, Chat. World Wide Web: HTTP protocol. Search Engines. E-commerce and security issues including symmetric and asymmetric key, encryption and digital signature, and authentication. Emerging trends, Internet telephony, and virtual reality over the web, etc. Intranet and extranet, firewall.

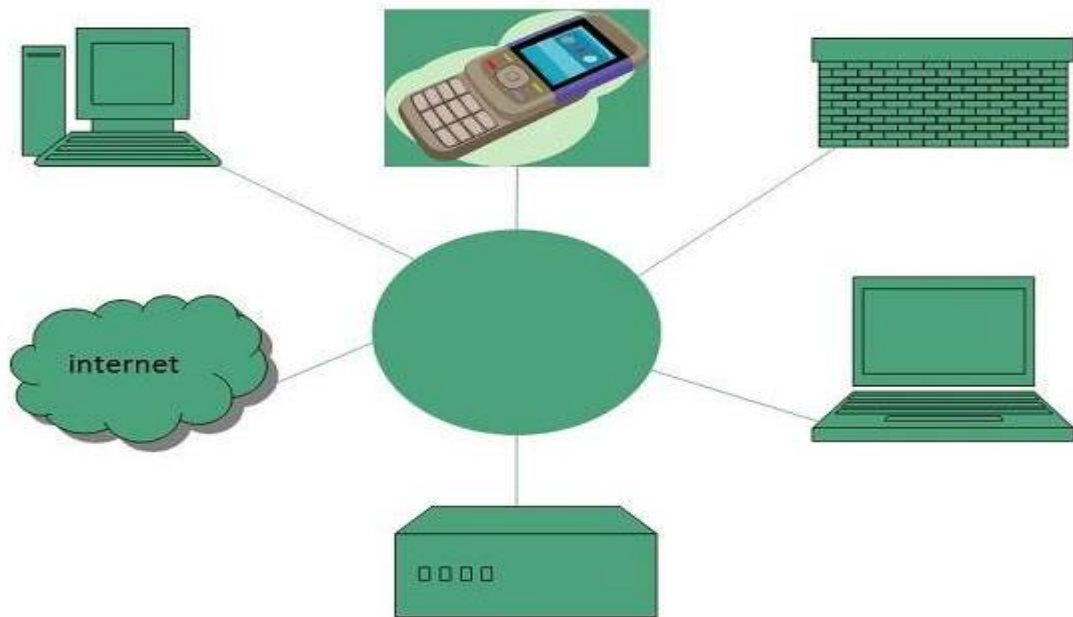
### **Books:**

1. Computer Networking: A Top-Down Approach Featuring the Internet by Kurose and Ross, Pearson.
2. Web Design the Complete Reference by Thomas Powell, Tata McGrawHill.
3. HTML The Complete Reference by Thomas Powell, Tata McGrawHill.

## Internet

Internet is defined as an Information super Highway, to access information over the web. However, It can be defined in many ways as follows:

- Internet is a world-wide global system of interconnected computer networks.
- Internet uses the standard Internet Protocol (TCP/IP).
- Every computer in internet is identified by a unique IP address.
- IP Address is a unique set of numbers (such as 110.22.33.114) which identifies a computer location.
- A special computer DNS (Domain Name Server) is used to give name to the IP Address so that user can locate a computer by a name.
- For example, a DNS server will resolve a name <http://www.tutorialspoint.com> to a particular IP address to uniquely identify the computer on which this website is hosted.
- Internet is accessible to every user all over the world.
- 



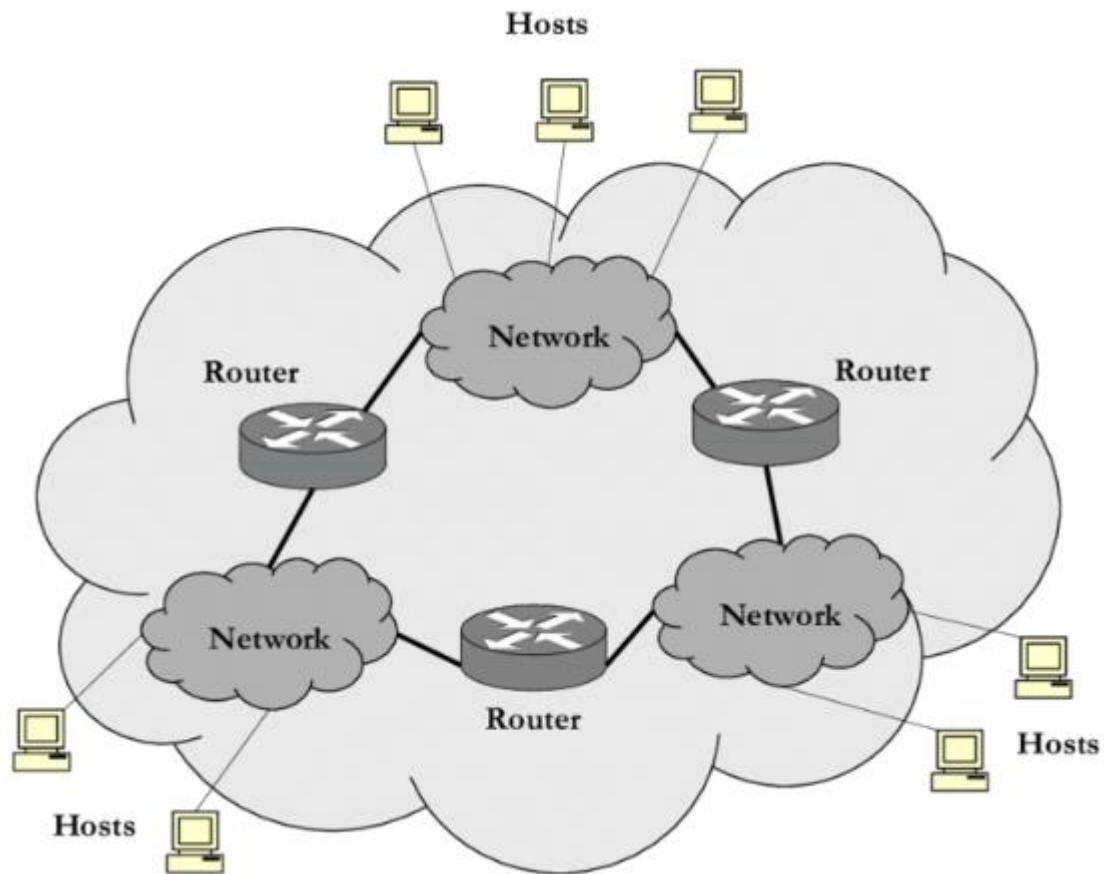
## Evolution

The concept of Internet was originated in 1969 and has undergone several technological & Infrastructural changes as discussed below:

- The origin of Internet devised from the concept of Advanced Research Project Agency Network (ARPANET).
- ARPANET was developed by United States Department of Defense.
- Basic purpose of ARPANET was to provide communication among the various bodies of government.
- Initially, there were only four nodes, formally called Hosts.
- In 1972, the ARPANET spread over the globe with 23 nodes located at different countries and thus became known as Internet.

- By the time, with invention of new technologies such as TCP/IP protocols, DNS, WWW, browsers, scripting languages etc. Internet provided a medium to publish and access information over the web.

### Internet components

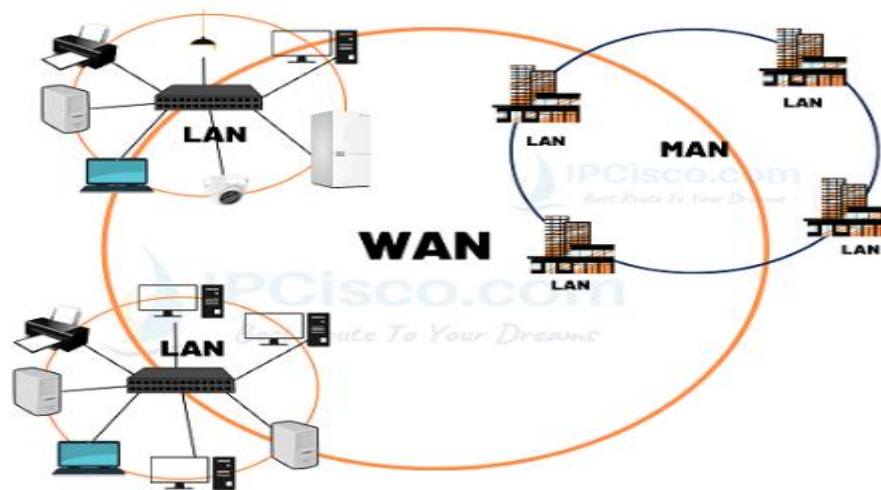


## Types of Networks

- LAN stands for **Local Area Network**.
- MAN stands for **Metropolitan Area Network**.
- WAN stands for **Wide Area Network**.

BASIS	LAN	MAN	WAN
Full Form	Local Area Network	Metropolitan Area Network	Wide Area Network
Range	A communication network linking a number of stations in same local area. Range is 1 to 10 km	This network shares the characteristics of packet broadcasting networks. Range is 100 km	A communication network distinguished from a Local Area Network. Range is Beyond 100 km
Media Used	Uses <b>guided media</b>	Uses <b>guided</b> as well as <b>unguided media</b>	Uses <b>unguided media</b>
Speed	A high speed i.e. 100kbps to 100mbps	Optimized for a <b>large geographical area</b> than LAN.	<b>Long distance</b> communications, which may or may not be provided by public packet network.
Cost	<b>cheaper</b>	<b>costly</b>	<b>expensive</b>
Equipment needed	NIC, switch and hub	Modem and router	Microwave, radio, infra-red laser
protocols	Attached Resource computer network (ARCNET), Token ring	Frame relay and asynchronous transfer mode(ATM)	ATM, FDDI, SMDS





## **Access Networks**

An access network is a type of network which physically connects an end system to the immediate router (also known as the “edge router”) on a path from the end system to any other distant end system. Examples of access networks are ISP, home networks, enterprise networks, ADSL, mobile network, FITH etc.

### **Types of access networks:**

#### **Ethernet –**

It is the most commonly installed wired LAN technology and it provides services on the Physical and Data Link Layer of OSI reference model. Ethernet LAN typically uses coaxial cable or twisted pair wires.

#### **DSL –**

DSL stands for Digital Subscriber Line and DSL brings a connection into your home through telephone lines and a DSL line can carry both data and voice signals and the data part of the line is continuously connected. In DSL you are able to use the Internet and make phone calls simultaneously. DSL modem uses the telephone lines to exchange data with digital subscriber line access multiplexer (DSLAMs). In DSL we get 24 Mbps downstream and 2.5 Mbps upstream.

#### **FTTH –**

Fiber to the home (FTTH) uses optical fiber from a central Office (CO) directly to individual buildings and it provides high-speed Internet access among all access networks. It ensures high initial investment but lesser future investment and it is the most expensive and most future-proof option amongst all these access networks.

