# Web Technology

## Unit-1

### Introduction and Web Development Strategies :-

Before web development strategies first of all we have to know what is worldwide web

### World Wide Web

worldwide web it is a system through which we can access the information

- World Wide Web, which is also known as a Web, is a collection of websites or web pages stored in web servers and connected to local computers through the internet.
- These websites contain text pages, digital images, audios, videos, etc. Users can access the content of these sites from any part of the world over the internet using their devices such as computers, laptops, cell phones, etc.
- The WWW, along with internet, enables the retrieval and display of text and media to your device.

It's made up of three main parts:

- Hypertext Markup Language (HTML): Formats web pages with hypertext links
- Uniform Resource Locator (URL): The addressing system that identifies a document's location
- Hypertext Transfer Protocol (HTTP): Connects computers together

## History of WWW

- WWW is created by Sir Tim Berners Lee in 1989 at CERN in Geneva.

- In 1990, the first text only browsers were setup and CERN scientist could access hypertext files and other information at CERN. HTML was based on a subset of the standard generalized markup language (SGML). To transfer HTML document to remote sites a new protocol was devised called HTTP (Hyper Text Transfer Protocol).
- In the fall of 1991, conference goes around the work started hearing about the promise but sparks still were not flying.

- In 1993, there are only about 50 websites world wide. A browser that allowed user to take advantage of the web's graphical capabilities was developed at the National center for Super Computing application (NCSA). NCSA called the browser Mosaic.

## Protocols Governing Web

Protocol: A protocol is a set of rules that is used to communicate applications to each other.
OR
A protocol is the interface required for communicating the different applications.
Classification:

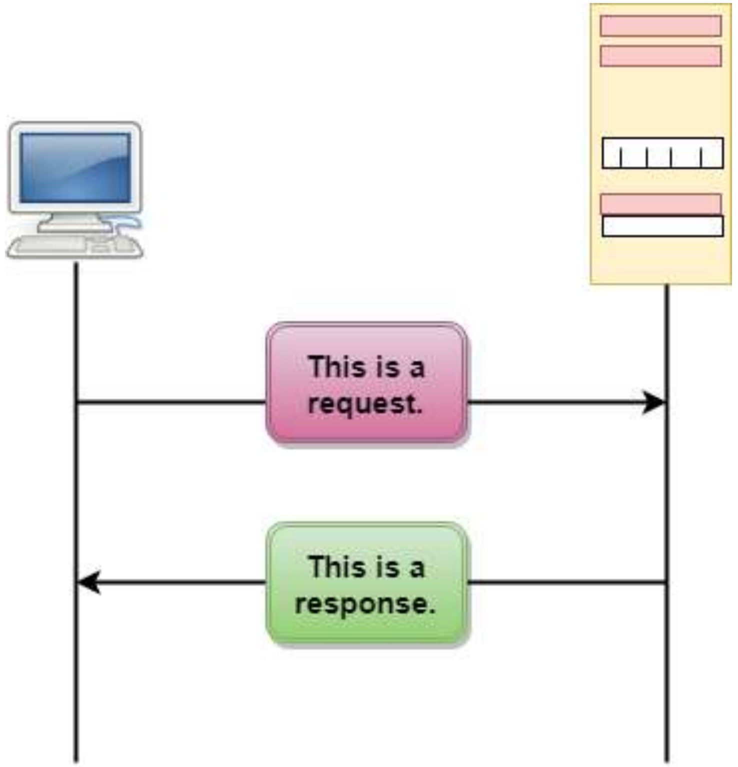
- HTTP
- TCP/IP
- FTP
- E-MAIL
- TELNET

- HTTP stands for **Hypertext Transfer Protocol**.-its run overTcp/IP.
- It is a protocol used to access the data on the World Wide Web (www).
- The HTTP protocol can be used to transfer the data in the form of plain text, hypertext, audio, video, and so on.
- This protocol is known as Hypertext Transfer Protocol because of its efficiency that allows us to use in a hypertext environment where there are rapid jumps from one document to another document.
- HTTP is similar to the FTP as it also transfers the files from one host to another host. But, HTTP is simpler than FTP as HTTP uses only one connection, i.e., no control connection to transfer the files.
- HTTP is used to carry the data in the form of MIME-like format.
- HTTP is similar to SMTP as the data is transferred between client and server. The HTTP differs from the SMTP in the way the messages are sent from the client to the server and from server to the client. SMTP messages are stored and forwarded while HTTP messages are delivered immediately

## Features of HTTP:

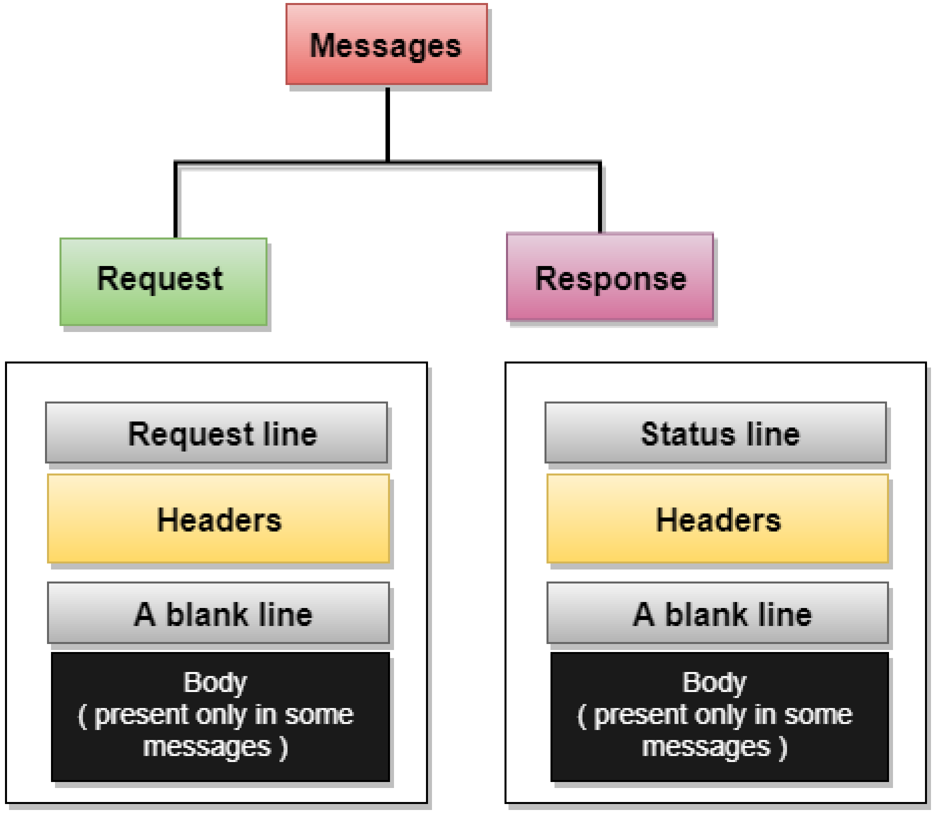
- **Connectionless protocol:** HTTP is a connectionless protocol. HTTP client initiates a request and waits for a response from the server. When the server receives the request, the server processes the request and sends back the response to the HTTP client after which the client disconnects the connection. The connection between client and server exist only during the current request and response time only.
- **Media independent:** HTTP protocol is a media independent as data can be sent as long as both the client and server know how to handle the data content. It is required for both the client and server to specify the content type in MIME-type header.
- **Stateless:** HTTP is a stateless protocol as both the client and server know each other only during the current request. Due to this nature of the protocol, both the client and server do not retain the information between various requests of the web pages.

HTTP Transactions



The above figure shows the HTTP transaction between client and server. The client initiates a transaction by sending a request message to the server. The server replies to the request message by sending a response message

## Messages



HTTP messages are of two types: request and response. Both the message types follow the same message format.

Another form of Http is Https, Which Stands for Http over Secure Socket Layer or HTTP Secure .Https can encrypt a user’s Http request and webpage. This provide more security to user and can prevent common cybersecurity thread, and its widely used on the internet

**Transmission Control Protocol (TCP)-** is a communication standard that enable application program and computing Device to exchange message over a network. It is designed to send packet across the internet and ensure the Successfully delivery of data and message over networks.
Tcp is one of the basic standard defined the rule of the internet and is included within the standard defined by the Internet engineering task force(IEFT).its  one of the most commonly used protocols with in digital network Communications and ensures end to end data delivery

**Transmission Control Protocol (TCP)** is a **connection-oriented protocol for communications** that helps in the exchange of messages between different devices over a network. The Internet Protocol (IP), which establishes the technique for sending data packets between computers, works with TCP

TCP also helps in ensuring that information is transmitted accurately by establishing a virtual connection between the sender and receiver.

## Working of Transmission Control Protocol (TCP)

**Transmission Control Protocol (TCP)** model breaks down the data into small bundles and afterward reassembles the bundles into the original message on the opposite end to make sure that each message reaches its target location intact. Sending the information in little bundles of information makes it simpler to maintain efficiency as opposed to sending everything in one go.
After a particular message is broken down into bundles, these bundles may travel along multiple routes if one route is jammed but the destination remains the same

## Internet Protocol (IP)?

Internet Protocol (IP) is a method that is useful for sending data from one device to another from all over the internet. It is a set of rules governing how data is sent and received over the internet. It is responsible for addressing and routing packets of data so they can travel from the sender to the correct destination across multiple networks. Every device contains a unique IP Address that helps it communicate and exchange data across other devices present on the internet

**For Example:** When a user requests a web page on the internet, somewhere in the world, the server processes that request and sends back an HTML Page to that user. The server makes use of a protocol called the HTTP Protocol. The HTTP then requests the TCP layer to set the required connection and send the HTML file.

Now, the TCP breaks the data into small packets and forwards it toward the Internet Protocol (IP) layer. The packets are then sent to the destination through different routes.

The TCP layer in the user's system waits for the transmission to get finished and acknowledges once all packets have been received

## File Transfer Protocol (FTP)

FTP or File Transfer Protocol is said to be one of the earliest and also the most common forms of transferring files on the internet. Located in the application layer of the OSI model, FTP is a basic system that helps in transferring files between a client and a server. It is what makes the FTP unique that the system provides a reliable and efficient means of transferring files from one system to another even if they have different file structures and operating systems. Contrary to other protocols such as http that cover hypertexts and web resources in general, ftp is dedicated to the management and the transfer of text, binary, or image files.

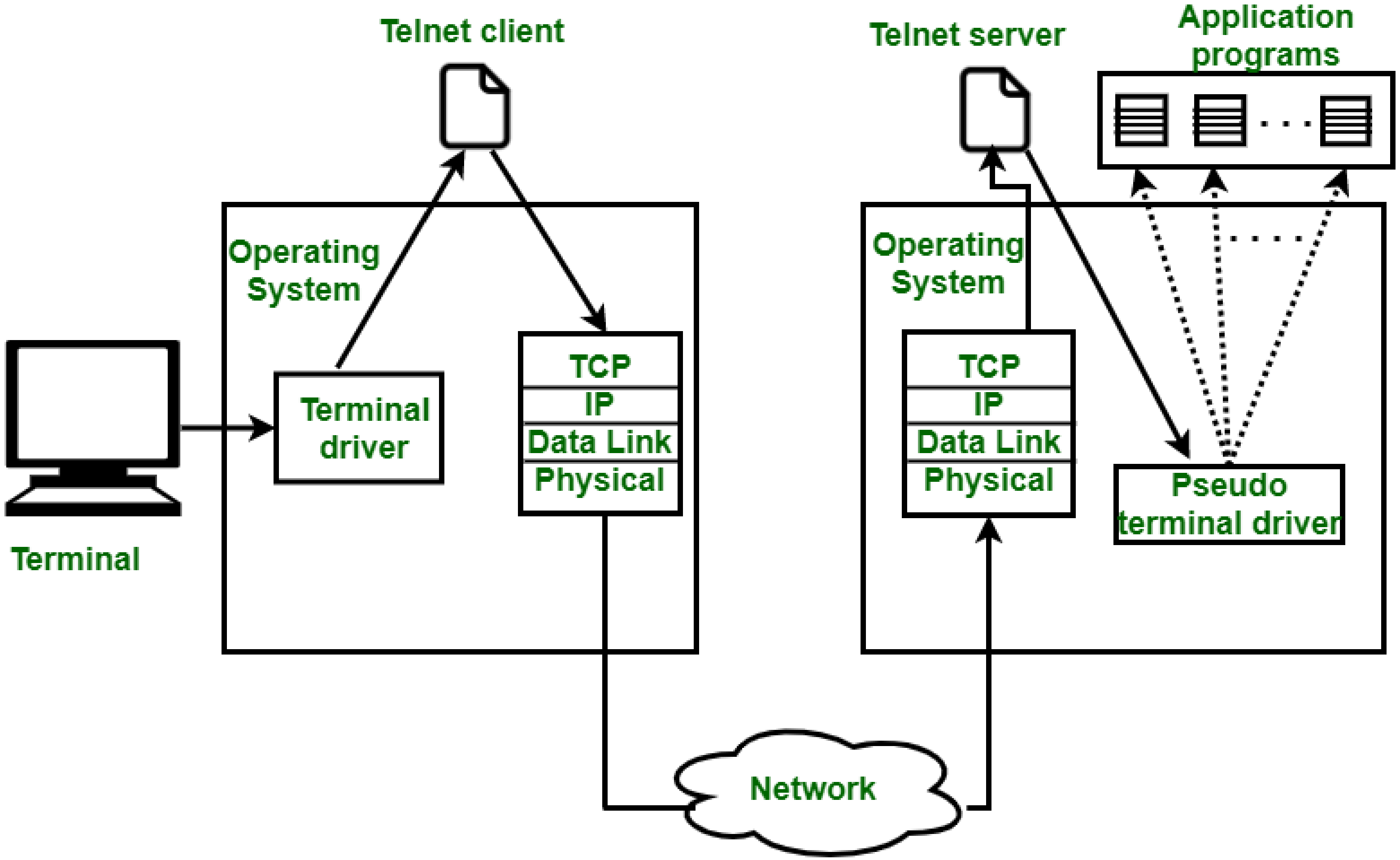
## Types of FTP

**There are different ways through which a server and a client do a file transfer using FTP. Some of them are mentioned below:**

- **Anonymous FTP:** Anonymous FTP is enabled on some sites whose files are available for public access. A user can access these files without having any username or password. Instead, the username is set to anonymous, and the password is to the guest by default. Here, user access is very limited. For example, the user can be allowed to copy the files but not to navigate through directories.
- **Password Protected FTP:** This type of FTP is similar to the previous one, but the change in it is the use of username and password.
- **FTP Secure (FTPS):** It is also called as FTP Secure Sockets Layer (FTP SSL). It is a more secure version of FTP data transfer. Whenever FTP connection is established, Transport Layer Security (TLS) is enabled.
- **FTP over Explicit SSL/TLS (FTPES):** FTPES helps by upgrading FTP Connection from port 21 to an encrypted connection.
- **Secure FTP (SFTP):** SFTP is not a FTP Protocol, but it is a subset of Secure Shell Protocol, as it works on port 22.

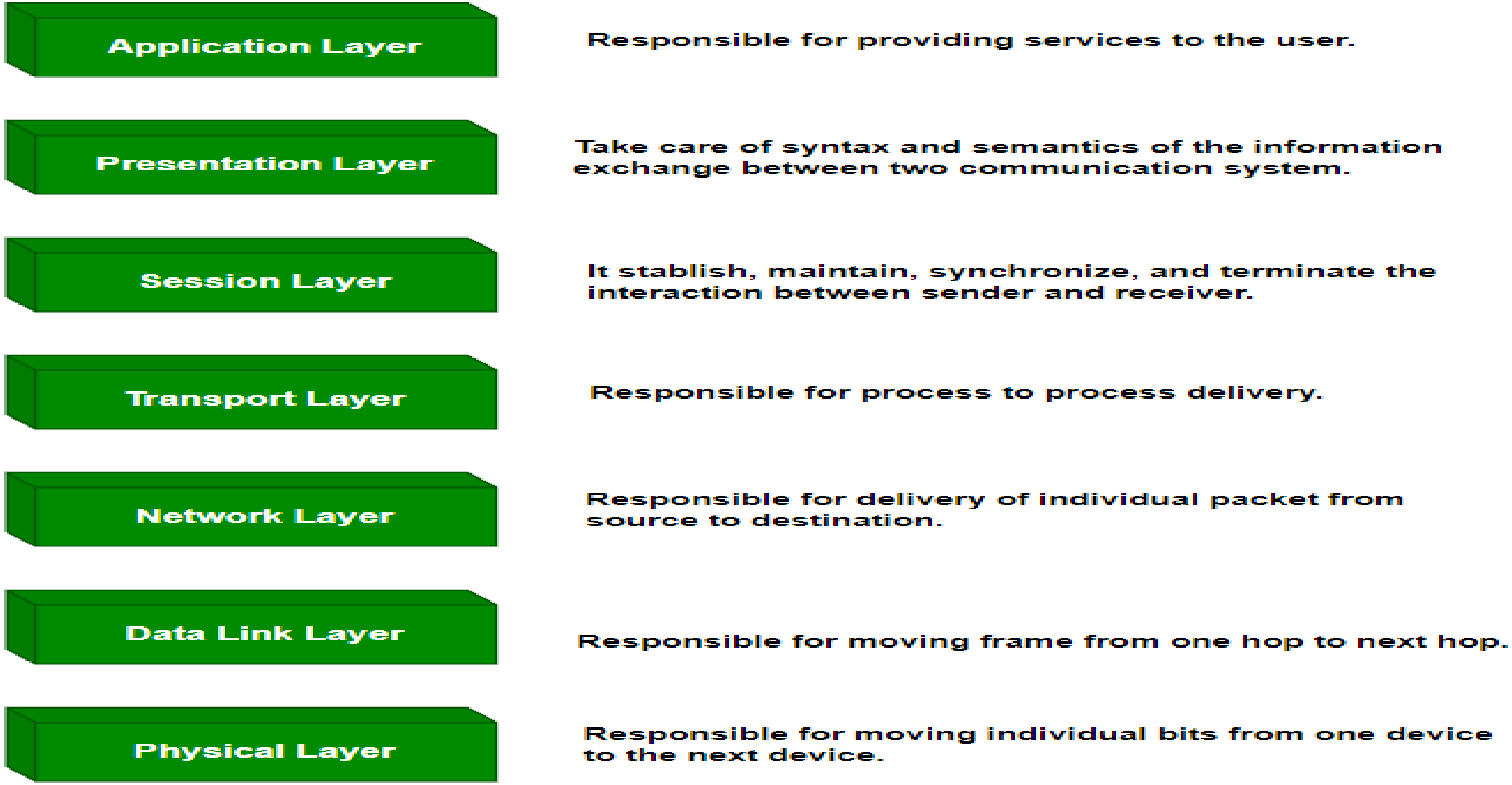
**TELNET** stands for Teletype Network. It is a **client/server application protocol** that provides access to virtual terminals of remote systems on local area networks or the Internet. The local computer uses a telnet client program and the remote computers use a telnet server program.

**TELNET** is a type of protocol that enables one computer to connect to the local computer. It is used as a standard **TCP/IP protocol** for virtual terminal service which is provided by **ISO**. The computer which starts the connection is known as the **local computer**. The computer which is being connected to i.e. which accepts the connection known as the **remote computer**. During telnet operation, whatever is being performed on the remote computer will be displayed by the local computer. Telnet operates on a client/server principle.


Telnet client
Telnet server
Application programs
Operating System
Operating System
Terminal
Terminal driver
TCP
IP
Data Link
Physical
TCP
IP
Data Link
Physical
Pseudo terminal driver
Network

## What is OSI Model?

The OSI model(**Open Systems Interconnection)**, created in 1984 by ISO , is a reference framework that explains the process of transmitting data between computers. It is divided into **seven layers that work together** to carry out specialized **network functions** , allowing for a more systematic approach to networking



| | |
|---|---|
| Application Layer | Responsible for providing services to the user. |
| Presentation Layer | Take care of syntax and semantics of the information exchange between two communication system. |
| Session Layer | It stablish, maintain, synchronize, and terminate the interaction between sender and receiver. |
| Transport Layer | Responsible for process to process delivery. |
| Network Layer | Responsible for delivery of individual packet from source to destination. |
| Data Link Layer | Responsible for moving frame from one hop to next hop. |
| Physical Layer | Responsible for moving individual bits from one device to the next device. |

## Data Flow In OSI Model

When we transfer information from one device to another, it travels through 7 layers of OSI model. First data travels down through 7 layers from the sender’s end and then climbs back 7 layers on the receiver’s end.

Data flows through the OSI model in a step-by-step process:

- **Application Layer:** Applications create the data.
- **Presentation Layer:** Data is formatted and encrypted.
- **Session Layer:** Connections are established and managed.
- **Transport Layer:** Data is broken into segments for reliable delivery.
- **Network Layer** : Segments are packaged into packets and routed.
- **Data Link Layer:** Packets are framed and sent to the next device.
- **Physical Layer:** Frames are converted into bits and transmitted physically.

## website

- A website (also written as web site) is a collection of web pages and related content that is identified by a common domain names and published on at least one web server. Notable examples are wikipedia.org, google.com, and amazon.com.
- All publicly accessible websites collectively constitute the World Wide Web. There are also private websites that can only be accessed on a private network, such as a company's internal website for its employees.
- Websites are typically dedicated to a particular topic or purpose, such as news, education, commerce, entertainment, or social networking. Hyperlinking between web pages guides the navigation of the site, which often starts with a home page.

## Web Application

- A web application (or web app) is application software that runs on a web server, unlike computer-based software programs that are run locally on the operating system (OS) of the device. Web applications are accessed by the user through a web browser with an active network connection.
- These applications are programmed using a client–server modeled structure—the user ("client") is provided services through an off-site server that is hosted by a third-party.
- Examples of commonly-used web applications include:-word processors, online spreadsheets

## Web design

- Web design refers to the design of websites that are displayed on the internet. It usually refers to the user experience aspects of website development rather than software development.
- Web design used to be focused on designing websites for desktop browsers; however, since the mid-2010s, design for mobile and tablet browsers has become ever-increasingly important.
- A web designer works on the appearance, layout, and, in some cases, content of a website. Appearance, for instance, relates to the colors, font, and images used. Layout refers to how information is structured and categorized.
- A good web design is easy to use, aesthetically pleasing, and suits the user group and brand of the website. Many webpages are designed with a focus on simplicity, so that no extraneous information and functionality that might distract or confuse users appears.
- As the keystone of a web designer’s output is a site that wins and fosters the trust of the target audience, removing as many potential points of user frustration as possible is a critical consideration.

## Web hosting

- A Web host is an organization that sells or leases memory space on its servers. Web hosting is typically done in a data center, which provides services to clients that enable them to publish websites on the Internet.
- A Web host can also provide data center space and an Internet connection for servers owned by others. The service provided by a Web host is called Web hosting.

## What is internet-

1- Internet is a global system of interconnected computer networks that use the standard Internet Protocols
suite (TCP/IP) to serve billions of users worldwide
2. Internet is a network of networks that consists of millions of private and public, academic, business, and government networks
of local to global scope that are linked by a broad array of electronic, wireless and optical networking technologies.
3. The internet carries a vast array of information resources and services. to support electronic mail.
4. Most traditional communications media, such as telephone and television services, are reshaped or redefined
using the technologies of the internet, giving rise to services such as Voice over Internet Protocol (VoIP).
5. Newspaper, book and other print publishing has been reshaped into web sites, blogging, and web feeds.
6. The internet has enabled or accelerated the creation of new forms of human interactions through instant messaging,
internet forums, and social networking sites.

## What is Internet Service

1. Internet service provides a way for data to be transferred from Internet servers to our computer.
2. Internet service allows us to access huge amount of information such as text, graphics, sound and software over the internet.

**Four different categories of internet services are as follows :**

**1. Communication services** : There are various communication services available that offer exchange of information with individuals or groups which are as follows :

a. Electronic mail
b. Telnet
c. Mailing lists
d. Internet telephony (VoP)

**2. Information retrieval services** : There exist several information retrieval services offering easy access to information present on the internet which are as follows :

a. File Transfer Protocol (FTP)
b. Gopher

**3. Web Service :**

a. Web service allow exchange of information between application on the web
b. Using web service application can easily interact with each other

**4 world wide web(www) .**

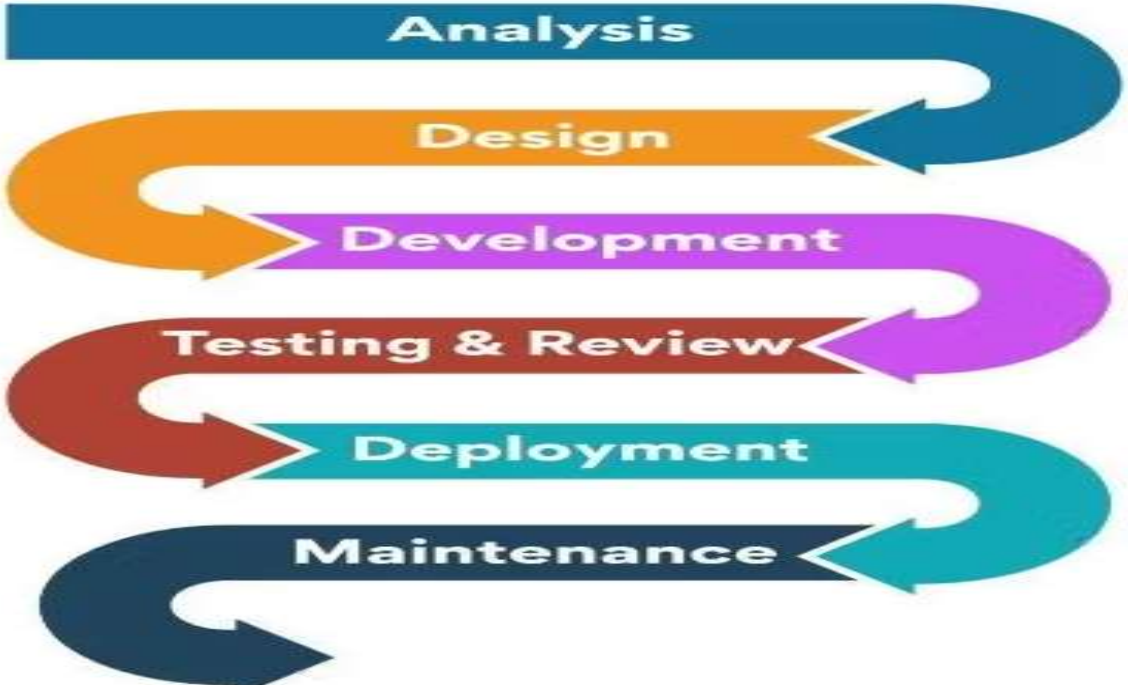
a. www offer a way to access document spread over several server over internet.
b .These document contains texts , graphic ,audio , video ,hyperlink
c. These hyperlink allow the users to navigate between document

## Web Development Strategies :

Web development strategy includes following :