#### **Wireless LANs –**

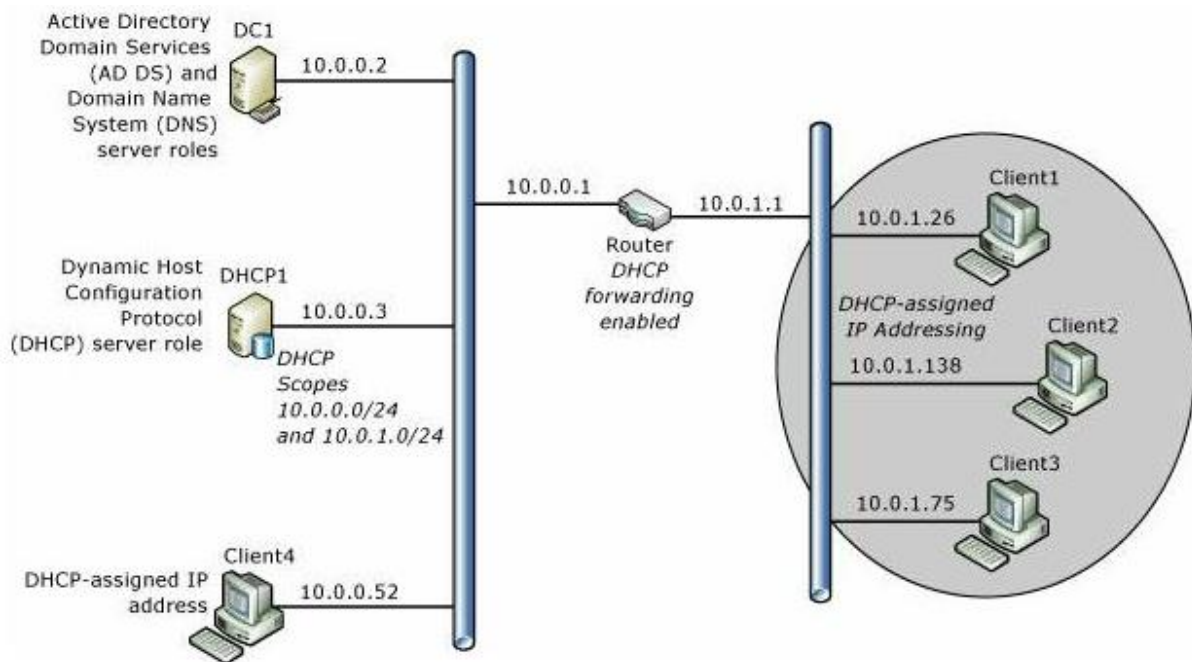
It links two or more devices using wireless communication within a range. It uses high-frequency radio waves and often include an access point for connecting to the Internet.

### 3G and LTE –

It uses cellular telephony to send or receive packets through a nearby base station operated by the cellular network provider. The term “3G internet” refers to the third generation of mobile phone standards as set by the International Telecommunications Union (ITU). Long Term Evolution (LTE) offers high-speed wireless communication for mobile devices and increased network capacity.

### Core Network Overview

The following illustration shows the Windows Server Core Network topology.



### Core Network Components

Following are the components of a core network.

#### Router

This deployment guide provides instructions for deploying a core network with two subnets separated by a router that has DHCP forwarding enabled. You can, however, deploy a Layer 2 switch, a Layer 3 switch, or a hub, depending on your requirements and resources. If you deploy a switch, the switch must be capable of DHCP forwarding or you must place a DHCP server on each subnet. If you deploy a hub, you are deploying a single subnet and do not need DHCP forwarding or a second scope on your DHCP server.

#### Static TCP/IP configurations

The servers in this deployment are configured with static IPv4 addresses. Client computers are configured by default to receive IP address leases from the DHCP server.

## **Active Directory Domain Services global catalog and DNS server DC1**

Both Active Directory Domain Services (AD DS) and Domain Name System (DNS) are installed on this server, named DC1, which provides directory and name resolution services to all computers and devices on the network.

## **DHCP server DHCP1**

The DHCP server, named DHCP1, is configured with a scope that provides Internet Protocol (IP) address leases to computers on the local subnet. The DHCP server can also be configured with additional scopes to provide IP address leases to computers on other subnets if DHCP forwarding is configured on routers.

## **Client computers**

Computers running Windows client operating systems are configured by default as DHCP clients, which obtain IP addresses and DHCP options automatically from the DHCP server.

## **Routers**

Routers are networking devices operating at layer 3 or a network layer of the OSI model. They are responsible for receiving, analysing, and forwarding data packets among the connected computer networks. When a data packet arrives, the router inspects the destination address, consults its routing tables to decide the optimal route and then transfers the packet along this route.

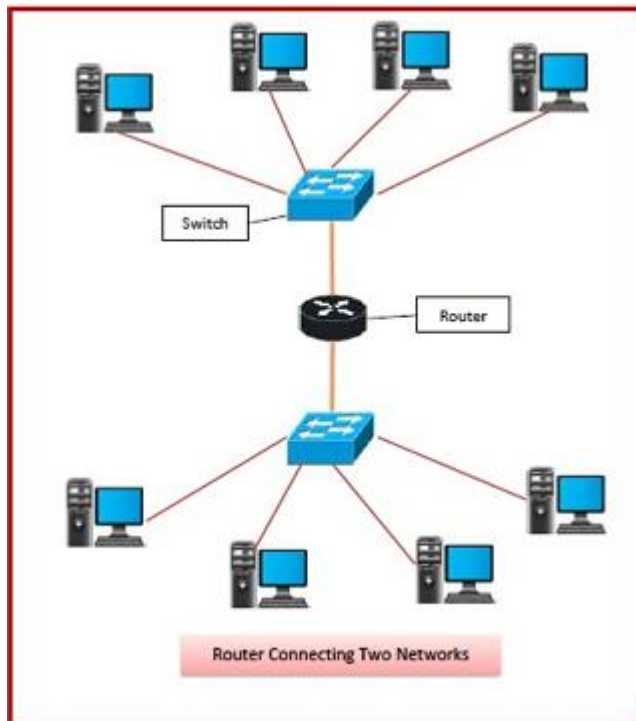
## **Features of Routers**

A router is a layer 3 or network layer device.

- It connects different networks together and sends data packets from one network to another.
- A router can be used both in LANs (Local Area Networks) and WANs (Wide Area Networks).
- It transfers data in the form of IP packets. In order to transmit data, it uses IP address mentioned in the destination field of the IP packet.
- Routers have a routing table in it that is refreshed periodically according to the changes in the network. In order to transmit data packets, it consults the table and uses a routing protocol.
- In order to prepare or refresh the routing table, routers share information among each other.
- Routers provide protection against broadcast storms.
- Routers are more expensive than other networking devices like hubs, bridges and switches.
- Routers are manufactured by some popular companies like –
  - Cisco
  - D-Link



HP  
3Com  
Juniper  
Nortel



## Types of Transmission Technology

Transmission is actually the process of sending and propagating analog or signals of digital information. Transmission technology generally refers to physical layer protocol duties like modulation, demodulation, line coding, and many more. It might also include higher-level protocol duties such as digitizing analog signals, data compression, etc.

### Types of Transmission Technology:

Transmission media is basically divided into two categories: Broadcast Networks, Point-to-Point Networks. These are explained as following below.

#### I. Broadcast Networks:

Broadcast networks are also known as terrestrial networks. It is basically a group of radio stations, television stations, or any other electronic media outlets that simply generate agreement to air, or broadcast, content generally from a centralized source. Broadcasting is simply a method of transferring messages to all the recipients simultaneously.



In this network, a message that is sent by a node is received by all the other nodes connected to the network and share a common medium of communication. Broadcast networks also avoid procedures of complex routing of switched network by simply confirming and ensuring that each transmission of nodes is basically received by all the other nodes in the network. This is the reason why the broadcast network has single communications channel.

In this network, each receiving station just receives all signals that are sent by transmitters. Even routing of signals is highly affected passively. These networks generally have single communication that is shared by all machines present on the network. Short messages also are known as packets that are sent by any of the machines present are received by all of the others present over there. Some of the systems of broadcast also support transmission to subset of machines also known as multicasting. It just links, in contrast, communication channel that is basically shared by all of machines in network.

#### **Advantages of Broadcast Networks –**

In this network, packets are generally transmitted and received by all of computers.

- It allows multicasting in the network.
- It has no limit. Even events can also run as long as required.
- It ensures better utilization of all resources available.

#### **Disadvantages of Broadcast Networks –**

- It cannot accommodate huge number of devices.
- It doesn't allow personalization of message.

#### **II.Point-to-Point Networks:**

Point-to-Point Networks or Point-to-Point Connection is type of private data connection that is connecting securely two or more locations for private data services. It might also be configured to usually carry voice, internet, and data services together all over same point-to-point network. It simply refers to type of communication connection among two endpoints or nodes of communication. It is connection among pairs of machines. Transmission from point-to-point with one sender and receiver is commonly known as unicasting.

This network is generally used for two locations that are required to securely send data that is very sensitive and confidential among each of locations. A point-to-point or P2P (Data Link) also gives or provides path from one point that is fixed to other point being fixed. It is very closed network data transport service that does not travel through public Internet. This network includes various connections

among individual pairs of machine. A packet present on these types of networks might be needed to go through intermediate computers before they reach desired or destination computer. The packets also need to follow multiple routes of different length sizes.

Therefore, routing algorithms are very essential and important in point-to-point connection. This network is generally available in range of bandwidth speeds along with point-to-point T1, point-to-point Ethernet, or many more.

#### **Advantages of Point-to-Point Networks –**

- It increases productivity.
- It generally uses leased lines so that speeds are guaranteed.
- It provides better security so that data can be transferred securely with confidence.

#### **Disadvantages of Point-to-Point Networks –**

- With this network, we can only connect two sites.
- It is very expensive for distant locations.

#### **ISP: Internet Service Provider**

ISP stands for Internet Service Provider. It is a company that provides access to the internet and similar services such as Website designing and virtual hosting. For example, when you connect to the Internet, the connection between your Internet-enabled device and the internet is executed through a specific transmission technology that involves the transfer of information packets through an Internet Protocol route.



Data is transmitted through different technologies, including cable modem, dial-up, DSL, high speed interconnects. Accordingly, based on the method of data transmission, the Internet access provided by ISPs can be divided into many types, some of which are as follows:

**Dial-up Internet access:** It is the oldest technology to provide Internet access by modem to modem connection using telephone lines. In this method, the user's computer is connected to a modem with a telephone line. This method has become outdated today due to slow connection speed. However, in remote areas, this method can be used where the broadband network is not available.

**DSL:** DSL, which stands for 'digital subscriber line' is an advanced version of the dial-up Internet access method. It uses high frequency to execute a connection over the telephone network and allows the internet and the phone connection to run on the same telephone line. This method offers an Asymmetric Digital Subscriber (ADSL), where the upload speed is less than the download speed, and a Symmetric Digital Subscriber Line (SDSL), which offers equal upload and download speeds. Out of these two, ADSL is more popular among users and is popularly known as DSL.

**Wireless Broadband (WiBB):** It is a modern broadband technology for Internet access. It allows high-speed wireless internet within a large area. To use this technology, you are required to place a dish on the top of your house and point it to the transmitter of your Wireless Internet Service Provider (WISP).

**Wi-Fi Internet:** It is the short form for "wireless fidelity," which is a wireless networking technology that provides wireless high-speed Internet connections using radio waves. To use the internet, you are required to be within the range of wi-fi network. It is commonly used in public places such as hotels, airports, restaurants to provide internet access to customers.