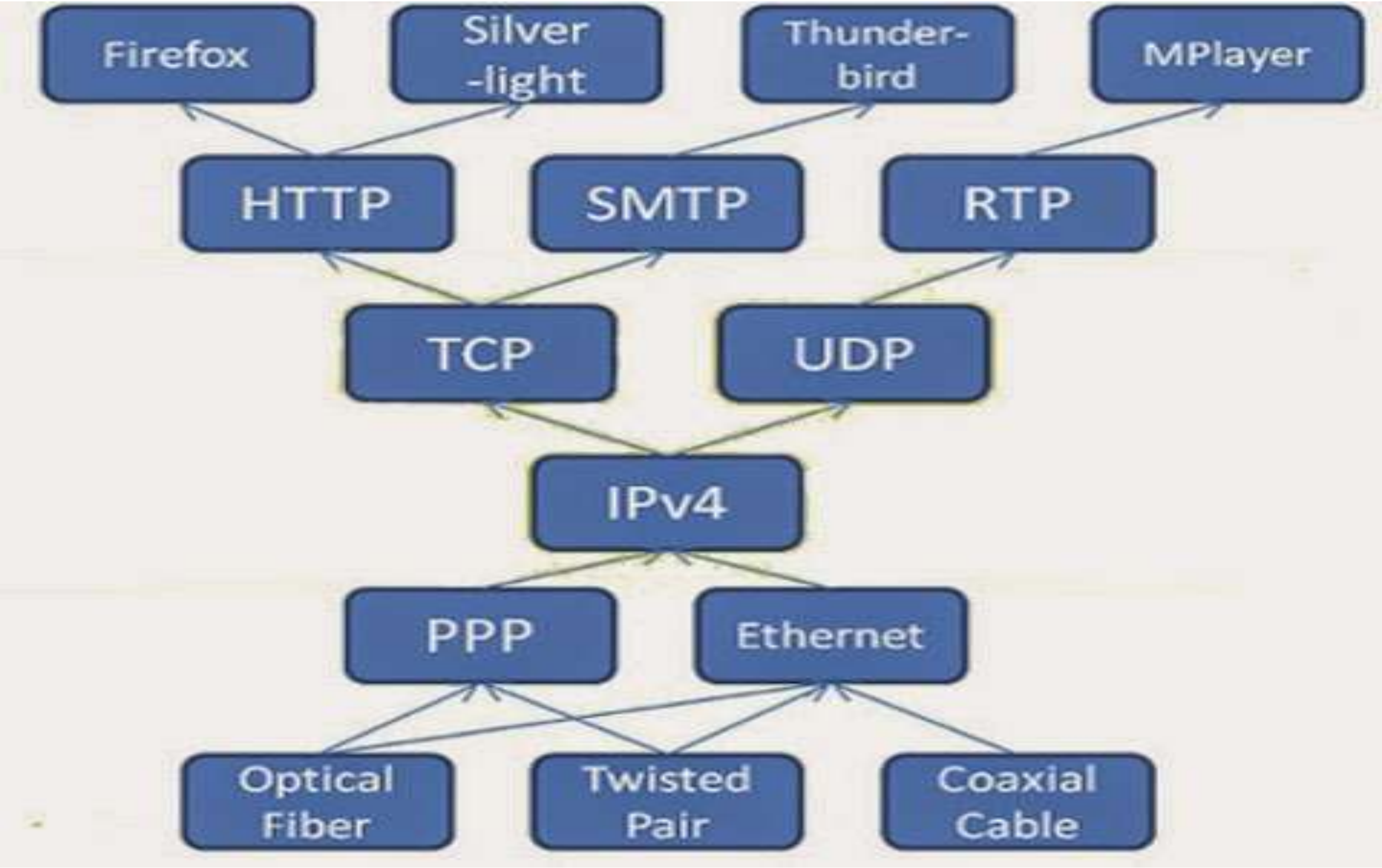
1. **Identify target user** : Identify the user of the website by doing market research.
2. **Make our design portable** : To be successful, website design should be portable and accessible across different browsers, operating systems, and computer platforms.
3. **Design for low bandwidth** : Web pages in website should be accessible at any connection speeds. If page downloaded slowly then users will leave the website before they see the content.
4. **Plan for clear presentation and easy access to information** : Presentation of the information on the website must be clear and easily accessible to the user.
5. **Create smooth transitions** : Plan to create a unified look among the sections and pages of site. Reinforce the identifying elements of the site and create smooth transitions from one page to another.

**Web Development Process-**



## Internet Protocols Governing Web

A protocol is a collection of rules used by computers that are combined, or networked collectively, which define how the computers interact or communicate with one other



1. **HTTP (Hypertext Transfer Protocol)** :

- Hypertext Transfer Protocol (HTTP) is a method used to transfer or convey information on the World Wide Web. HTIP is a request/response protocol between clients and servers.
- The originating client, such as a web browser is referred as the user agent.
- The destination server, which stores or creates resources such as HTML files and images, is called server.

2. **ICMP (Internet Control Message Protocol)** :

- ICMP is primarily used by networked computers operating systems to send error messages.
- The purpose of these control messages is to provide feedback about problems in the communication environment.

**3. RIP (Routing Information Protocol)** :

- RIP is a dynamic routing protocol based on the Bellman-Ford algorithm.
- Routing is the method by which the host or gateway decides where to send the datagram.
- The goal of RIP is to supply the information that is needed to do routing.

**4. OSPF (Open Shortest Path First)** :

- OSPF is classified as an Interior Gateway Protocol (IGP).
- It distributes routing information between routers belonging to a single Autonomous System (AS).
- OSPF also provides the authentication of routing updates and utilizes IP multicast.

5. **TCP/IP**

- TCP/IP stands for Transmission Control Protocol / Internet Protocol
- It is the communication protocol for communication between computers on the internet.
- TCP is connection oriented protocol.
- TCP allows the transmission of arbitrary amount of data by breaking it into stream of separate IP packets.

### 6- UDP

- User Datagram Protocol (UDP) is a connectionless protocol without any error detection facility.
- It is also used for transmission of data .
- This protocol provides a procedure for application programs to send messages to other programs with a minimum of protocol mechanism

## Web Technology

Web Technology refers to the various tools and techniques that are utilized in the process of communication between different types of devices over the Internet. A web browser is used to access web pages. Web browsers can be defined as programs that display text, data, pictures, animation, and video on the Internet. Hyperlinked resources on the World Wide Web can be accessed using software interfaces provided by Web browser

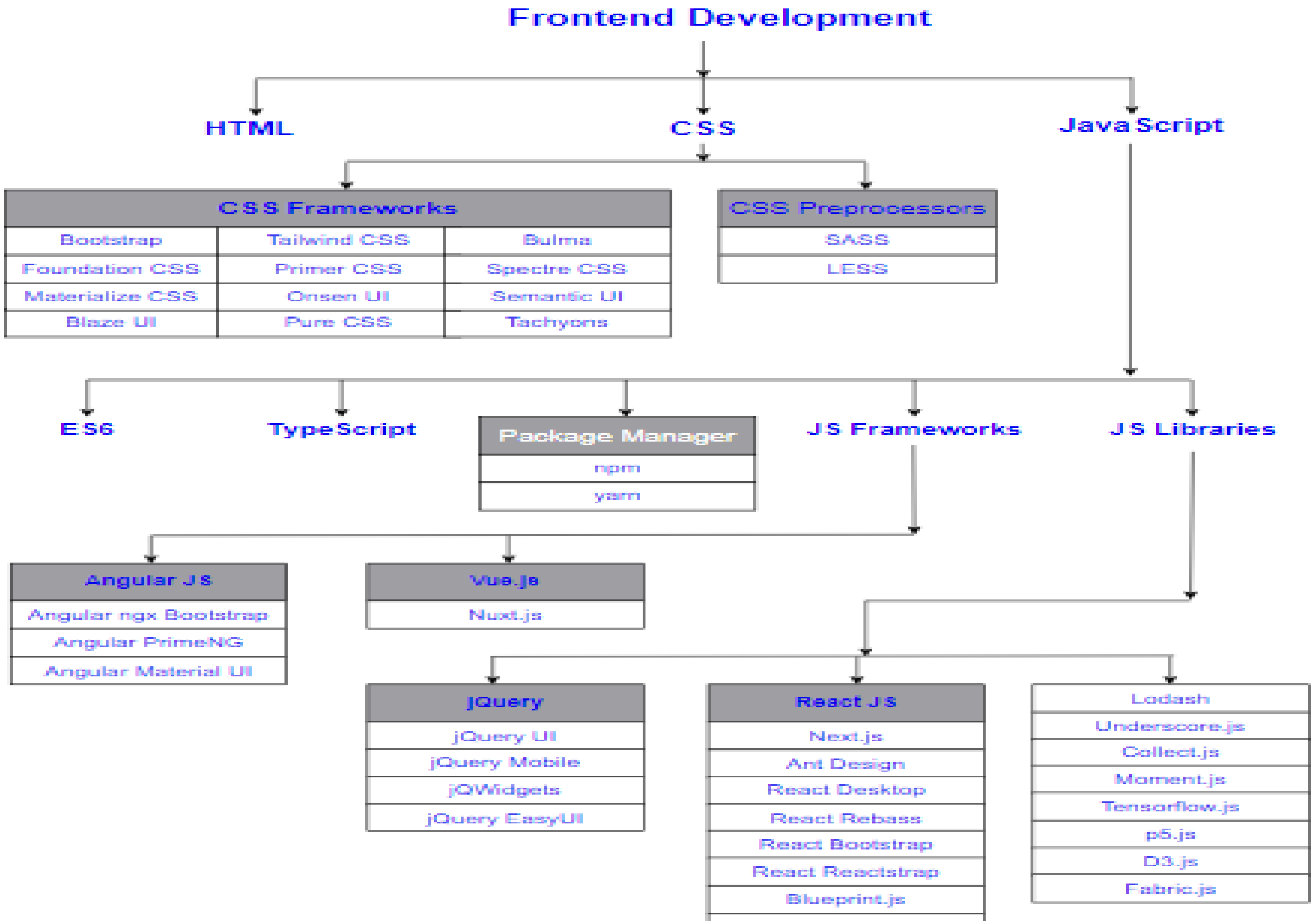
### Web Technology can be Classified into the Following Sections:

- World Wide Web (WWW)
- Web Browser
- Web Server
- Web Pages
- Web Development

### Web Development can be Classified into Two Ways:

Frontend Development
Backend Development

**Frontend Development:** The part of a website that the user interacts directly is termed as front end. It is also referred to as the ‘client side’ of the application



Frontend Development

HTML

CSS

Java Script

| CSS Frameworks | | |
|---|---|---|
| Bootstrap | Tailwind CSS | Bulma |
| Foundation CSS | Primer CSS | Spectre CSS |
| Materialize CSS | Onsen UI | Semantic UI |
| Blaze UI | Pure CSS | Tachyons |

| CSS Preprocessors |
|---|
| SASS |
| LESS |

ES6

TypeScript

| Package Manager |
|---|
| npm |
| yarn |

JS Frameworks

JS Libraries

| Angular JS |
|---|
| Angular ngx Bootstrap |
| Angular PrimeNG |
| Angular Material UI |

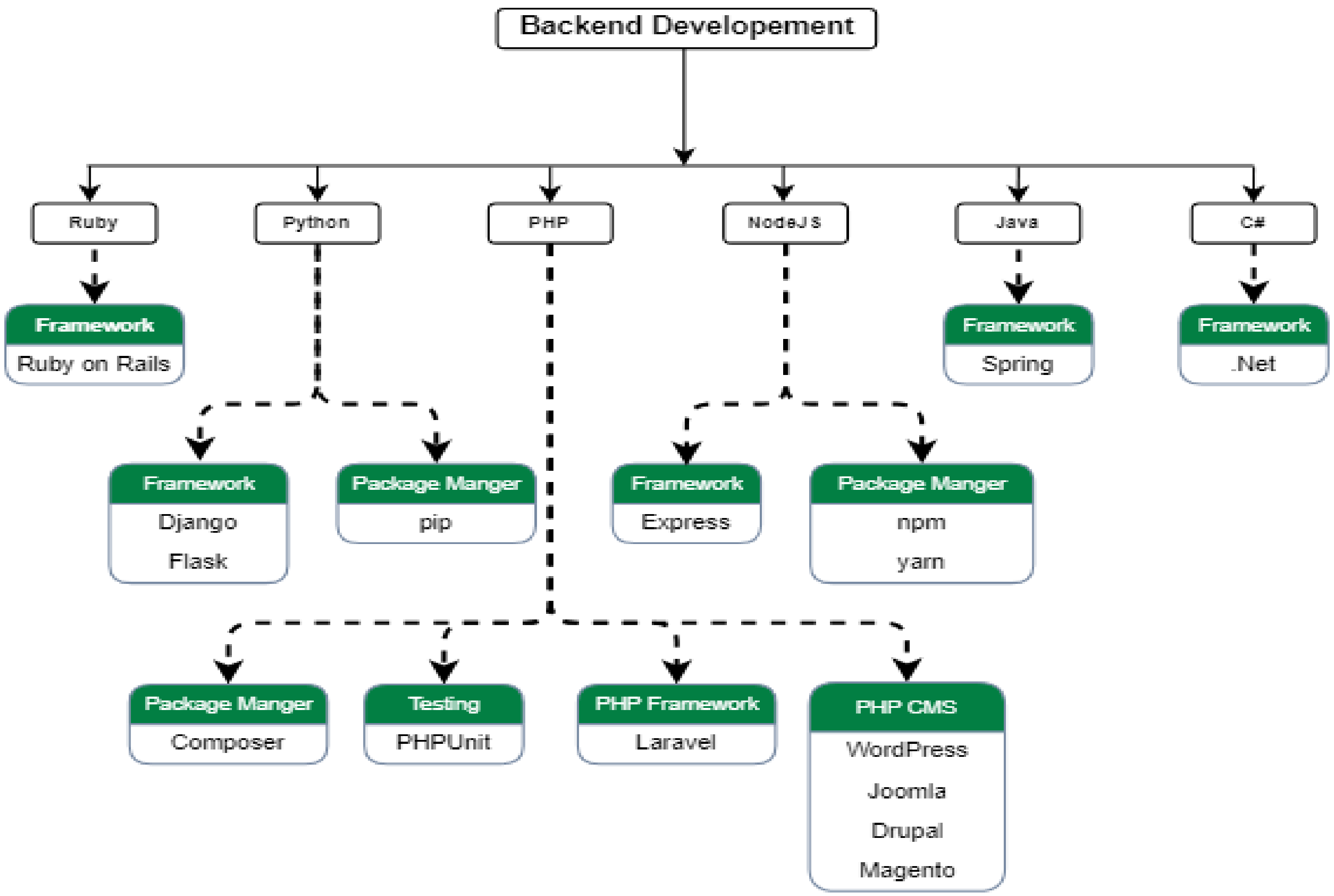
| Vue.js |
|---|
| Nuxt.js |

| jQuery |
|---|
| jQuery UI |
| jQuery Mobile |
| jQWidgets |
| jQuery EasyUI |

| React JS |
|---|
| Next.js |
| Ant Design |
| React Desktop |
| React Rebass |
| React Bootstrap |
| React Reactstrap |
| Blueprint.js |

| |
|---|
| Lodash |
| Underscore.js |
| Collect.js |
| Moment.js |
| Tensorflow.js |
| p5.js |
| D3.js |
| Fabric.js |

**Backend Development:** Backend is the server side of a website. It is the part of the website that users cannot see and interact. It is the portion of software that does not come in direct contact with the users. It is used to store and arrange data.