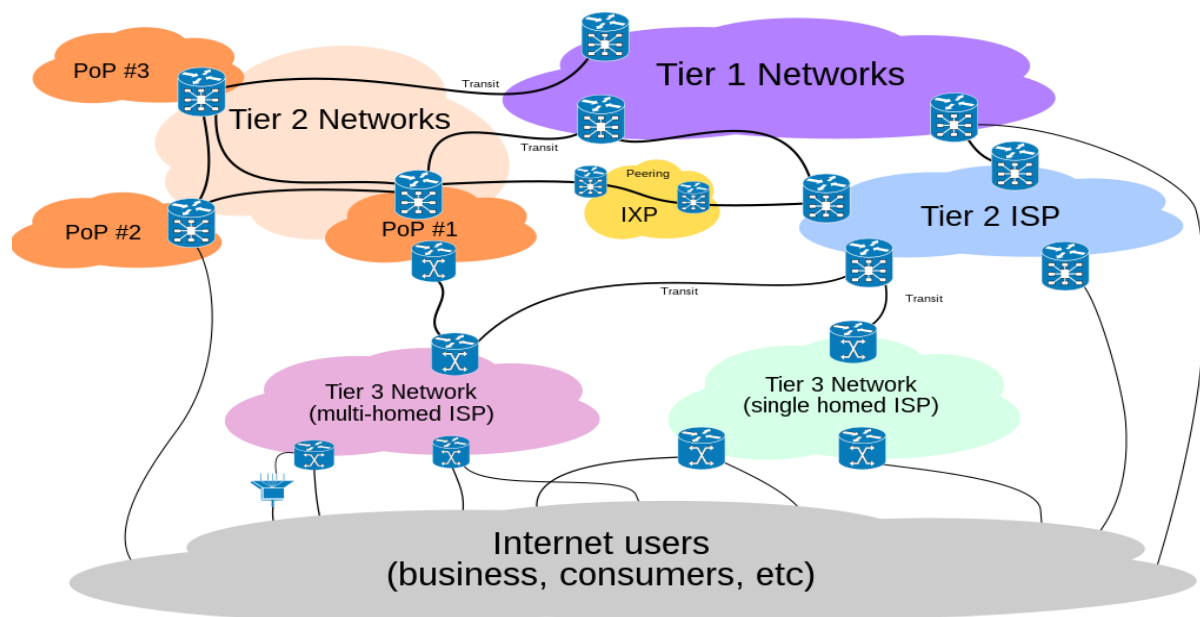
**ISDN:** It is a short form of Integrated Services Digital Network. It is a telephone system network which integrates a high-quality digital transmission of voice and data over the same standard phone line. It offers a fast upstream and downstream Internet connection speed and allows both voice calls and data transfer.

**Ethernet:** It is a wired LAN (Local Area Network) where computers are connected within a primary physical space. It enables devices to communicate with each other via a protocol (a set of rules or common network language). It may provide different speeds such as 10 Mbps, 100 Mbps and 10 Gbps.

## **Internet Backbone**

The Internet backbone may be defined by the principal data routes between large, strategically interconnected computer networks and core routers of the Internet. These data routes are hosted by commercial, government, academic and other high-capacity network centers, as well as the Internet exchange points and network access points, that exchange Internet traffic between the countries, continents, and across the oceans. Internet service providers, often Tier 1 networks, participate in Internet backbone traffic by privately negotiated interconnection agreements, primarily governed by the principle of settlement-free peering.

Tier 1 internet service providers (ISP) mesh their high-speed fiber-optic networks together to create the internet backbone, which moves traffic efficiently among geographic regions.



## What is POP in Internet?

On the Internet, a *point-of-presence (POP)* is an access point from one place to the rest of the Internet. A POP necessarily has a unique Internet Protocol (IP) address.

## How traffic gets on the backbone

- Below the Tier 1 ISPs are smaller Tier 2 and Tier 3 ISPs.
- Tier 3 providers provide businesses and consumers with access to the internet. These providers have no access of their own to the internet backbone, so on their own would not be able to connect their customers to all of the billions of internet-attached computers.
- Buying access to Tier 1 providers is expensive.
- So often Tier 3 ISPs contract with Tier 2 (regional) ISPs that have their own networks that can deliver traffic to a limited geographic area but not to all internet-attached devices.
- In order to do that, Tier 2 ISPs contract with Tier 1 ISPs for access to the global backbone, and in that way make the entire internet accessible to their customers.
- This arrangement makes it possible for traffic from a computer on one side of the world to connect to one on the other side. That traffic goes from a source computer to a Tier 3 ISP that routes it to a Tier 2 ISP that routes it to a Tier 1 backbone provider that routes it to the appropriate Tier 2 ISP that routes it to a Tier 3 access provider that delivers it to the destination computer.

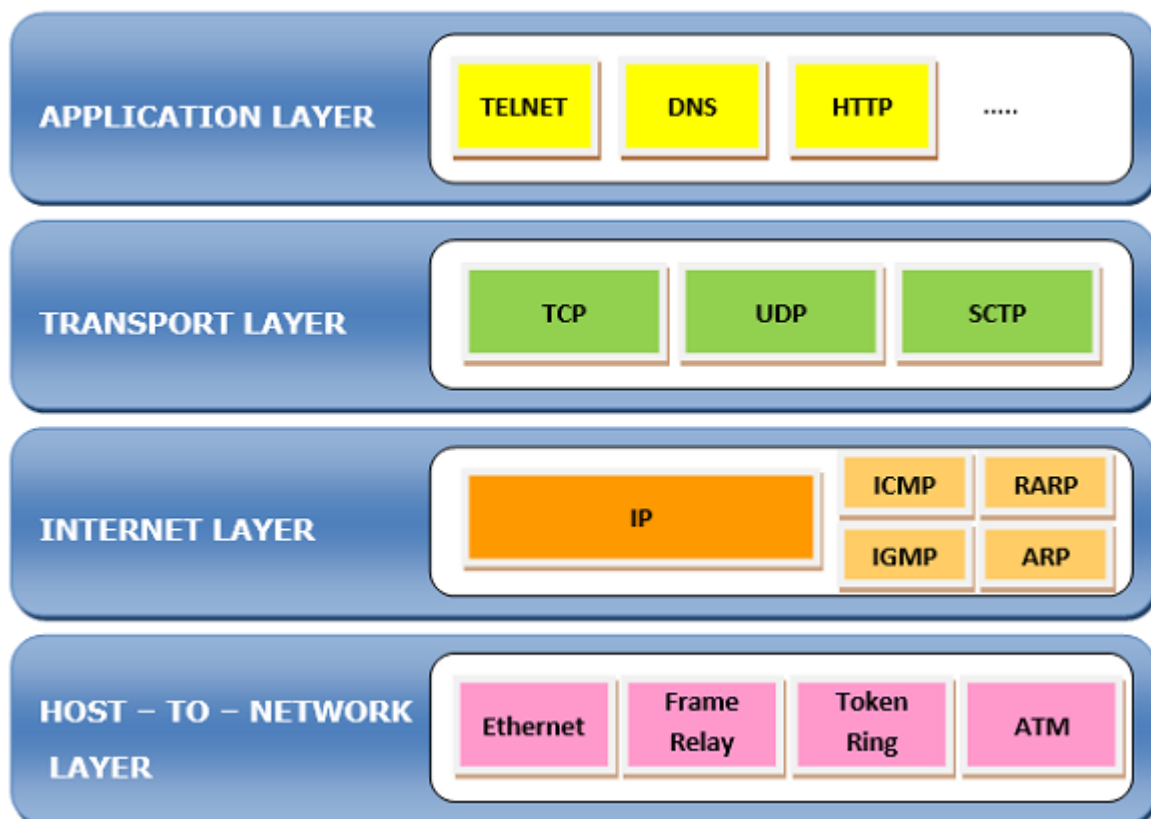
## The TCP/IP Reference Model

TCP/IP Reference Model is a four-layered suite of communication protocols. It was developed by the DoD (Department of Defence) in the 1960s. It is named after the two main protocols that are used in the model, namely, TCP and IP. TCP stands for Transmission Control Protocol and IP stands for Internet Protocol.

The four layers in the TCP/IP protocol suite are –

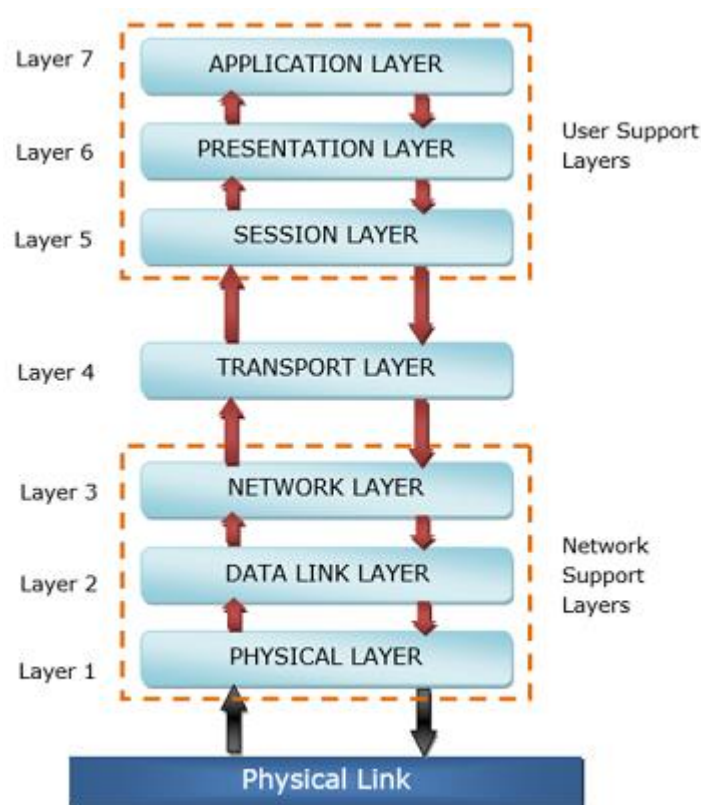
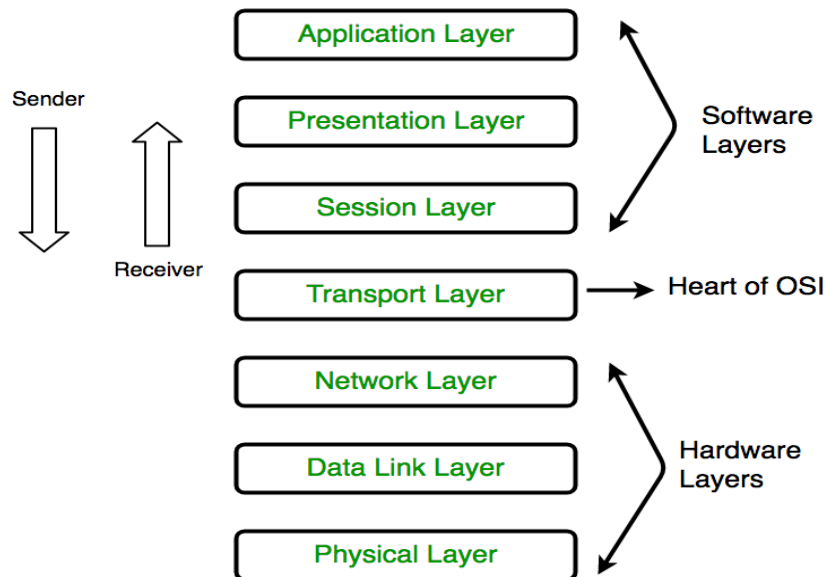
- **Host-to- Network Layer** –It is the lowest layer that is concerned with the physical transmission of data. TCP/IP does not specifically define any protocol here but supports all the standard protocols.
- **Internet Layer** –It defines the protocols for logical transmission of data over the network. The main protocol in this layer is Internet Protocol (IP) and it is supported by the protocols ICMP, IGMP, RARP, and ARP.
- **Transport Layer** – It is responsible for error-free end-to-end delivery of data. The protocols defined here are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).
- **Application Layer** – This is the topmost layer and defines the interface of host programs with the transport layer services. This layer includes all high-level protocols like Telnet, DNS, HTTP, FTP, SMTP, etc.

The following diagram shows the layers and the protocols in each of the layers –



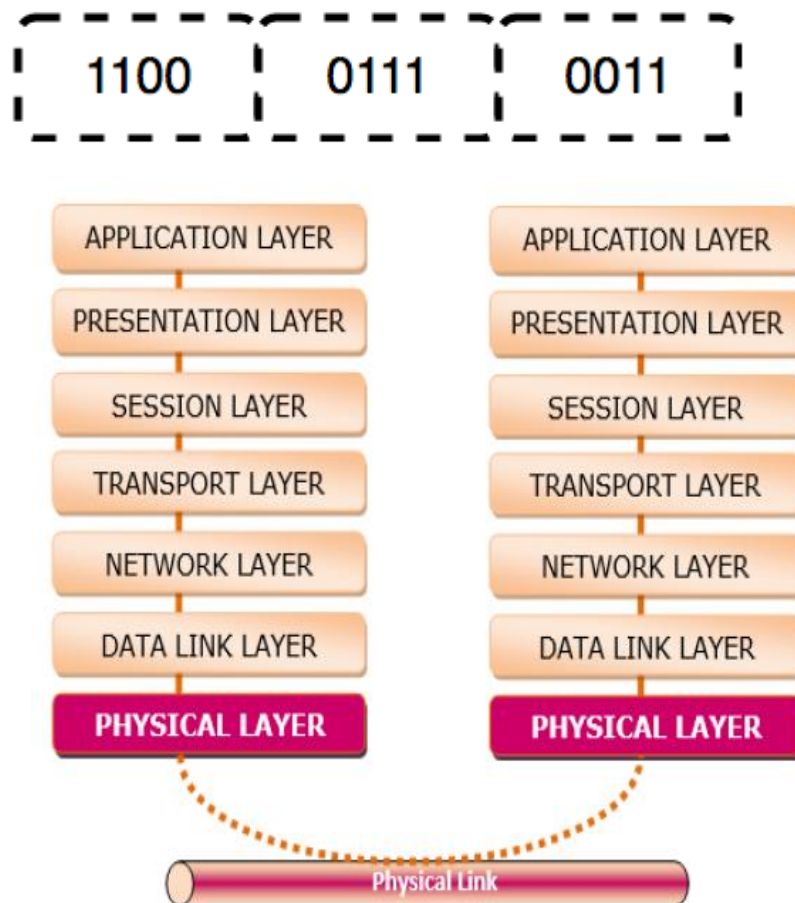
## Open Systems Interconnection (OSI) Model

OSI stands for **Open Systems Interconnection**. It has been developed by ISO – ‘**International Organization for Standardization**’, in the year 1984. It is a 7 layer architecture with each layer having specific functionality to perform. All these 7 layers work collaboratively to transmit the data from one person to another across the globe.



### Physical Layer (Layer 1):

The lowest layer of the OSI reference model is the physical layer. It is responsible for the actual physical connection between the devices. The physical layer contains information in the form of **bits**. It is responsible for transmitting individual bits from one node to the next. When receiving data, this layer will get the signal received and convert it into 0s and 1s and send them to the Data Link layer, which will put the frame back together.



The functions of the physical layer are as follows:

- **Bit synchronization:** The physical layer provides the synchronization of the bits by providing a clock. This clock controls both sender and receiver thus providing synchronization at bit level.
- **Bit rate control:** The Physical layer also defines the transmission rate i.e. the number of bits sent per second.
- **Physical topologies:** Physical layer specifies the way in which the different, devices/nodes are arranged in a network i.e. bus, star, or mesh topology.
- **Transmission mode:** Physical layer also defines the way in which the data flows between the two connected devices. The various transmission modes possible are Simplex, half-duplex and full-duplex.

\* Hub, Repeater, Modem, Cables are Physical Layer devices.

\*\* Network Layer, Data Link Layer, and Physical Layer are also known as **Lower Layers** or **Hardware Layers**.



## Data Link Layer (DLL) (Layer 2):

The data link layer is responsible for the node-to-node delivery of the message. The main function of this layer is to make sure data transfer is error-free from one node to another, over the physical layer. When a packet arrives in a network, it is the responsibility of DLL to transmit it to the Host using its MAC address.

Data Link Layer is divided into two sublayers:

- Logical Link Control (LLC)
- Media Access Control (MAC)

The packet received from the Network layer is further divided into frames depending on the frame size of NIC(Network Interface Card). DLL also encapsulates Sender and Receiver's MAC address in the header.

The Receiver's MAC address is obtained by placing an ARP(Address Resolution Protocol) request onto the wire asking "Who has that IP address?" and the destination host will reply with its MAC address.



The functions of the Data Link layer are:

- **Framing:** Framing is a function of the data link layer. It provides a way for a sender to transmit a set of bits that are meaningful to the receiver. This can be accomplished by attaching special bit patterns to the beginning and end of the frame.
- **Physical addressing:** After creating frames, the Data link layer adds physical addresses (MAC address) of the sender and/or receiver in the header of each frame.
- **Error control:** Data link layer provides the mechanism of error control in which it detects and retransmits damaged or lost frames.
- **Flow Control:** The data rate must be constant on both sides else the data may get corrupted thus, flow control coordinates the amount of data that can be sent before receiving acknowledgement.
- **Access control:** When a single communication channel is shared by multiple devices, the MAC sub-layer of the data link layer helps to determine which device has control over the channel at a given time.

\* *Packet in Data Link layer is referred to as **Frame**.*

\*\* Data Link layer is handled by the NIC (Network Interface Card) and device drivers of host machines.

\*\*\* Switch & Bridge are Data Link Layer devices.

## Network Layer (Layer 3):

The network layer works for the transmission of data from one host to the other located in different networks. It also takes care of packet routing i.e. selection of the shortest path to transmit the packet, from the number of routes available. The sender & receiver's IP addresses are placed in the header by the network layer.

The functions of the Network layer are:

- **Routing:** The network layer protocols determine which route is suitable from source to destination. This function of the network layer is known as routing.
- **Logical Addressing:** In order to identify each device on internetwork uniquely, the network layer defines an addressing scheme. The sender & receiver's IP addresses are placed in the header by the network layer. Such an address distinguishes each device uniquely and universally.

\* *Segment* in Network layer is referred to as **Packet**.



\*\* Network layer is implemented by networking devices such as routers.

#### **Transport Layer (Layer 4):**

The transport layer provides services to the application layer and takes services from the network layer. The data in the transport layer is referred to as *Segments*. It is responsible for the End to End Delivery of the complete message. The transport layer also provides the acknowledgement of the successful data transmission and re-transmits the data if an error is found.

- **At sender's side:** Transport layer receives the formatted data from the upper layers, performs **Segmentation**, and also implements **Flow & Error control** to ensure proper data transmission. It also adds Source and Destination port numbers in its header and forwards the segmented data to the Network Layer.

*Note: The sender needs to know the port number associated with the receiver's application.*

Generally, this destination port number is configured, either by default or manually. For example, when a web application makes a request to a web server, it typically uses port number 80, because this is the default port assigned to web applications. Many applications have default ports assigned

- **At receiver's side:** Transport Layer reads the port number from its header and forwards the Data which it has received to the respective application. It also performs sequencing and reassembling of the segmented data.

The functions of the transport layer are as follows:

- **Segmentation and Reassembly:** This layer accepts the message from the (session) layer, and breaks the message into smaller units. Each of the segments produced has a header associated with it. The transport layer at the destination station reassembles the message.
- **Service Point Addressing:** In order to deliver the message to the correct process, the transport layer header includes a type of address called service

point address or port address. Thus by specifying this address, the transport layer makes sure that the message is delivered to the correct process.

The services provided by the transport layer:

- **Connection-Oriented Service:** It is a three-phase process that includes
  - Connection Establishment
  - Data Transfer
  - Termination / disconnection

In this type of transmission, the receiving device sends an acknowledgement, back to the source after a packet or group of packets is received. This type of transmission is reliable and secure.

- **Connectionless service:** It is a one-phase process and includes Data Transfer. In this type of transmission, the receiver does not acknowledge receipt of a packet. This approach allows for much faster communication between devices. Connection-oriented service is more reliable than connectionless Service.

*\* Data in the Transport Layer is called as **Segments**.*

*\*\* Transport layer is operated by the Operating System. It is a part of the OS and communicates with the Application Layer by making system calls.*

*Transport Layer is called as **Heart of OSI model**.*

### **Session Layer (Layer 5):**

This layer is responsible for the establishment of connection, maintenance of sessions, authentication, and also ensures security.

The functions of the session layer are:

- **Session establishment, maintenance, and termination:** The layer allows the two processes to establish, use and terminate a connection.
- **Synchronization:** This layer allows a process to add checkpoints which are considered synchronization points into the data. These synchronization points help to identify the error so that the data is re-synchronized properly, and ends of the messages are not cut prematurely and data loss is avoided.
- **Dialog Controller:** The session layer allows two systems to start communication with each other in half-duplex or full-duplex.

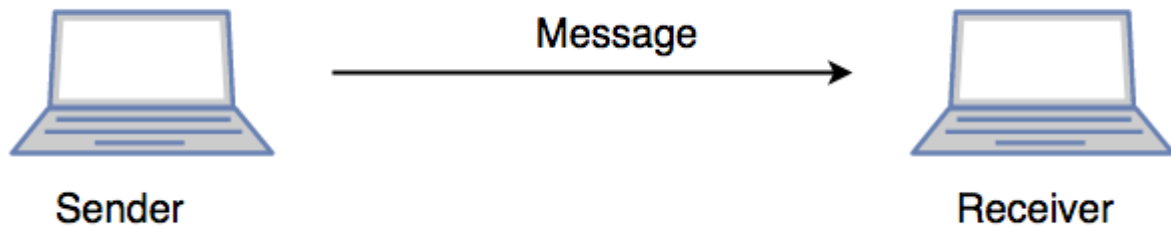
*\*\*All the below 3 layers (including Session Layer) are integrated as a single layer in the TCP/IP model as “Application Layer”.*

*\*\*Implementation of these 3 layers is done by the network application itself. These are also known as **Upper Layers** or **Software Layers**.*

### **Scenario:**

Let us consider a scenario where a user wants to send a message through some Messenger application running in his browser. The “Messenger” here acts as the application layer which provides the user with an interface to create the data. This

message or so-called Data is compressed, encrypted (if any secure data), and converted into bits (0's and 1's) so that it can be transmitted.



### **Presentation Layer (Layer 6):**

The presentation layer is also called the **Translation layer**. The data from the application layer is extracted here and manipulated as per the required format to transmit over the network.

The functions of the presentation layer are :

- **Translation:** For example, ASCII to EBCDIC.
- **Encryption/ Decryption:** Data encryption translates the data into another form or code. The encrypted data is known as the ciphertext and the decrypted data is known as plain text. A key value is used for encrypting as well as decrypting data.
- **Compression:** Reduces the number of bits that need to be transmitted on the network.

### **Application Layer (Layer 7):**

At the very top of the OSI Reference Model stack of layers, we find the Application layer which is implemented by the network applications. These applications produce the data, which has to be transferred over the network. This layer also serves as a window for the application services to access the network and for displaying the received information to the user.

Example: Application – Browsers, Skype Messenger, etc.