## Databases

In web technology, a database is a structured collection of data that is stored electronically and accessed via a web application. It serves as the backend component where data is stored, managed, and retrieved. Databases can be relational (like MySQL, PostgreSQL) using structured tables and SQL for queries, or non-relational (like MongoDB, CouchDB) which store data in flexible, document-oriented formats. They enable web applications to handle dynamic content, user data, transactions, and more by providing efficient storage, retrieval, and manipulation capabilities. Database management systems (DBMS) are used to interact with the database, ensuring data integrity, security, and performance

## Relational Database

A relational database stores data in tables, similar to a spreadsheet, where each table has rows and columns. The rows hold individual records, and the columns define the data attributes. Tables can be linked to each other through special keys, allowing related data to be connected.

•**Postgre SQL**: PostgreSQL is a powerful, open-source relational database that supports advanced SQL features and complex queries. It handles structured data, ensures ACID compliance, and is known for its reliability and extensibility.

•**Maria DB**: MariaDB is an open-source relational database that evolved from MySQL, offering improved performance, security, and features. It supports SQL queries, ACID compliance, and is highly compatible with MySQL.

•**MySQL**: MySQL is an open-source relational database management system that uses SQL for managing structured data. It’s known for its reliability, ease of use, and performance, widely used in web applications.

## NoSql Databases

A NoSQL database stores data in a flexible, non-tabular format, unlike traditional relational databases. Instead of using tables with rows and columns, NoSQL databases might use documents, key-value pairs, wide-columns, or graphs to store data. This allows them to handle large amounts of unstructured or semi-structured data efficiently. They are designed to scale easily and manage big data applications.

•**Mongodb**: MongoDB is a NoSQL database storing data in JSON-like documents. It handles unstructured data, supports powerful queries, and scales easily across servers, making it popular for flexible, scalable applications.

•**Cassandra**: Apache Cassandra is an open-source NoSQL database that is used for handling big data. It has the capability to handle structure, semi-structured, and unstructured data.

## Data Format

Format of data is used by web applications to communicate with each other. It is light weight text based data interchange format which means, it is simpler to read and write.

**Below are two common data formats used in web development:**

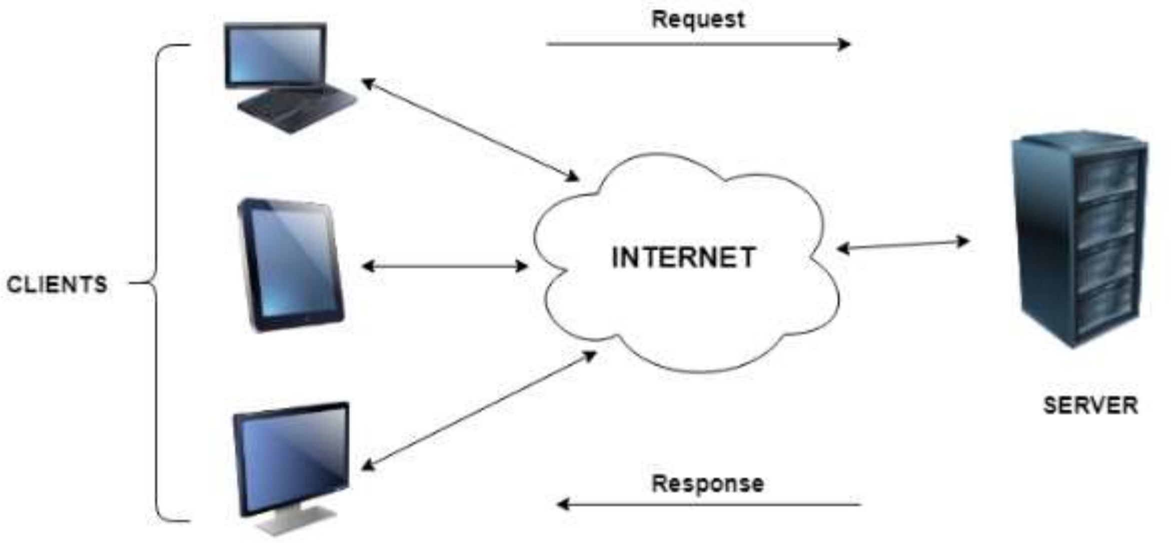
•**XML:** Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

•**JSON:** JSON or JavaScript Object Notation is a format for structuring data.

•**API:** API is an abbreviation for Application Programming Interface which is a collection of communication protocols and subroutines used by various programs to communicate between them

## client-server computing.

In client server computing, the clients requests a resource and the server provides that resource. A server may serve multiple clients at the same time while a client is in contact with only one server. Both the client and server usually communicate via a **computer network** but sometimes they may reside in the same system.



## Characteristics of Client Server Computing

The salient points for client server computing are as follows:

- The client server computing works with a system of request and response. The client sends a request to the server and the server responds with the desired information.
- The client and server should follow a common communication protocol so they can easily interact with each other. All the communication protocols are available at the application layer.
- A server can only accommodate a limited number of client requests at a time. So it uses a system based to priority to respond to the requests.
- Denial of Service attacks hindera servers ability to respond to authentic client requests by inundating it with false requests.
- An example of a client server computing system is a web server. It returns the web pages to the clients that requested them.

## Advantages of Client Server Computing

The different advantages of client server computing are −

- All the required data is concentrated in a single place i.e. the server. So it is easy to protect the data and provide authorization and authentication.
- The server need not be located physically close to the clients. Yet the data can be accessed efficiently.
- It is easy to replace, upgrade or relocate the nodes in the client server model because all the nodes are independent and request data only from the server.
- All the nodes i.e clients and server may not be build on similar platforms yet they can easily facilitate the transfer of data.

## Disadvantages of Client Server Computing

The different disadvantages of client server computing are −

- If all the clients simultaneously request data from the server, it may get overloaded. This may lead to congestion in the network.
- If the server fails for any reason, then none of the requests of the clients can be fulfilled. This leads of failure of the client server network.
- The cost of setting and maintaining a client server model are quite high.

# INTRODUCTION TO HTML

## INTRODUCTION:

HTML stands for Hypertext Markup Language. It is a Language used to createWeb Pages or Hypertext document. A Markup Language is a set of instructions often called TAGS which can be addedto text files. HTML is only a formatting language are not a programming language. The idea behind hypertext is that instead of reading text in a right linear structure we can easily jump from one point to another point.HTML is all about specifying the structure and format of our webpage i.e, it is mainly used for describing the structure document.

HTML is platform independent i.e, for example if we can access internet, we can access WORLD WIDE WEB (WWW) irrespective of client OS and OS of the webserver are accessing. So, we can view one download HTML files on www throughbrowser.

Elements of a web document are labelled through the usage of HTML tags. It is the tags that describe the document. Anything that is not a tag will bedisplayed in the document itself. HTML does not describe any page layout i.e, for example, word for windows havedifferent styles for headings, font size etc. But HTML doesn't have all these. Based on the Platforms, appearance of any element will change. The formatted textwill appear differently on different machines / Platforms. By separating the Structure of the document and appearance, a Program that reads and Understands HTML can make formatting decision based on capabilities of individual Platform. Web Browsers are best examples of HTML formatters.

## Advantages of HTML:-

- A HTML document is small and hence easy to send over the net. It is smallbecause it doesn't include format information.
- HTML documents are cross platform compatible and device independent.We need a HTML readable browser to view them.

## Basic HTML tags:-

**(1) <!doctype> :**

This tag formally starts an HTML document and it also indicates theversion of HTML used.

<!doctype HTML PUBLIC "//w3c//DTDHTML Q.o//EN">

**(2) <HTML>:**

Every HTML document starts with a <html> tag and it is always the firsttag in a html page and indicates that the document is a HTML document.The end tag <html> is </html>.

Example:

<html>........</html>

**(3) <head>:**

It contains the head of an html document, which holds about the document such as title. Each property defined html page should have a head which we create with <head> tag. It has header information and it isdisplayed at the top of the browser. Each tag for <head> is </head>.

<head>.......</head>

**(4) <title>:**

It contains the title of the html document which includes the content thatwill actually appear in the web browser. The entire content of the web page is placed in the pages <body> tag. The end tag <body> is </body>

<title>.......</title>

**(5) <body>:**

It contains the body of the HTML Document , which includes the contentthat will actuall appear in the web browser. The entire content of the webpage will be placed in the pages <body> tag. The end tag of the

<body> tag will be </body>.

<body>.........</body>

## STRUCTURE OF THE HTML PROGRAM:-

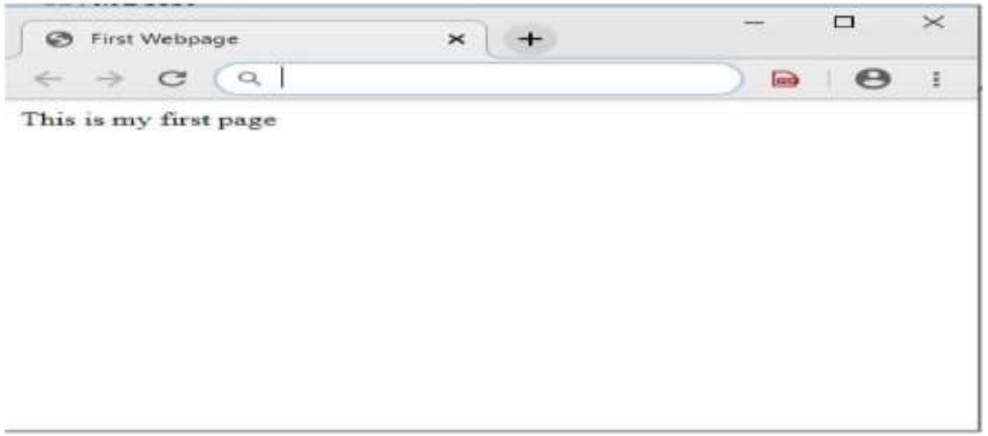
The HTML Program is generally divided into two sections i.e head and body. Weuse <head> and <body> tags to indicate these two sections. <head> section holds the header information of a webpage document indicated by a title that is provided by using <title> tag in the <head>. The title helps us to referto the webpage. <body> section contains the content which we want to display which the webpage.Anything that is not a tag will be displayed within the webpage.

Example:

```
<html>
        <head>
                <title>First Webpage</title>

        </head>
        <body>
                This is my first page
        </body>
</html>
```

**Output:**



This is my first page

**Attribute:**

An Attribute is a Keyword we use in an opening tag to give more information tothe web browser. HTML tags tell the web browsers how to format and organizeour webpages. But we can customize tags using attributes. The Format of an attribute is:

<tagname Attribute=value>

**Attributes of the <body> tag:**

(1) Background:

The URL or a graphic file to be used in the filling the browser'sBackground.

(2) Bgcolor:

The color of the browser's background.

(3) Bgproperties:

It Indicates if the background should scroll when text does. If we set it to"FIXED", the background will not scroll when the text does.

(4) Bottommargin:

Specifies the bottom margin ,the empty space at the bottom of the documentin pixels.

(5) Id:

It is a unique alphanumeric identifier for the tag which we can use to refer toit.

(6) Language:

Scripting language used for the tag.

(7) Leftmargin:

Specifies the left margin, the empty space at the left of the document.

(8) Marginheight:

Gives the height of the margin at the top and bottom of the page in pixels.

(9) MarginWidth:

Gives the width of the left and right margins of the page in pixels.

(10) Rightmargin:

It specifies the right margin, the empty space to the right margin of thedocument in pixels.

(11) Scroll:

It specifies whether a vertical scrollbar appears to the right of the documentcan be yes (or) no.

(12) Style:

Inline style indicating how to render the element.

(13) Text:

Color of the in the document.

(14) Topmargin:

It specifies the top margin the space at the top of the document in pixels.

(15) Link:

It specifies the color of hyperlinks that have not yet been visited.

(16) Alink:

It specifies the color of hyperlinks as they are being clicked.

(17) Vlink:

It specifies the color of hyperlinks as they have been visited.

(18)<!-- --> Comment tag:

Annotates a web page with a comment. In the HTML that we can by lookingat the HTML but it will not be displayed in the web browser.

<! ------ This is a comment--------- >

**Formatting with HTML tags:**

To set the actual style of text as displayed in a web page we can text style tags. There are a number of ways to apply styles to text.

(1) <b>:

It creates a bold text i.e, sets the text style to bold.

**Attributes:**

a. Id:

It is a unique alphanumeric identifier for the tag which we can use to refer toit.

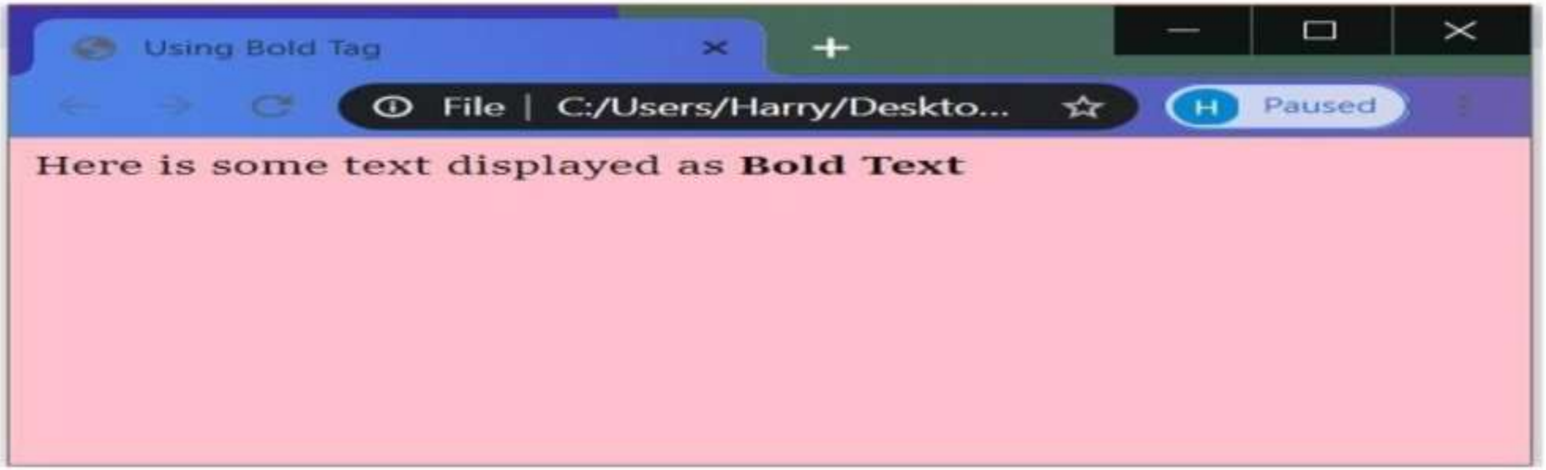
b. Style:

The Inline style indicating how to render the element.

**Example:**

```
<html>
        <head>
                <title>Using Bold Tag </title>
        </head>
        <body bgcolor=”pink”>
                Here is some text displayed as <b> Bold Text </b>
        </body>
</html>
```

**Output:**



Here is some text displayed as **Bold Text**

(2)<I>:

It displays text in Italics.(3)

<U>:

It displays text in Underlined text.(4)

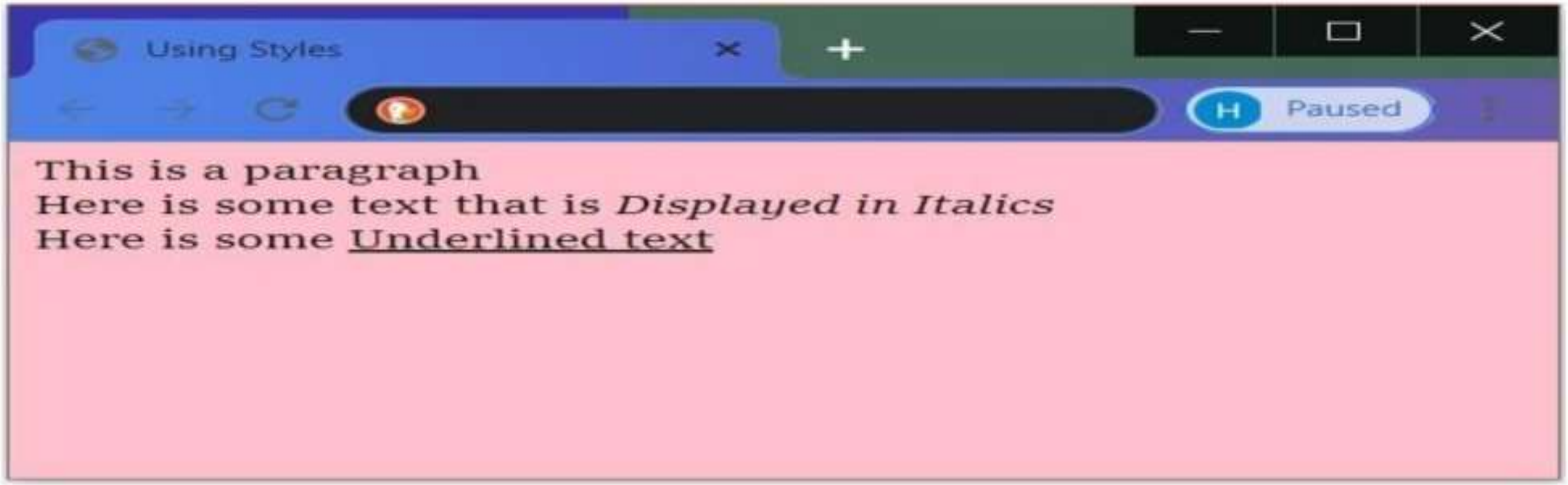
<P>:

It displays the Paragraph text.

**Example:**

```
<html>
        <head>
                <title> Using Styles </title>
        </head>
        <body bgcolor=”pink”>
                <p> This is a paragraph <br>
                Here is some text that is <i> Displayed in Italics </i>
                <br>Here is some <u> Underlined text </u>
        </body>
</html>
```

**Output:**



This is a paragraph
Here is some text that is *Displayed in Italics*
Here is some Underlined text

(5) <s> and <strike>:

It Displays text in smile through style. The <s> and <style> tags are used forthe same effect. HTML 2 used <strike> , HTML 3 called it <s>, HTML 3.2 caused it <strike> again.

(6) <big>:

Renders text in a bigger font than the current default.

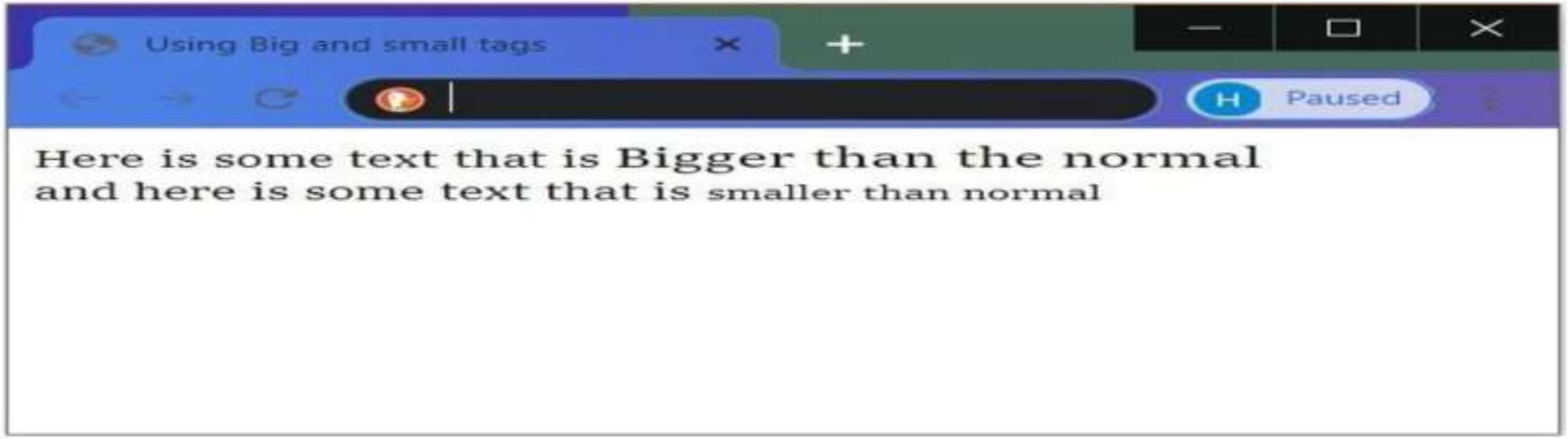
(7) <small>:

Renders text in a smaller font than the current default.

**Example:**

```
<html>
        <head>
                <title> Using Big and small tags </title>
        </head>
        <body>
                Here is some text that is <big> Bigger than the normal </big> <br>and here is
                some text that is <small> smaller than normal </small>
        </body>
</html>
```

**Output:**



Here is some text that is Bigger than the normal
and here is some text that is smaller than normal

(8) <sub>: It Styles the text as a subscript.

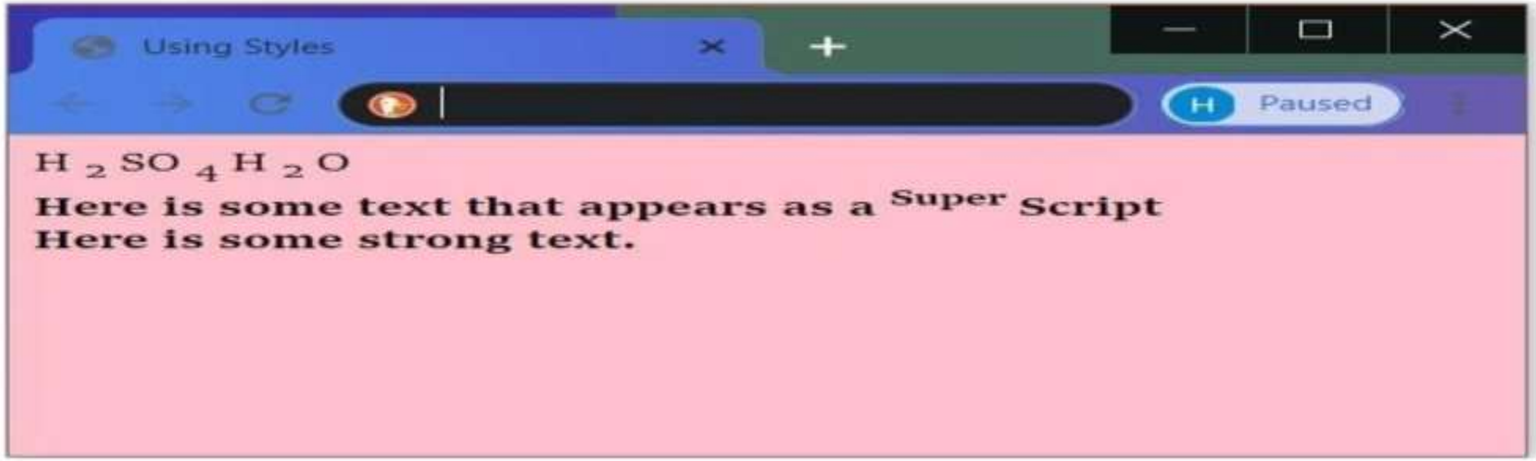
(9) <sup>: It Styles the text as a superscript.

(10) <strong>: Emphasizes text strongly, usually rendered in bold.

**Example:**

```
<html>
        <head>
                <title>Using Styles </title>
        </head>
        <body bgcolor=”pink”>
                H <sub> 2 </sub> SO <sub> 4 </sub>H
                <sub> 2 </sub> O
                <br>
        <b>Here is some text that appears as a <sup> Super </sup> Script <br>Here is some
<strong> strong </strong> text.
        </body>
</html>
```

**Output:**



$H_2SO_4H_2O$

**Here is some text that appears as a $^{\text{Super}}$ Script**
**Here is some strong text.**

(11)Headings:

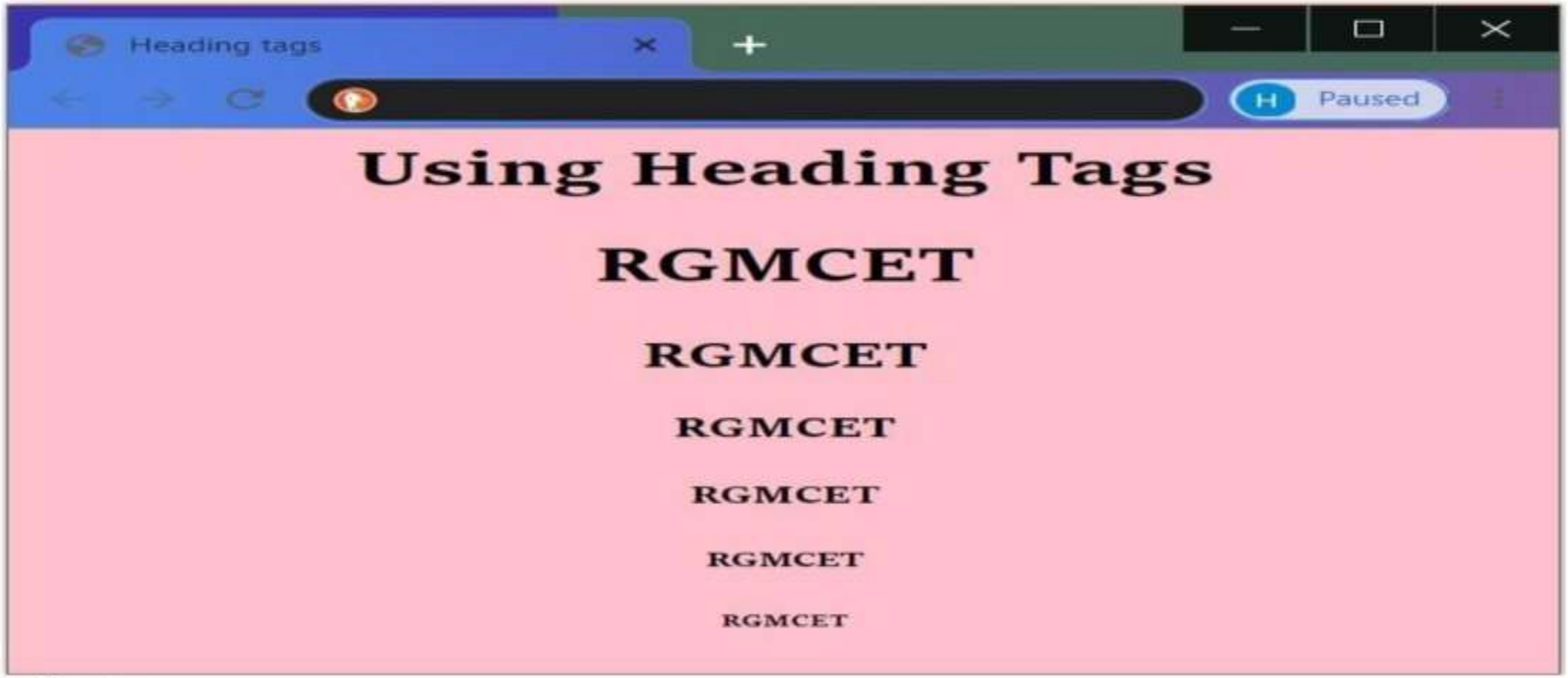
<h1>,<h2>,<h3>,<h4>,<h5> & <h6>

The heading element tags are <h1>,<h2>,<h3>,<h4>,<h5><h6>. These elements create the headings in our web pages by displaying bold text in avariety of sizes <h1> being larger <h6> being smaller.