*\*\*Application Layer is also called Desktop Layer.*



The functions of the Application layer are :

- Network Virtual Terminal
- FTAM-File transfer access and management
- Mail Services
- Directory Services

## OSI model in a nutshell

No.	Layer Name	Responsibility	Information Form (Data Unit)	Device
7	Application Layer	Helps in identifying the client and synchronize communication	Message	-
6	Presentation Layer (Translation Layer)	Data from application layer is extracted and manipulated as required format for transmission	Message	-
5	Session Layer	Establishes connection, maintenance, authentication and ensure security	Message	Gateway
4	Transport Layer (HEART of OSI)	Take service from network layer and provide it to application layer	Segment	Firewall
3	Network Layer	Transmission of data from one host to other. Located in different network	Packet	Router
2	Data Link Layer	Node to node delivery of messages	Frame	Switch, Bridge
1	Physical Layer	Establishing physical connection between devices	Bits	Hub, Repeater, Modem, Cables

OSI model summarized (table form)

## OSI vs. TCP/IP Reference Model

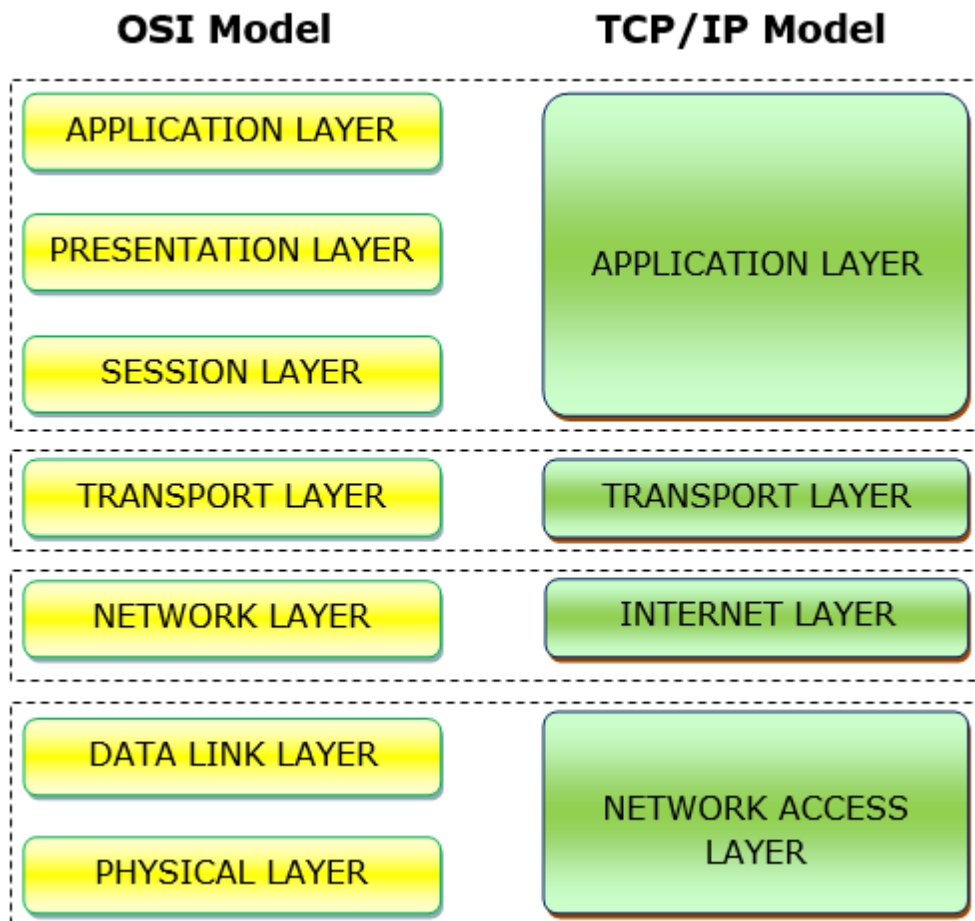
### Similarities between OSI and TCP / IP Reference Models

- Both the reference models are based upon layered architecture.
- The layers in the models are compared with each other. The physical layer and the data link layer of the OSI model correspond to the link layer of the TCP/IP model. The network layers and the transport layers are the same in both the models. The session layer, the presentation layer and the application layer of the OSI model together form the application layer of the TCP/IP model.
- In both the models, protocols are defined in a layer-wise manner.
- In both models, data is divided into packets and each packet may take the individual route from the source to the destination.

### Differences between OSI and TCP / IP Reference Models

- OSI model is a generic model that is based upon functionalities of each layer. TCP/IP model is a protocol-oriented standard.
- OSI model distinguishes the three concepts, namely, services, interfaces, and protocols. TCP/IP does not have a clear distinction between these three.
- OSI model gives guidelines on how communication needs to be done, while TCP/IP protocols layout standards on which the Internet was developed. So, TCP/IP is a more practical model.
- In OSI, the model was developed first and then the protocols in each layer were developed. In the TCP/IP suite, the protocols were developed first and then the model was developed.
- The OSI has seven layers while the TCP/IP has four layers.

The following diagram shows the corresponding layers of OSI and TCP/IP models



**N.B.OSI model** acts as a reference model and is not implemented on the Internet because of its late invention. The current model being used is the **TCP/IP model**. TCP/IP is the communication protocol suite that connects network devices to the Internet, while OSI is a reference model that outlines the functions of a networking system.

### **HTML - Overview**

HTML stands for **Hyper Text Markup Language**, which is the most widely used language on Web to develop web pages. HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

HTML stands for **Hypertext Markup Language**, and it is the most widely used language to write Web Pages.

- **Hypertext** refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.
- As its name suggests, HTML is a **Markup Language** which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

### **Why to Learn HTML?**

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.



### Some of the key advantages of learning HTML:

- **Create Web site –**  
You can create a website or customize an existing web template if you know HTML well.
- **Become a web designer –**  
If you want to start a career as a professional web designer, HTML and CSS designing is a must skill.
- **Understand web –**  
If you want to optimize your website, to boost its speed and performance, it is good to know HTML to yield best results.
- **Learn other languages –**  
Once you understand the basic of HTML then other related technologies like javascript, php, or angular are become easier to understand.

### Hello World using HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>CIME,Bhubaneswar</title>
  </head>
  <body>
    <h1>MCA 4th Semester</h1>
    <p> Internet and Web Programming(MCA02005)</p>
  </body>
</html>
```

## Applications of HTML

- **Web pages development –**

HTML is used to create pages which are rendered over the web. Almost every page of web is having html tags in it to render its details in browser.

- **Internet Navigation –**

HTML provides tags which are used to navigate from one page to another and is heavily used in internet navigation.

- **Responsive UI –**

HTML pages now-a-days works well on all platform, mobile, tabs, desktop or laptops owing to responsive design strategy.

- **Offline support**

HTML pages once loaded can be made available offline on the machine without any need of internet.

- **Game development-**

HTML5 has native support for rich experience and is now useful in gaming development arena as well.

## Basic HTML Document

In its simplest form, following is an example of an HTML document –

```
<!DOCTYPE html>
<html>

  <head>
    <title>This is document title</title>
  </head>

  <body>
    <h1>This is a heading</h1>
    <p>Document content goes here.....</p>
  </body>

</html>
```

## HTML Tags

As told earlier, HTML is a markup language and makes use of various tags to format the content. These tags are enclosed within angle braces **<Tag Name>**. Except few tags, most of the tags have their corresponding closing tags. For example, **<html>** has its closing tag **</html>** and **<body>** tag has its closing tag **</body>** tag etc.

Above example of HTML document uses the following tags –

Sr.No	Tag & Description
1	<b>&lt;!DOCTYPE...&gt;</b> This tag defines the document type and HTML version.
2	<b>&lt;html&gt;</b> This tag encloses the complete HTML document and mainly comprises of document header which is represented by <b>&lt;head&gt;...&lt;/head&gt;</b> and document body which is represented by <b>&lt;body&gt;...&lt;/body&gt;</b> tags.
3	<b>&lt;head&gt;</b> This tag represents the document's header which can keep other HTML tags like <b>&lt;title&gt;</b> , <b>&lt;link&gt;</b> etc.
4	<b>&lt;title&gt;</b> The <b>&lt;title&gt;</b> tag is used inside the <b>&lt;head&gt;</b> tag to mention the document title.
5	<b>&lt;body&gt;</b> This tag represents the document's body which keeps other HTML tags like <b>&lt;h1&gt;</b> , <b>&lt;div&gt;</b> , <b>&lt;p&gt;</b> etc.
6	<b>&lt;h1&gt;</b> This tag represents the heading.
7	<b>&lt;p&gt;</b> This tag represents a paragraph.

To learn HTML, you will need to study various tags and understand how they behave, while formatting a textual document. Learning HTML is simple as users have to learn the usage of different tags in order to format the text or images to make a beautiful webpage.

World Wide Web Consortium (W3C) recommends to use lowercase tags starting from HTML 4.

## HTML Document Structure

A typical HTML document will have the following structure –

```
<html>

    <head>
        Document header related tags
    </head>

    <body>
        Document body related tags
    </body>

</html>
```

### **The <!DOCTYPE> Declaration**

The <!DOCTYPE> declaration tag is used by the web browser to understand the version of the HTML used in the document. Current version of HTML is 5 and it makes use of the following declaration –

```
<!DOCTYPE html>
```

There are many other declaration types which can be used in HTML document depending on what version of HTML is being used.

## HTML Elements

An HTML element is defined by a start tag, some content, and an end tag.

### HTML Elements

The HTML **element** is everything from the start tag to the end tag:

<tagname>Content goes here...</tagname>

Examples of some HTML elements:

<h1>My First Heading</h1>

<p>My first paragraph.</p>

Start tag	Element content	End tag
<h1>	My First Heading	</h1>
<p>	My first paragraph.	</p>
 	<i>none</i>	<i>none</i>

## Nested HTML Elements

HTML elements can be nested (this means that elements can contain other elements).

All HTML documents consist of nested HTML elements.

The following example contains four HTML elements (<html>, <body>, <h1> and <p>):

### Example

<!DOCTYPE html>

<html>

<body>

<h1>My First Heading</h1>

<p>My first paragraph. </p>

</body>

</html>

### O/P

**My First Heading**

My first paragraph.

## Example Explained

The `<html>` element is the root element and it defines the whole HTML document.

It has a start tag `<html>` and an end tag `</html>`.

Then, inside the `<html>` element there is a `<body>` element:

```
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
</body>
```

The `<body>` element defines the document's body.

It has a start tag `<body>` and an end tag `</body>`.

Then, inside the `<body>` element there are two other elements: `<h1>` and `<p>`:

```
<h1>My First Heading</h1>
<p>My first paragraph.</p>
```

The `<h1>` element defines a heading.

It has a start tag `<h1>` and an end tag `</h1>`:

```
<h1>My First Heading</h1>
```

The `<p>` element defines a paragraph.

It has a start tag `<p>` and an end tag `</p>`:

```
<p>My first paragraph.</p>
```

## Never Skip the End Tag

Some HTML elements will display correctly, even if you forget the end tag:

Example

```
<html>
<body>
<p>This is a paragraph
<p>This is a paragraph
</body>
</html>
```

## O/P

This is a paragraph.

This is a paragraph.

However, never rely on this! Unexpected results and errors may occur if you forget the end tag!

### **Empty HTML Elements**

HTML elements with no content are called empty elements.

The `<br>` tag defines a line break, and is an empty element without a closing tag:

Example

```
<p>This is a <br> paragraph with a line break.</p>
```

**O/P**

**This is a  
paragraph with a line break.**

### **HTML is Not Case Sensitive**

HTML tags are not case sensitive: `<P>` means the same as `<p>`.

The HTML standard does not require lowercase tags, it is recommended lowercase in HTML, and demands lowercase for stricter document types like XHTML.

### **HTML Tag Reference**

Tag	Description
<code>&lt;html&gt;</code>	Defines the root of an HTML document
<code>&lt;body&gt;</code>	Defines the document's body
<code>&lt;h1&gt;</code> to <code>&lt;h6&gt;</code>	Defines HTML headings

### **HTML Attributes**

HTML attributes provide additional information about HTML elements.

### **HTML Attributes**

All HTML elements can have attributes

Attributes provide additional information about elements

Attributes are always specified in the start tag

Attributes usually come in name/value pairs like: `name="value"`

### **The href Attribute**

The `<a>` tag defines a hyperlink. The href attribute specifies the URL of the page the link goes to.



### Example-

```
<html>
<body>
<h1>The href Attribute</h1>
<a href="https://www.bput.ac.in">Visit BPUT</a>
</body>
</html>
```

(hypertext reference) HTML code used to create a link to another page.

### The src Attribute

The <img> tag is used to embed an image in an HTML page. The **src** attribute specifies the path to the image to be displayed:

Example

```

```

**Ex**

```
<!DOCTYPE html>
<html>
<body>
<h2>The src Attribute</h2>
<p>HTML images are defined with the img tag, and the filename of the image
source is specified in the src attribute:</p>