**Example:**

```
<html>
        <head>
                <title> Heading tags </title>
        </head>
        <body bgcolor=”pink”>
                <center>
                <h1> Using Heading Tags</h1><br>
                <h1> RGMCET </h1><br>
                <h2> RGMCET </h2><br>
                <h3> RGMCET </h3><br>
                <h4> RGMCET </h4><br>
                <h5> RGMCET </h5><br>
                <h6> RGMCET </h6><br>
                </center>
        </body>
</html>
```

**Output**



Using Heading Tags

RGMCET

RGMCET

RGMCET

RGMCET

RGMCET

RGMCET

(12)<font>:

This tag will give us an option to select text size, color and face.

**Attributes:**

a. color: Color of the text.

b. Size: Size of the text in points

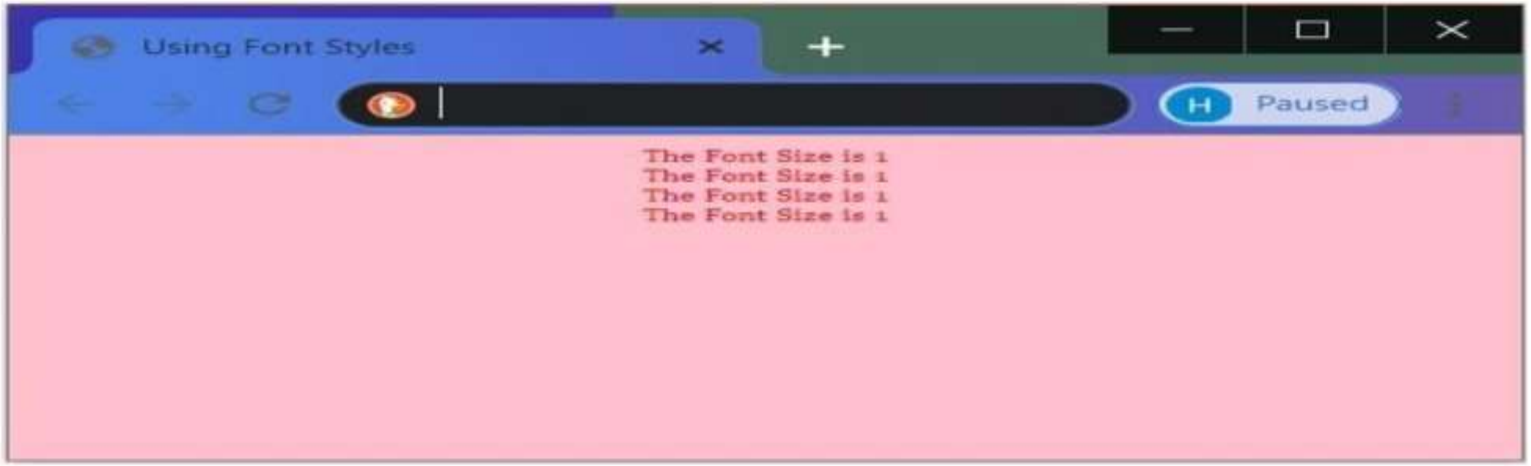
c. Face: The font face can be a list of names separated by commas.

d. Id: Unique alphanumeric identifier for a tag, which we can use to refer to it.

**Example:**

```
<html>
        <head>
                <title> Using Font Styles </title>
        </head>
        <body bgcolor=”pink”>
                <center>
                <font size=”1” color=”red”> The Font Size is 1 </font> <br>
                <font size=”10” color=”yellow”> The Font Size is 10 </font> <br>
                <font size=”20” color=”orange”> The Font Size is 20</font> <br>
                <font size=”30” color=”aqua”> The Font Size is 30 </font> <br>
                </center>
        </body>
</html>
```

**Output**



```
The Font Size is 1
The Font Size is 1
The Font Size is 1
The Font Size is 1
```

(13)<marquee> tag:

Displays scrolling text in a marquee style.

**Attributes:**

a. Align:

Sets the alignment of the text relative to marquee.Set to:

Top(default), middle (or) bottom.

b. behavior:

Sets how the text in the marquee should move can be scroll (default), slide(text enters from one side and stops at the other side), alternate (text seemsto bounce from one side to the other).

c. bgcolor:

It sets the background color for the marquee box.

d. Direction:

Sets the direction the text should scroll can be left, right, down or up.

e. Height:

It specifies the height of the marquee.

f. Loop:

Sets how many times we want the marquee to cycle. Is set to positive integeror -1 for continuous cycling.

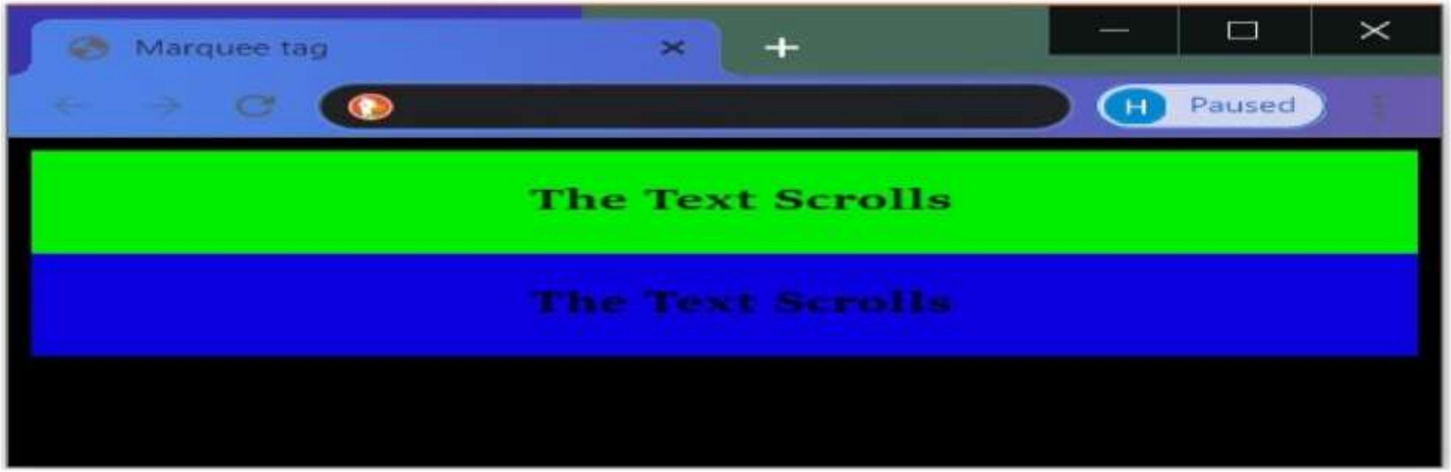
g. Scrolldelay:

Sets the number of the milliseconds between each successive display text.

**Example:**

```
<html>
        <head>
                <title> Marquee tag </title>
        </head>
        <body bgcolor=”pink”>
                <marquee   align=”top”   loop=”infinite”   behavior=”scroll”   bgcolor=”red”
direction=”right”> <h3> The Text Scrolls </h3></marquee>
<marquee   align=”middle”   loop=”infinite”   behavior=”slide”   bgcolor=”blue”
direction=”left”> <h3> The Text Scrolls </h3></marquee>
        </body>
</html>
```

**Output**



The Text Scrolls

The Text Scrolls

(14) <pre> tag(preformatted text):

<pre> marks the text as preformatted text i.e, all the spaces and carriage returns as rendered exactly as you type them.

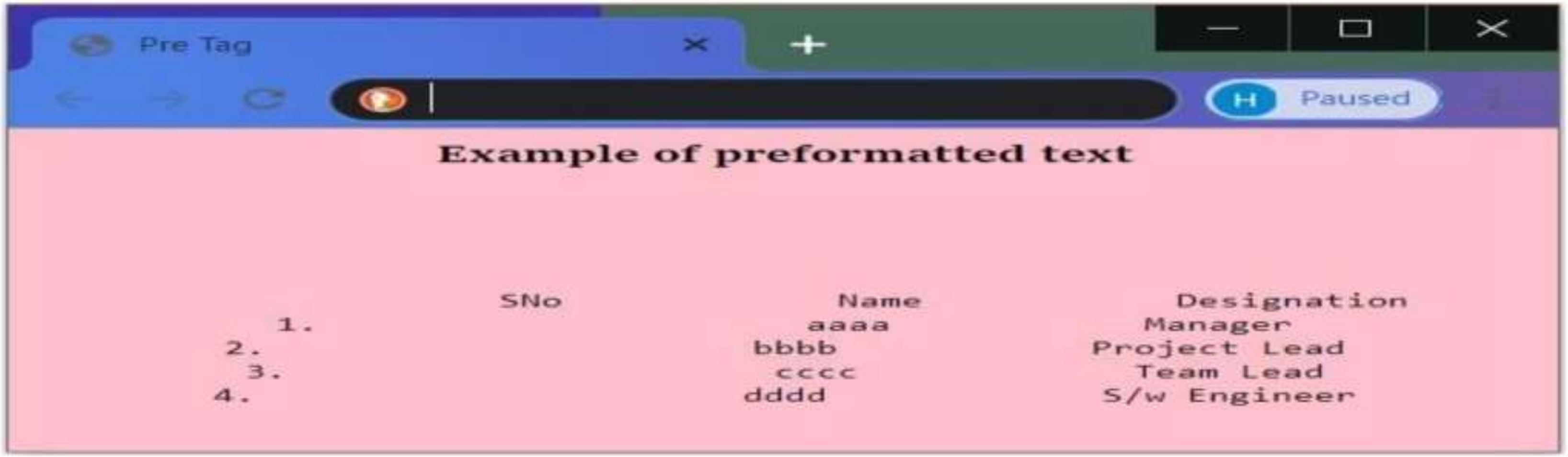
**Example:**

```
<html>
          <head>
                    <title> Pre Tag </title>
          </head>
          <body bgcolor=”pink”>
                    <center>
                    <h4> Example of preformatted text </h4> <br> <br>
                    <pre>

                    SNo           Name                 Designation
                    1.            aaaa                  Manager
                    2.            bbbb                  Project Lead
                    3.            cccc                  Team Lead
                    4.            dddd                   S/w Engineer
                    </pre>
```

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

**Output:**



**Example of preformatted text**

```
                         SNo                Name                Designation
            1.                            aaaa                Manager
        2.                              bbbb                Project Lead
          3.                             cccc                Team Lead
        4.                             dddd                S/w Engineer
```

**Lists:**

Lists lets us display information in a compact, right format. There are three kindsof lists:

1. Unordered List
2. Ordered List
3. Definition List

**Unordered List:**

An Unordered list is a list of items that are marked with burden. The Unordered list is created by using <ul>tag are the list items in the list are createdby </ul> tag and the list items in the list are created by <li> tag.

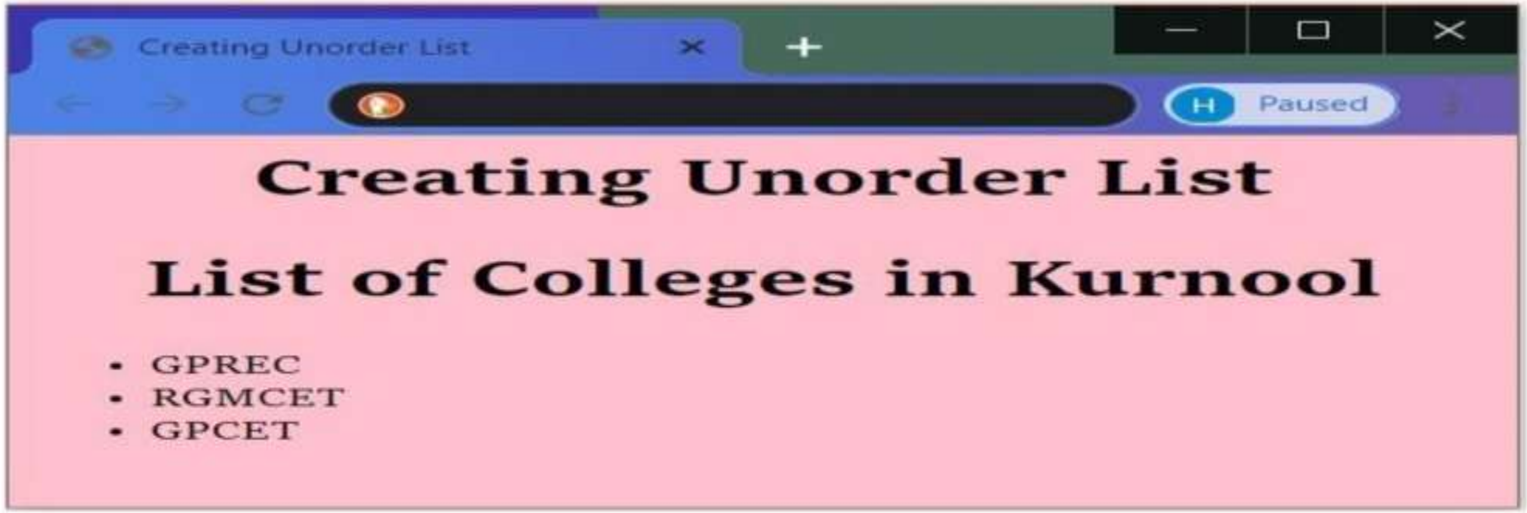
```
<ul>
   <li>List Item 1 </li>
  <li>List Item 2 </li>
</ul>
```

**Example:**

```
<html>
```

```
<head>
      <title> Creating Unorder List </title>
</head>
<body bgcolor=”pink”>
  <h1 align=”center”> Creating Unorder List</h1>
  <h1 align=”center”>List of Colleges in Kurnool</h1>
<ul>
   <li>GPREC</li>
   <li>RGMCET</li>
   <li>GPCET</li>
 </ul>
</body>
</html>
```