</body>
</html>
```

### The alt Attribute

The required alt attribute for the <img> tag specifies an alternate text for an image, if the image for some reason cannot be displayed. This can be due to slow connection, or an error in the src attribute, or if the user uses a screen reader.

**Example**

```

```

```
<!DOCTYPE html>
<html>
<body>

<p>If we try to display an image that does not exist, the value of the alt attribute
will be displayed instead. </p>
</body>
</html>
```

## The style Attribute

The style attribute is used to add styles to an element, such as color, font, size, and more.

Example

```
<p style="color:red;">This is a red paragraph.</p>
<!DOCTYPE html>
<html>
<body>
<h2>The style Attribute</h2>
<p>The style attribute is used to add styles to an element, such as color:</p>
<p style="color:red;">This is a red paragraph.</p>
</body>
</html>
```

**O/P**

## The style Attribute

The style attribute is used to add styles to an element, such as color:

**This is a red paragraph.**

## The lang Attribute

You should always include the **lang** attribute inside the <html> tag, to declare the language of the Web page. This is meant to assist search engines and browsers.

The following example specifies English as the language:

```
<!DOCTYPE html>
<html lang="en">
<body>
...
</body>
</html>
```

Country codes can also be added to the language code in the lang attribute. So, the first two characters define the language of the HTML page, and the last two characters define the country.

The following example specifies English as the language and United States as the country:

```
<!DOCTYPE html>
<html lang="en-US">
<body>
...
```

```
</body>
</html>
```

## **The title Attribute**

The title attribute defines some extra information about an element.

The value of the title attribute will be displayed as a tooltip when you mouse over the element:

### **Ex**

```
<!DOCTYPE html>
<html>
<body>
<h2 title="I'm a header">The title Attribute</h2>
<p title="I'm a tooltip">Mouse over this paragraph, to display the title attribute as a
tooltip.</p>
</body>
</html>
```

### **O/P**

## **The title Attribute**

Mouse over this paragraph, to display the title attribute as a tooltip.

## **<a>: The Anchor element**

The HTML anchor tag defines a hyperlink that links one page to another page. It can create hyperlink to other web page as well as files, location, or any URL. The "href" attribute is the most important attribute of the HTML a tag. and which links to destination page or URL.

The <a> HTML element (or anchor element), with its href attribute, creates a hyperlink to web pages, files, email addresses, locations in the same page, or anything else a URL can address.

Content within each <a> should indicate the link's destination.

If the href attribute is present, pressing the enter key while focused on the <a> element will activate it.

The <a> tag defines a hyperlink, which is used to link from one page to another.

The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

By default, links will appear as follows in all browsers:

- An unvisited link is underlined and blue

- A visited link is underlined and purple

- An active link is underlined and red

## HTML Table

HTML table tag is used to display data in tabular form (row \* column). There can be many columns in a row.

We can create a table to display data in tabular form, using `<table>` element, with the help of `<tr>`, `<td>`, and `<th>` elements.

In Each table, table row is defined by `<tr>` tag, table header is defined by `<th>`, and table data is defined by `<td>` tags.

HTML tables are used to manage the layout of the page e.g. header section, navigation bar, body content, footer section etc. But it is recommended to use `div` tag over `table` to manage the layout of the page .

## HTML Table Tags

Tag	Description
<code>&lt;table&gt;</code>	It defines a table.
<code>&lt;tr&gt;</code>	It defines a row in a table.
<code>&lt;th&gt;</code>	It defines a header cell in a table.
<code>&lt;td&gt;</code>	It defines a cell in a table.
<code>&lt;caption&gt;</code>	It defines the table caption.
<code>&lt;colgroup&gt;</code>	It specifies a group of one or more columns in a table for formatting.
<code>&lt;col&gt;</code>	It is used with <code>&lt;colgroup&gt;</code> element to specify column properties for each column.
<code>&lt;tbody&gt;</code>	It is used to group the body content in a table.
<code>&lt;thead&gt;</code>	It is used to group the header content in a table.
<code>&lt;tfooter&gt;</code>	It is used to group the footer content in a table.

## HTML Table Example

Let's see the example of HTML table tag. Its output is shown above.

**Problem :** Create a table to describe

Header name :	First_Name	Last_Name	Marks
Rows	Sonoo	Jaiswal	60
	James	William	80
	Swati	Sironi	82
	Chetna	Singh	72

**Ans:**

```
<html>
<body>
<table>
<tr><th>First_Name</th><th>Last_Name</th><th>Marks</th></tr>
<tr><td>Sonoo</td><td>Jaiswal</td><td>60</td></tr>
<tr><td>James</td><td>William</td><td>80</td></tr>
<tr><td>Swati</td><td>Sironi</td><td>82</td></tr>
<tr><td>Chetna</td><td>Singh</td><td>72</td></tr>
</table>
</body>
</html>
```

**O/P**

First_Name	Last_Name	Marks
Sonoo	Jaiswal	60
James	William	80
Swati	Sironi	82
Chetna	Singh	72

## HTML Table with Border

### HTML Border attribute

The HTML `<table>` border Attribute is used to specify the border of a table. It sets the border around the table cells.

### Syntax:

**`<table border="1|0">`**

### Attribute Values:

1: It sets the border around the table cells.

0: It removes (not set) the border around the table cells.

### Problem

First_Name	Last_Name	Marks
Sonoo	Jaiswal	60
James	William	80
Swati	Sironi	82
Chetna	Singh	72

### Answer

```
<html>
<body>
<table border="1">
<tr><th>First_Name</th><th>Last_Name</th><th>Marks</th></tr>
<tr><td>Sonoo</td><td>Jaiswal</td><td>60</td></tr>
<tr><td>James</td><td>William</td><td>80</td></tr>
<tr><td>Swati</td><td>Sironi</td><td>82</td></tr>
<tr><td>Chetna</td><td>Singh</td><td>72</td></tr>
</table>
</body>
</html>
```

**O/P:**

First_Name	Last_Name	Marks
Sonoo	Jaiswal	60
James	William	80
Swati	Sironi	82
Chetna	Singh	72

## HTML Table Sizes

### HTML Table Width

To set the width of a table, add the style attribute to the <table> element:

#### Example

**Set the width of the table to 100%:**

```
<html>
<body>
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
</body>
</html>
```



## Example

**Set the width of the first column to 70%:**

```
<table style="width:100%">
  <tr>
    <th style="width:70%">Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

## HTML Table Row Height

To set the height of a specific row, add the style attribute on a table row element:

**Example Set the height of the second row to 200 pixels:**

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr style="height:200px">
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

## HTML Table - Colspan

To make a cell span over multiple columns, use the colspan attribute:

### Example

```
<table>
  <tr>
    <th colspan="2">Name</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>43</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>57</td>
  </tr>
</table>
```

Note: The value of the colspan attribute represents the number of columns to span.

## HTML Table - Rowspan

To make a cell span over multiple rows, use the rowspan attribute:

### Example

```
<table>
  <tr>
    <th>Name</th>
    <td>Jill</td>
  </tr>
  <tr>
    <th rowspan="2">Phone</th>
    <td>555-1234</td>
  </tr>
  <tr>
    <td>555-8745</td>
  </tr>
</table>
```

## HTML Form

An **HTML form** is a section of a document which contains controls such as text fields, password fields, checkboxes, radio buttons, submit button, menus etc.

An HTML form facilitates the user to enter data that is to be sent to the server for processing such as name, email address, password, phone number, etc.

## Why use HTML Form

HTML forms are required if you want to collect some data from of the site visitor.

For example: If a user want to purchase some items on internet, he/she must fill the form such as shipping address and cred7t/debit card details so that item can be sent to the given address.

## HTML Form Syntax

```
<form action="server url" method="get|post">
```

```
//input controls e.g. textfield, textarea, radiobutton, button
```

```
</form>
```

## HTML Form Tags

Let's see the list of HTML 5 form tags.

Tag	Description
<form>	It defines an HTML form to enter inputs by the used side.
<input>	It defines an input control.
<textarea>	It defines a multi-line input control.
<label>	It defines a label for an input element.
<fieldset>	It groups the related element in a form.
<legend>	It defines a caption for a <fieldset> element.
<select>	It defines a drop-down list.
<optgroup>	It defines a group of related options in a drop-down list.
<option>	It defines an option in a drop-down list.
<button>	It defines a clickable button.

<datalist>	It specifies a list of pre-defined options for input control.
<keygen>	It defines a key-pair generator field for forms.
<output>	It defines the result of a calculation.

### HTML <form> element

The HTML <form> element provide a document section to take input from user. It provides various interactive controls for submitting information to web server such as text field, text area, password field, etc.

The <form> element does not itself create a form but it is container to contain all required form elements, such as <input>, <label>, etc.

### Syntax:

```
<form>
//Form elements
</form>
```

### Get vs. Post

There are many differences between the Get and Post request. Let's see these differences:

GET	POST
1) In case of Get request, only <b>limited amount of data</b> can be sent because data is sent in header.	In case of post request, <b>large amount of data</b> can be sent because data is sent in body.
2) Get request is <b>not secured</b> because data is exposed in URL bar.	Post request is <b>secured</b> because data is not exposed in URL bar.
3) Get request <b>can be bookmarked</b> .	Post request <b>cannot be bookmarked</b> .
4) Get request is <b>idempotent</b> . It means second request will be ignored until response of first request is delivered	Post request is <b>non-idempotent</b> .
5) Get request is <b>more efficient</b> and used more than Post.	Post request is <b>less efficient</b> and used less than get.

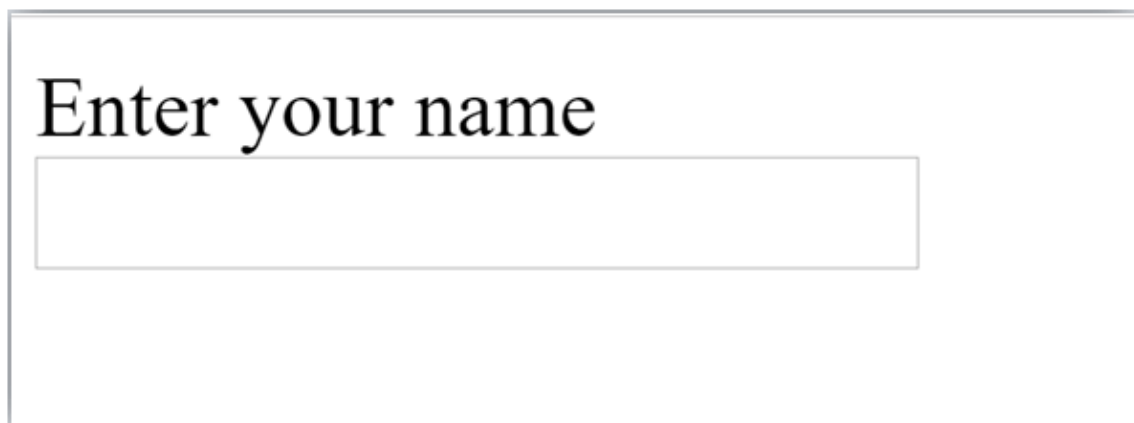
## HTML <input> element

The HTML <input> element is fundamental form element. It is used to create form fields, to take input from user. We can apply different input field to gather different information from user. Following is the example to show the simple text input.

### Example:

```
<html>
<body>
  <form>
    Enter your name <br>
    <input type="text" name="username">
  </form>
</body>
</html>
```

### Output:

The screenshot shows a web browser window with a simple form. The text "Enter your name" is displayed in a large, black, serif font. Below the text is a single-line text input field with a light gray border and a white background. The entire form is enclosed in a larger rectangular frame with a thin gray border.

## HTML TextField Control

The type="text" attribute of input tag creates textfield control also known as single line textfield control. The name attribute is optional, but it is required for the server side component such as JSP, ASP, PHP etc.