**Output**



Creating Unorder List

List of Colleges in Kurnool

- GPREC
- RGMCET
- GPCET

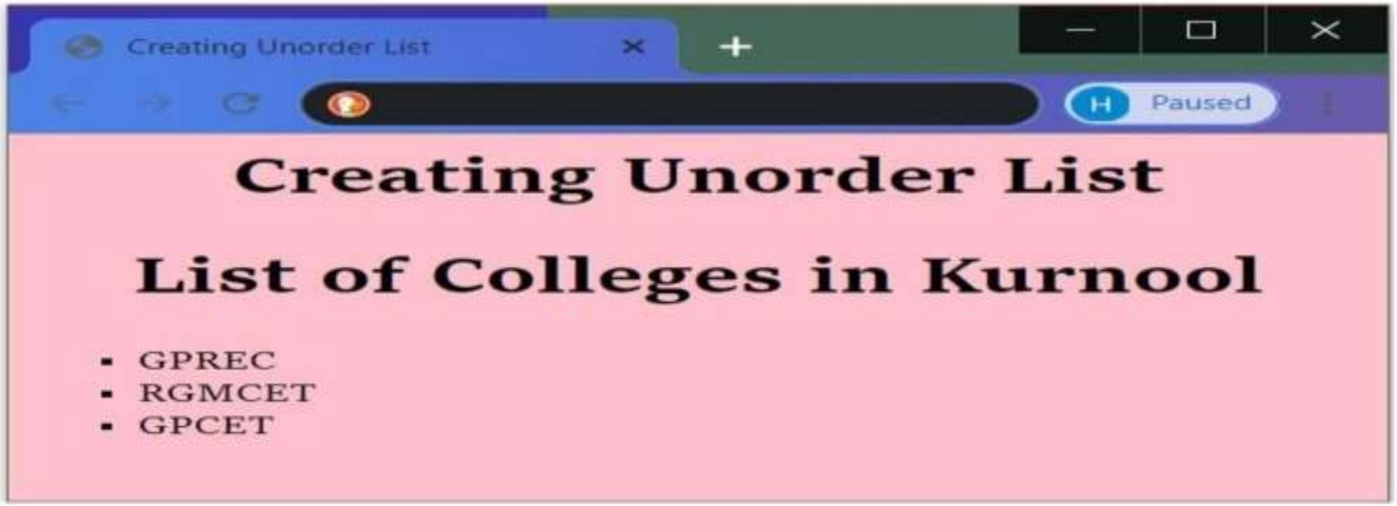
**Creating Customized Unordered Lists:**

We customized unordered lists by setting the “Type” attribute to three different values. DISC (default), SQUARE and CIRCLE which sets the type of bullet thatappears before the list item.

**Example:**

```
<html>
<head>
      <title> Creating Unorder List </title>
</head>
<body bgcolor=”pink”>
  <h1 align=”center”> Creating Unorder List</h1>
  <h1 align=”center”>List of Colleges in Kurnool</h1>
<ul type=”square”>
   <li>GPREC</li>
   <li>RGMCET</li>
   <li>GPCET</li>
 </ul>
</body>
</html>
```

**Output**



Creating Unorder List

List of Colleges in Kurnool

- GPREC
- RGMCET
- GPCET

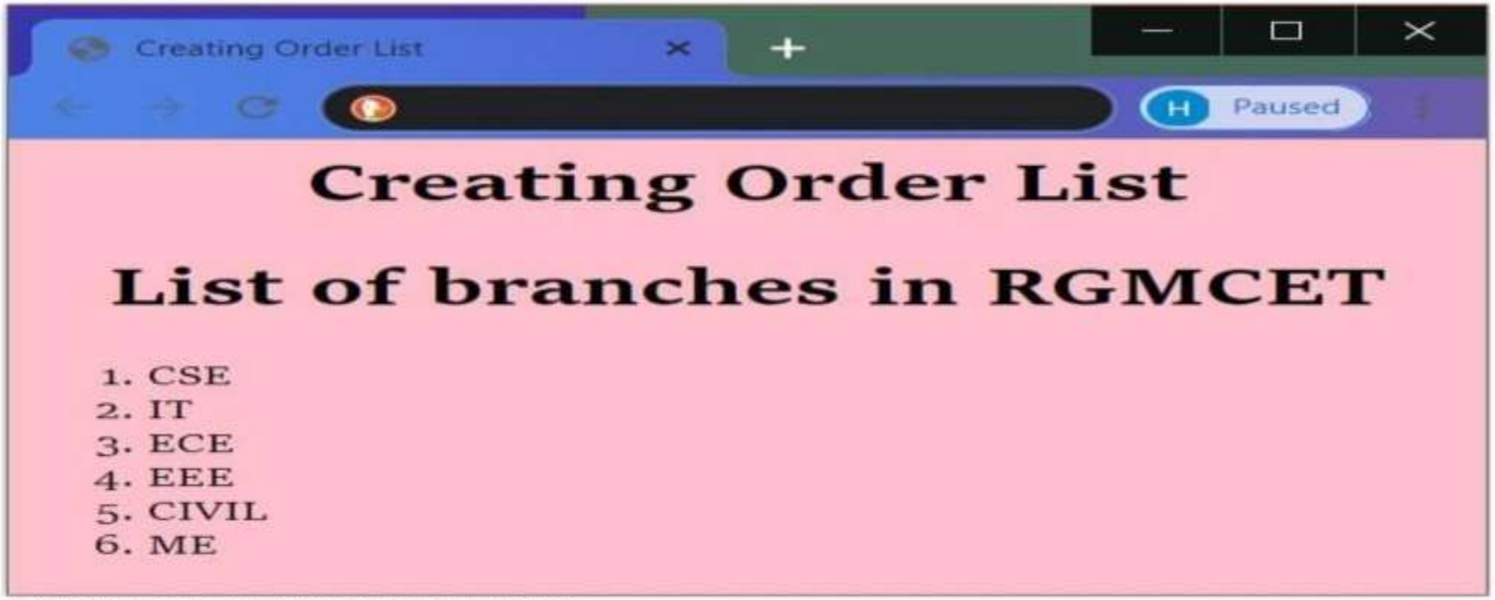
**Ordered List:**

While the unordered lists display simple bullet before each list item. Ordered lists use a number system / lettering scheme to indicate that the items are ordered in some ways, ordered lists are created by <ol> tag and the list items are created using <li> tag.

**Example:**

```
<html>
<head>
    <title> Creating Order List </title>
</head>
<body bgcolor=”pink”>
 <h1 align=”center”> Creating Order List</h1>
 <h1 align=”center”>List of branches in RGMCET</h1>
<ol>
   <li>CSE</li>
   <li>IT</li>
   <li>ECE</li>
   <li>EEE</li>
   <li>CIVIL</li>
   <li>ME</li>
 </ol>
</body>
</html>
```

**Output**



Creating Order List

List of branches in RGMCET

1. CSE
2. IT
3. ECE
4. EEE
5. CIVIL
6. ME

**Creating Customized Ordered Lists:-**

We can customize the numbering system used in ordered lists by using the TYPEattribute, which we can set to these values:

1. Default numbering system (1, 2, 3, ....)

A. Uppercase Letters (A, B, C, .....)

a. Lowercase Letters (a, b, c, ...)

I. Large Roman Numerals (I, II , III, ....)

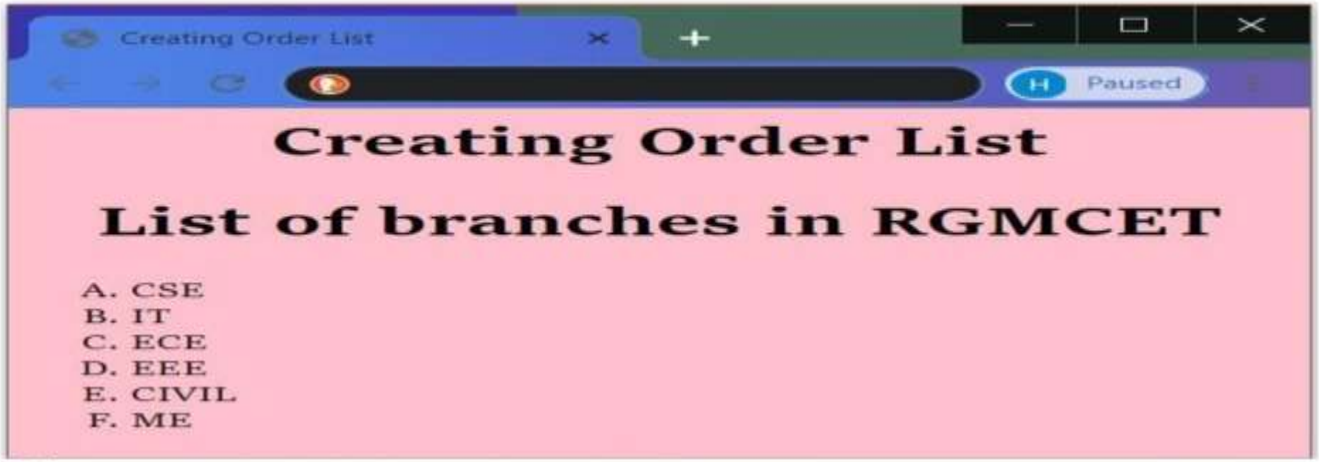
i. Small Roman Numerals (i, ii, iii, .....)

**Example:**

```
<html>
<head>
 <title> Creating Order List </title>
 </head>
 <body bgcolor=”pink”>
   <h1 align=”center”> Creating Order List</h1>
      <h1 align=”center”>List of branches in RGMCET</h1>
```

```
            <ol type=”A”>
                  <li>CSE</li>
                  <li>IT</li>
                  <li>ECE</li>
                  <li>EEE</li>
                  <li>CIVIL</li>
                  <li>ME</li>
         </ol>
 </body>
</html>
```

**Output**



Creating Order List

List of branches in RGMCET

```
A. CSE
B. IT
C. ECE
D. EEE
E. CIVIL
F. ME
```

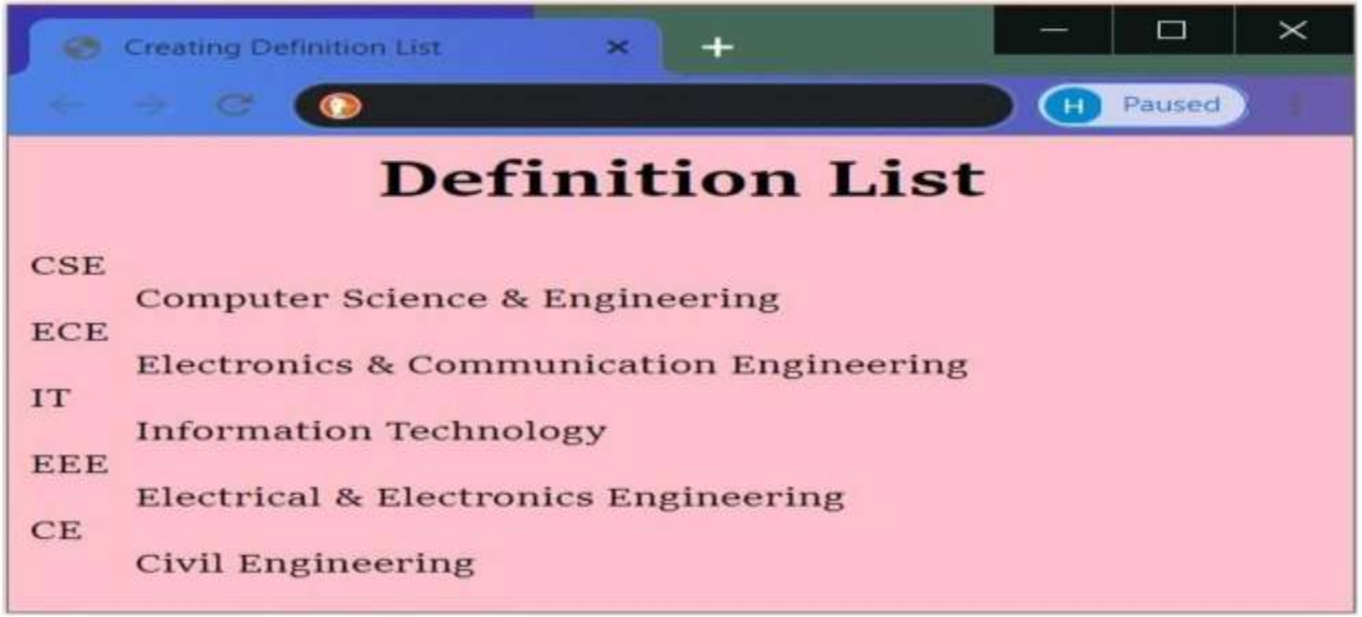
**Definition List:-**

These lists include both definition terms as well as their definition. To create the definition lists we use <dl> tag. For creating definition terms we use <dt> tag andfor data definitions we use <dd> tag.