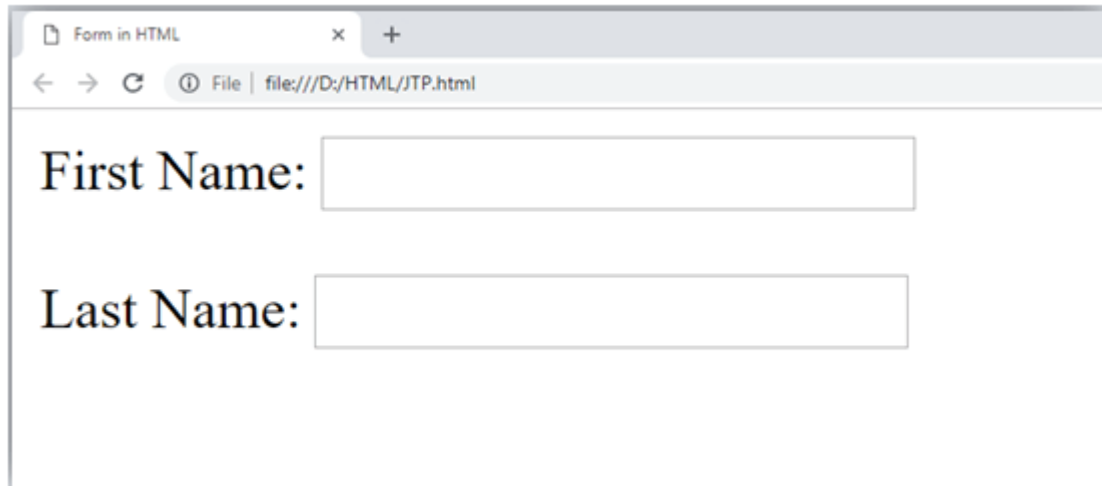
### <form>

First Name: <input type="text" name="firstname"/> <br/>

Last Name: <input type="text" name="lastname"/> <br/>

</form>

### Output:

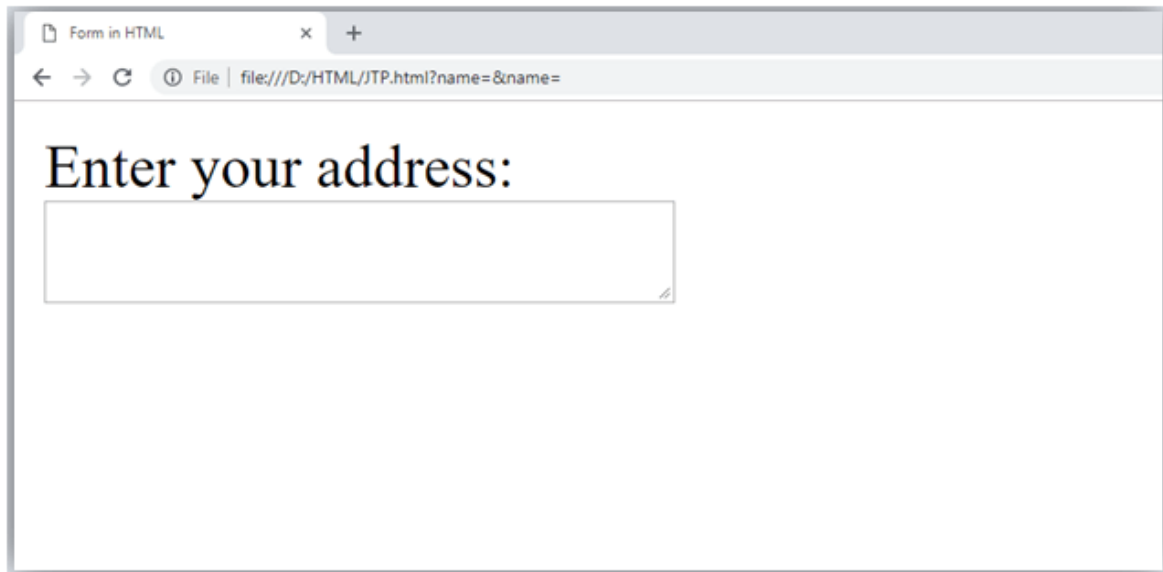
A screenshot of a web browser window. The title bar says "Form in HTML". The address bar shows "File | file:///D:/HTML/JTP.html". The main content area contains two text input fields. The first field is preceded by the label "First Name:" and the second field is preceded by the label "Last Name:". Both labels and fields are in a simple, clean font.

### HTML `<textarea>` tag in form

The `<textarea>` tag in HTML is used to insert multiple-line text in a form. The size of `<textarea>` can be specify either using "rows" or "cols" attribute or by CSS.

#### Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Form in HTML</title>
</head>
<body>
  <form>
    Enter your address:<br>
    <textarea rows="2" cols="20"></textarea>
  </form>
</body>
</html>
```



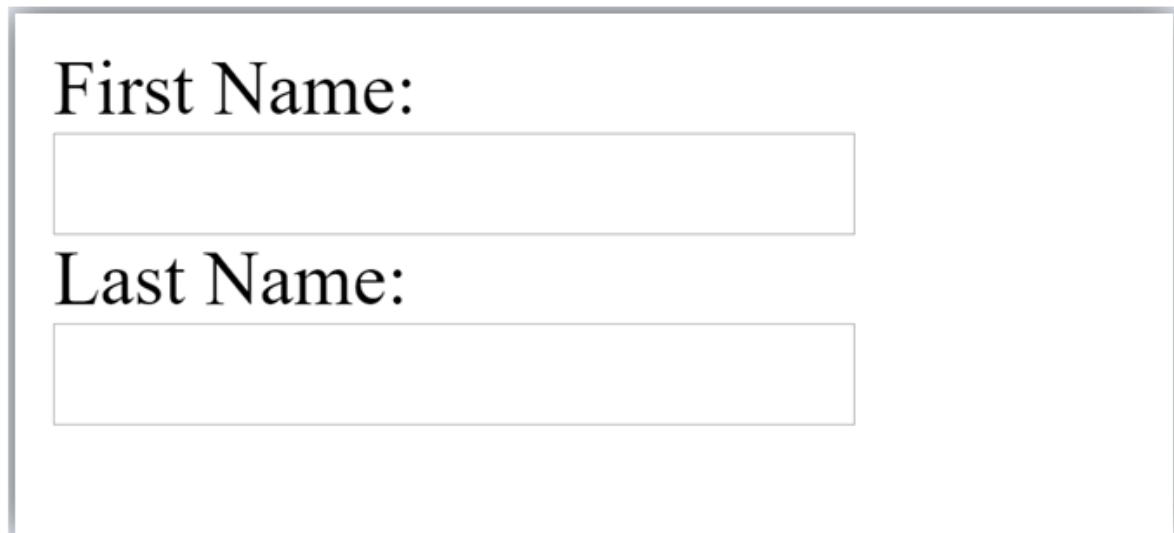
### Label Tag in Form

It is considered better to have label in form. As it makes the code parser/browser/user friendly.

If you click on the label tag, it will focus on the text control. To do so, you need to have for attribute in label tag that must be same as id attribute of input tag.

```
<form>
  <label for="firstname">First Name: </label> <br/>
    <input type="text" id="firstname" n
="firstname"/> <br/>
  <label for="lastname">Last Name: </label>
    <input type="text" id="lastname" name="lastname"/> <br/>
</form>
```

**Output:**

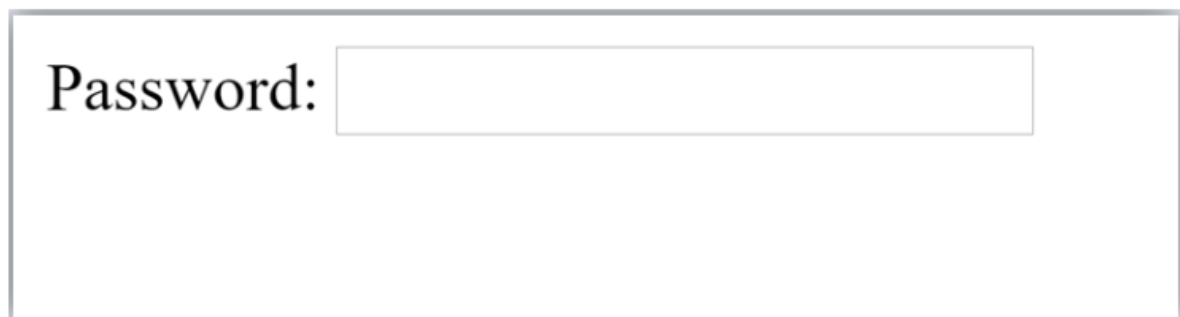
A rectangular box containing two text input fields. The first field is preceded by the label "First Name:" and the second field is preceded by the label "Last Name:". Both labels and fields are left-aligned.

### HTML Password Field Control

The password is not visible to the user in password field control.

```
<form>  
  <label for="password">Password: </label>  
    <input type="password" id="password" name="password"/> <br/>  
</form>
```

**Output:**

A rectangular box representing the browser output. It contains the text "Password:" followed by an empty password input field.

### HTML 5 Email Field Control

The email field is new in HTML 5. It validates the text for correct email address. You must use @ and . in this field.

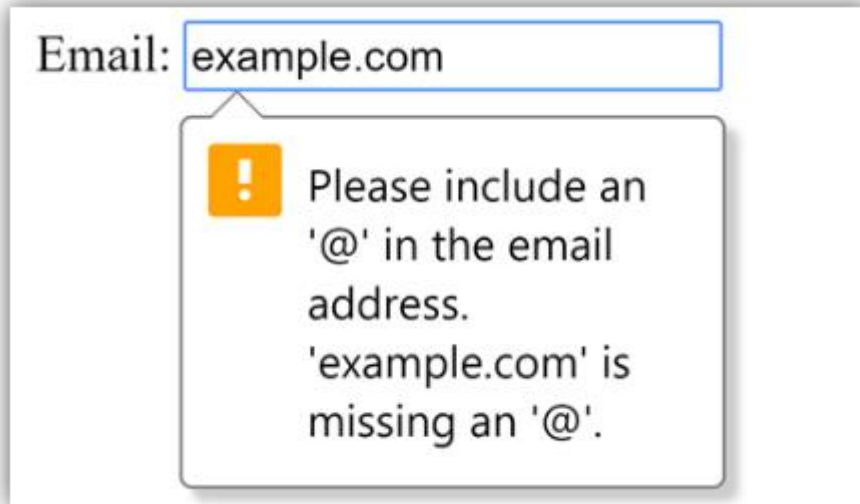
```
<form>  
  <label for="email">Email: </label>  
    <input type="email" id="email" name="email"/> <br/>  
</form>
```

It will display in browser like below:



Email:

Note: If we will not enter the correct email, it will display error like:



### Radio Button Control

The radio button is used to select one option from multiple options. It is used for selection of gender, quiz questions etc.

If you use one name for all the radio buttons, only one radio button can be selected at a time.

Using radio buttons for multiple options, you can only choose a single option at a time.

```
<form>
<!DOCTYPE html>
<html>
<body>
<h2>Radio Buttons</h2>
<p>Choose your favorite Web language:</p>
</form>
```

```

<input type="radio" id="html" name="fav_language1" value="HTML">
<label for="html">HTML</label><br>
<input type="radio" id="css" name="fav_language2" value="CSS">
<label for="css">CSS</label><br>
<input type="radio" id="javascript" name="fav_language3" value="JavaScript">
<label for="javascript">JavaScript</label>
</form>
</body>
</html>
</form>

```

**Output:**

## Radio Buttons

Choose your favorite Web language:

- ☐ HTML
- ☐ CSS
- ☐ JavaScript

## Checkbox Control

The checkbox control is used to check multiple options from given checkboxes.

```

<form>
Hobby:<br>
    <input type="checkbox" id="cricket" name="cricket" value="cricket"/>
    <label for="cricket">Cricket</label> <br>
    <input type="checkbox" id="football" name="football" value="football"/>
    <label for="football">Football</label> <br>
    <input type="checkbox" id="hockey" name="hockey" value="hockey"/>
    <label for="hockey">Hockey</label>
</form>

```

Note: These are similar to radio button except it can choose multiple options at a time and radio button can select one button at a time, and its display.

**Output:**

Hobby:

☒ Cricket

☒ Football

☐ Hockey

### The <select> Element

The <select> element defines a drop-down list. The <option> elements defines an option that can be selected.

By default, the first item in the drop-down list is selected.

```
<form>
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars">
    <option value="volvo">Volvo</ option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select>
</form>
```

### Visible Values:

Use the size attribute to specify the number of visible values.

**Syntax:** <select id="cars" name="cars" size="3">

### Allow Multiple Selections:

Use the multiple attribute to allow the user to select more than one value.

**Syntax:**<select id="cars" name="cars" size="4" multiple>

### Submit button control

HTML <input type="submit"> are used to add a submit button on web page. When user clicks on submit button, then form get submit to the server.

Syntax:

```
<input type="submit" value="submit">
```

The type = submit , specifying that it is a submit button

The value attribute can be anything which we write on button on web page.

The name attribute can be omit here.

Example:

```
<form>
```

```
  <label for="name">Enter name</label><br>
```

```
  <input type="text" id="name" name="name"><br>
```

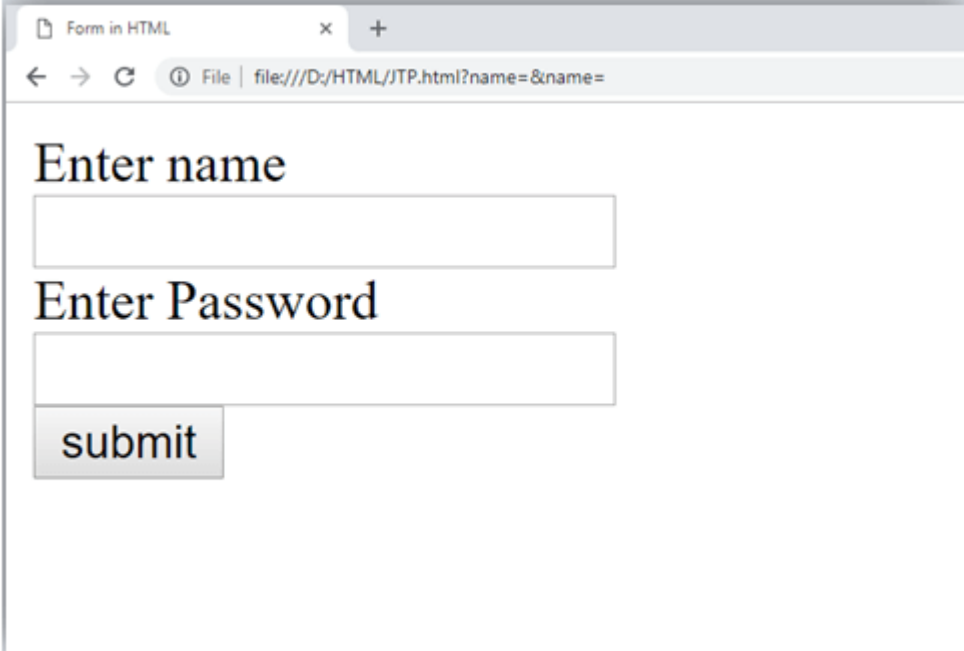
```
  <label for="pass">Enter Password</label><br>
```

```
  <input type="Password" id="pass" name="pass"><br>
```

```
  <input type="submit" value="submit">
```

```
</form>
```

**Output:**

A screenshot of a web browser window titled "Form in HTML". The address bar shows the file path "file:///D:/HTML/JTP.html?name=&name=". The form contains two text input fields. The first field is preceded by the label "Enter name". The second field is preceded by the label "Enter Password". Below the second input field is a button labeled "submit".

**HTML <fieldset> element:**

The <fieldset> element in HTML is used to group the related information of a form. This element is used with <legend> element which provide caption for the grouped elements.

Example:

```
<form>
  <fieldset>
    <legend>User Information:</legend>
    <label for="name">Enter name</label><br>
    <input type="text" id="name" name="name"><br>
    <label for="pass">Enter Password</label><br>
    <input type="Password" id="pass" name="pass"><br>
    <input type="submit" value="submit">
  </fieldset>
</form>
```

## Output:

The image shows a rendered HTML form. It has a legend titled "User Information:". Below the legend, there are two text input fields. The first is labeled "Enter name" and the second is labeled "Enter Password". Below these fields is a button labeled "submit". The entire form is enclosed in a rectangular border.

## HTML Form Example1

# The form element

First name:

Last name:

Click the "Submit" button and the form-data will be sent to a page on the server called "action\_page.php".

```
<!DOCTYPE html>
<html>
<body>
```

<h1>The form element</h1>

```
<form action="/action_page.php">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <label for="lname">Last name:</label>
  <input type="text" id="lname" name="lname"><br><br>
  <input type="submit" value="Submit">
</form>
```

<p>Click the "Submit" button and the form-data will be sent to a page on the server called "action\_page.php".</p>

</body>

</html>

## HTML Form Example2

### Registration form

User personal information

Enter your full name

Enter your email

Enter your password

confirm your password

Enter your gender

☐ Male

☐ Female

☐ others

Enter your Address:

<!DOCTYPE html>

<html>

<head>

<title>Form in HTML</title>

</head>

<body>

```

<h2>Registration form</h2>
<form>
  <fieldset>
    <legend>User personal information</legend>
    <label>Enter your full name</label><br>
    <input type="text" name="name"><br>
    <label>Enter your email</label><br>
    <input type="email" name="email"><br>
    <label>Enter your password</label><br>
    <input type="password" name="pass"><br>
    <label>confirm your password</label><br>
    <input type="password" name="pass"><br>
    <br><label>Enter your gender</label><br>
    <input type="radio" id="gender" name="gender" value="male"/>Male <br>
    <input type="radio" id="gender" name="gender" value="female"/>Female
  <br/>
    <input type="radio" id="gender" name="gender" value="others"/>others
  <br/>
    <br>Enter your Address:<br>
    <textarea></textarea><br>
    <input type="submit" value="sign-up">
  </fieldset>
</form>
</body>
</html>

```

### HTML Form Example3

Enter name:	<input type="text"/>
Enter password:	<input type="password"/>
Enter Email:	<input type="email"/>
Enter Gender:	<input type="radio"/> male <input type="radio"/> female
Select Country:	<input type="text" value="india"/> ▼
<input type="button" value="register"/>	

```
<!DOCTYPE>
<html>
<body>
<form action="http://www.javatpoint.com/javascriptpages/valid.jsp">
<table>
<tr>
    <td class="tdLabel"><label for="register_name" class="label">Enter
name:</label></td>
    <td><input type="text" name="name" value="" id="register_name"
style="width:160px"/></td>
</tr>
<tr>
    <td class="tdLabel"><label for="register_password" class="label">Enter
password:</label></td>
    <td><input type="password" name="password" id="register_password"
style="width:160px"/></td>
</tr>
<tr>
    <td class="tdLabel"><label for="register_email" class="label">Enter
Email:</label></td>
    <td>
<input type="email" name="email" value="" id="register_email"
style="width:160px"/></td>
</tr>
<tr>
    <td class="tdLabel"><label for="register_gender" class="label">Enter
Gender:</label></td>
    <td>
<input type="radio" name="gender" id="register_gendermale" value="male"/>
<label for="register_gendermale">male</label>
<input type="radio" name="gender" id="register_genderfemale" value="female"/>
<label for="register_genderfemale">female</label>
</td>
</tr>
<tr>
    <td class="tdLabel"><label for="register_country" class="label">Select
Country:</label></td>
    <td><select name="country" id="register_country" style="width:160px">
    <option value="india">india</option>
```



```

    <option value="pakistan">pakistan</option>
    <option value="africa">africa</option>
    <option value="china">china</option>
    <option value="other">other</option>
</select>
</td>
</tr>
<tr>
    <td colspan="2"><div align="right"><input type="submit" id="register_0"
value="register"/>
</div></td>
</tr>
</table>
</form>
</body>
</html>

```

## HTML <frame> Tag

Not Supported in HTML5.

The <frame> tag was used in HTML 4 to define one particular window (frame) within a <frameset>.

What to Use Instead?

Example

Use the <iframe> tag to embed another document within the current HTML document:

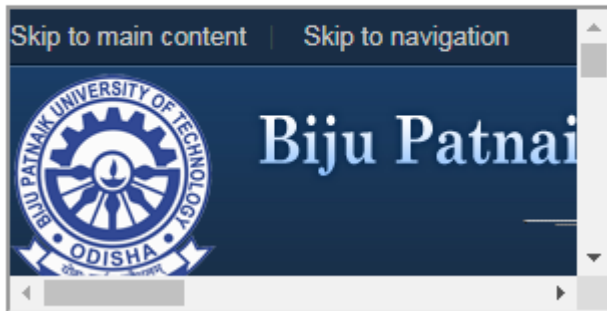
```

<!DOCTYPE html>
<html>
<body>
<h1>The iframe element</h1>
<iframe src="https://www.bput.ac.in" title="Biju
Patnaik University of Technology">
</iframe>
</body>
</html>

```

**O/P**

# The iframe element



## <div> and <span> Tag in HTML

Both the tags (<div> and <span>) are used to represent the part of the web page, <div> tag is used as a block part of the webpage and <span> tag is used as an inline part of the webpage like below:

```
<div>An IT institute <span>CIME</span></div>
```

**HTML <div> tag:** The div tag is known as Division tag. The div tag is used in HTML to make divisions of content on the web page like (text, images, header, footer, navigation bar, etc). Div tag has both opening(<div>) and closing (</div>) tags and it is mandatory to close the tag. As we know Div tag is a block-level tag. In this example, the div tag contains the entire width. It will be displayed div tag each time on a new line, not on the same line.

- **Example:**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Div tag</title>
    <style>
      div {
        color: white;
        background-color: #009900;
        margin: 2px;
        font-size: 25px;
      }
    </style>
```

```

</head>
<body>
  <div> div tag </div>
  <div> div tag </div>
  <div> div tag </div>
  <div> div tag </div>
</body>
</html>

```

### Output:



### TML <span> tag:

The HTML span element is a generic inline container for inline elements and content. It used to group elements for styling purposes (by using the class or id attributes). A better way to use it when no other semantic element is available. The span tag is very similar to the div tag, but div is a block-level tag and span is an inline tag.

### Example

```

<!DOCTYPE html>
<html>
<head>
  <title>span tag</title>
</head>
<body>
  <h2>Welcome To CIME</h2>
  <!-- Inside paragraph applying span tag
  with different style -->
  <p><span style="background-color:lightgreen">
    CIME</span> is An IT institute
    where you can<span style="color:blue;">
    study</span> Post Graduate Programmes <span
    style="background-color:lightblue;"> likes MCA or M.Tech(CSE)</span>

```

and build a professional career in IT field.

```
</p>  
</body>  
</html>
```

### Output:

## Welcome To CIME

CIME is An IT institute where you can study Post Graduate Programmes likes MCA or M.Tech(CSE) and build a professional career in IT field.

### Differences between <div> and <span> tag:

<div>	<span>
The <div> tag is a block level element.	The <span> tag is an inline element.
It is best to attach it to a section of a web page.	It is best to attach a CSS to a small section of a line in a web page.
It accepts align attribute.	It does not accept align attribute.
This tag should be used to wrap a section, for highlighting that section.	This tag should be used to wrap any specific word that you want to highlight in your webpage.