**Example:**

```
<html>
    <head>
         <title>Creating Definition List</title>
    </head>
    <body bgcolo=”pink”>
        <h1 align=”center”>Definition List</h1>
        <dl>
                <dt>CSE<dd>Computer Science & Engineering
                <dt>ECE<dd>Electronics & Communication Engineering
                <dt>IT<dd>Information Technology
                <dt>EEE<dd>Electrical & Electronics Engineering
                <dt>CE<dd>Civil Engineering
       </dl>
    </body>
</html>
```

**Output**



```
                    Definition List

CSE
        Computer Science & Engineering
ECE
        Electronics & Communication Engineering
IT
        Information Technology
EEE
        Electrical & Electronics Engineering
CE
        Civil Engineering
```

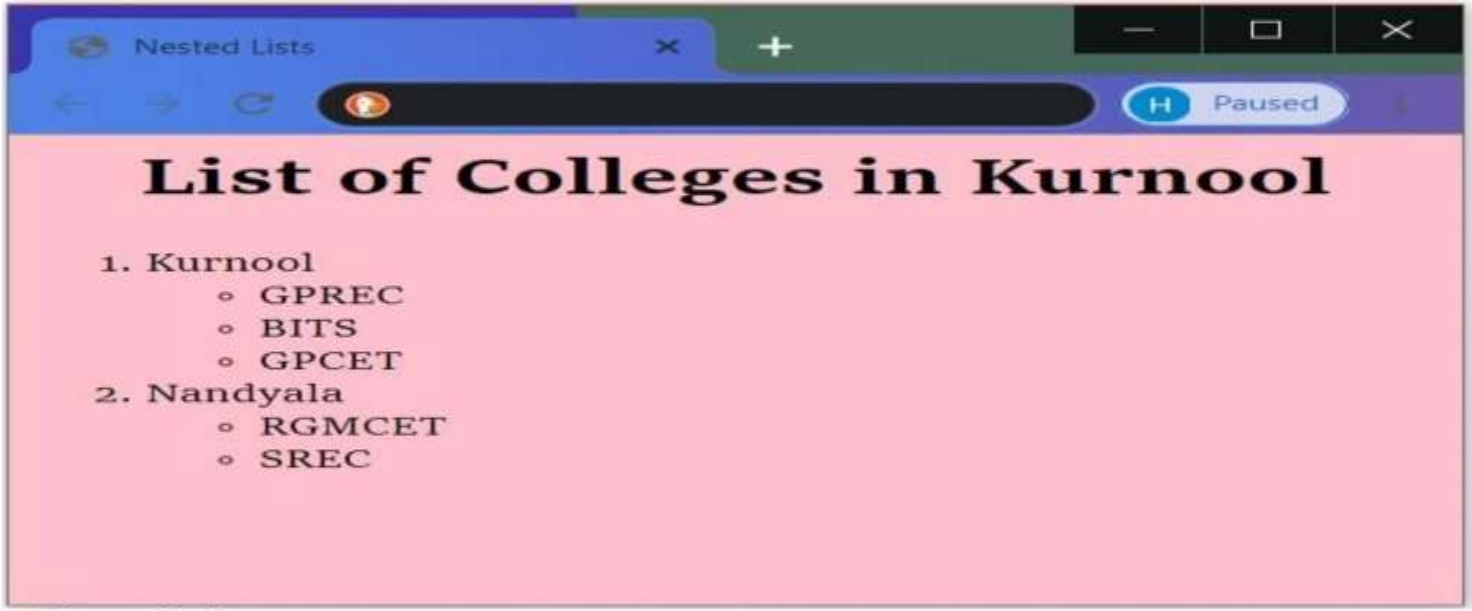
**Nesting Lists:-**

We have the capability of nesting lists inside other lists.

**Example:**

```
<html>
    <head>
         <title>Nested Lists</title>
    </head>
    <body bgcolor=”pink”>
        <h1 align=”center”>List of Colleges in Kurnool</h1>
 <ol>
     <li>Kurnool</li>
       <ul>
           <li>GPREC</li>
           <li>BITS</li>
             <li>GPCET</li>
      </ul>
   <li>Nandyala</li>
      <ul>
           <li>RGMCET</li>
           <li>SREC</li>
      </ul>
 </ol>
</body>
</html>
```

**Output**



# List of Colleges in Kurnool

1. Kurnool
   - GPREC
   - BITS
   - GPCET
2. Nandyala
   - RGMCET
   - SREC

**Creating Hyperlinks:**

What makes the web so effective is the ability to define links from one page to another. In web terms, a "hyperlinks" is a reference on the web. Hyperlinks can point to any resources on the web. An anchor is a term used to define a hyperlinkdestination inside a document. Format of anchor tag is:

<a href="address"> Line Text </a>

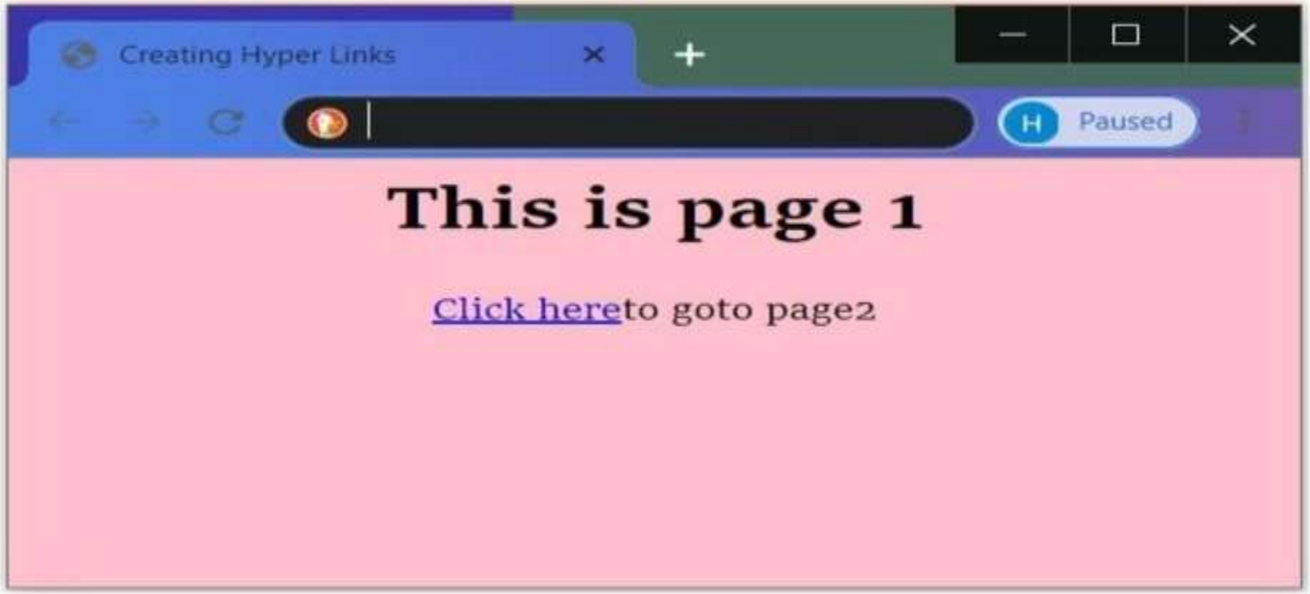
The <a> anchor tag has the following attributes.

1. href: It holds the target URL of the hyperlink.
2. Id: A unique alphanumeric identifier for the tag, which we can use to refer toit.
3. name: It specifies an anchor name, the name we want to use when referringto enclose items.
4. Target: This attribute defines where the linked document will be opened.

**Example:**

```
<html>
 <head>
  <title>Creating Hyper Links</title>
  </head>
  <body bgcolor="pink">
    <center><h1>This is page 1</h1>
    <a href="page2.html">Click here</a>to goto page2
    </center>
  </body>
</html>
```

**Output**



This is page 1

Click hereto goto page2

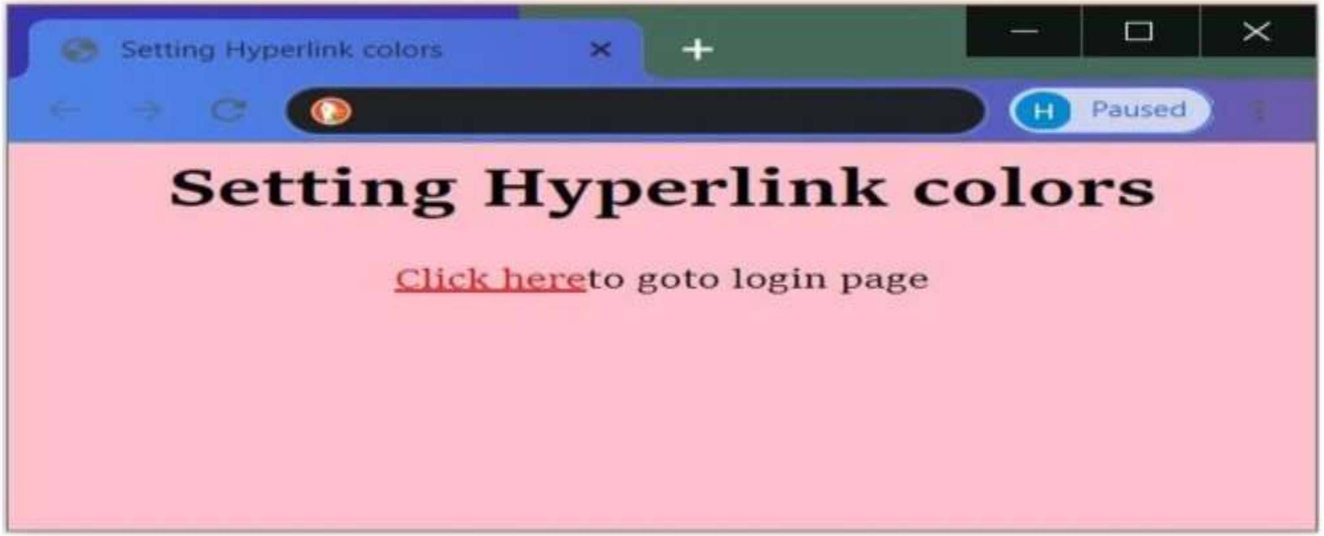
**Setting hyperlink colors:**

The default color of hyperlinks in a page is blue. Hyperlink that we have already visited are displayed in violet and when we click a hyperlink,it turns red when themouse button is down.We can set these colos in <body> tag attributes link,vlink(visited link),alink(active link).

**Example**

```
      <html>
       <head>
         <title>Setting Hyperlink colors</title>
       </head>
       <body bgcolor="pink" link="green" vlink="blue" alink="red">
       <center><h1>Setting Hyperlink colors</h1>
       <a href="login.html">Click here</a>to goto login page
       </body>
     </html>
```

**Output**



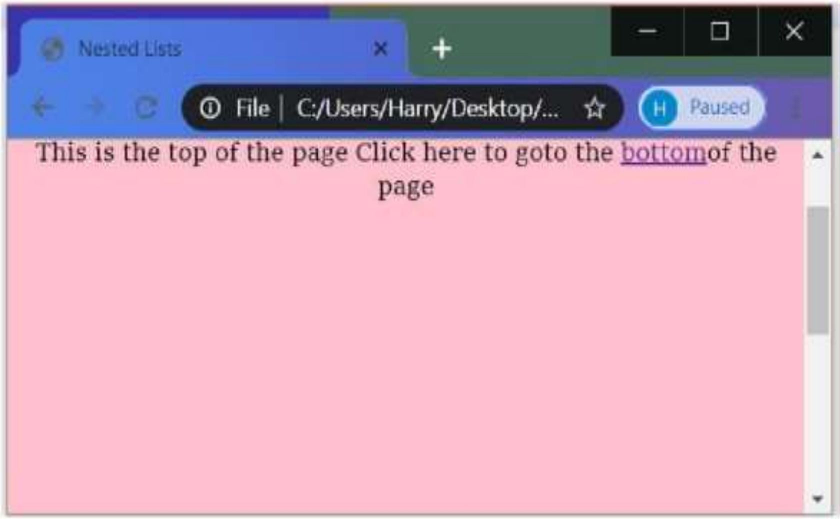
Setting Hyperlink colors

Click hereto goto login page

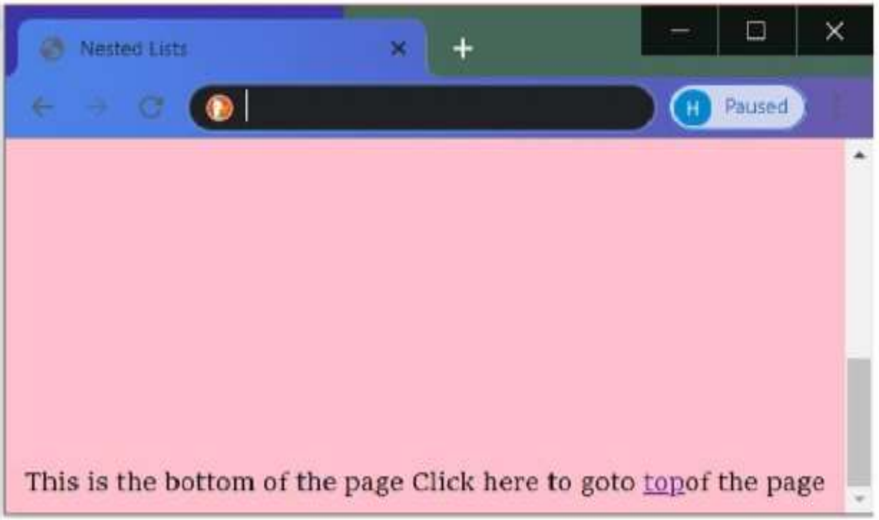
**Providing navigation with in the page:**

```
<html>
   <head>
    <title>Nested Lists</title>
  </head>
  <body bgcolor="pink">
  <center><h1>Linking to a section in a page</h1>
  <a name=”top”>This is the top of the page</a>
  Click here to goto the <a target=”#bottom”>bottom</a>of the page
<br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br>
<br><br><br><br><br>
<a name=”bottom”>This is the bottom of the page</a> Click here to
goto <a target=”#top”>top</a>of the page
</center>
</body>
</html>
```

**Output**



This is the top of the page Click here to goto the bottomof the page



This is the bottom of the page Click here to goto topof the page

**Creating HML tables:**

A HTML table arranges data/information in terms of rows and columns.Tables are defined in HTML using <table> tag.A table is divided into rows and each row each divided into data cells(columns).The rows of table are created using <tr> tag and data cells are created by <td> tag.

<tr> - Table row

<td> - Table data

**Format**

```
<table>
      <tr>
           <td>row1,col1</td>
           <td>row1,col2</td>
       </tr>
      <tr>
           <td>row2,col1</td>
           <td>row2,col2</td>
       </tr>
 </table>
```

- Heading in a table are defined with <th> tag

**Format**

```
<table>
      <tr>
           <th>heading 1</th>
           <th>heading 2</th>
       </tr>
      <tr>
           <td>data1</td>
           <td>data2</td>
       </tr>
 </table>
```

**Attributes of <table> tag:**

- align : specifies the horizontal alignment of the table in the browserwindow,set to ''left,center,right''.
- background : specifies the URL of a background image to be used asbackground for the table.
- bgcolor : sets the background color of the table cells.
- border : sets the border width.
- bordercolor : sets the external border color of the entire table.
- cellpadding : sets the spacing between cell walls and content.
- cellspacing : sets the spacing between table cells.
- height : sets the height of the whole table.
- width : sets the width of the table.

**Attributes of <tr> tag:**

- align : specifies the horizontal alignment content in the table cells set to "left,center,right".
- bgcolor : sets the background color of the table cells.
- bordercolor : sets the external border color of the entire table.
- Valign : sets the vertical alignment of data,set to top,middle,bottom.

**Alignment of <td> tag**

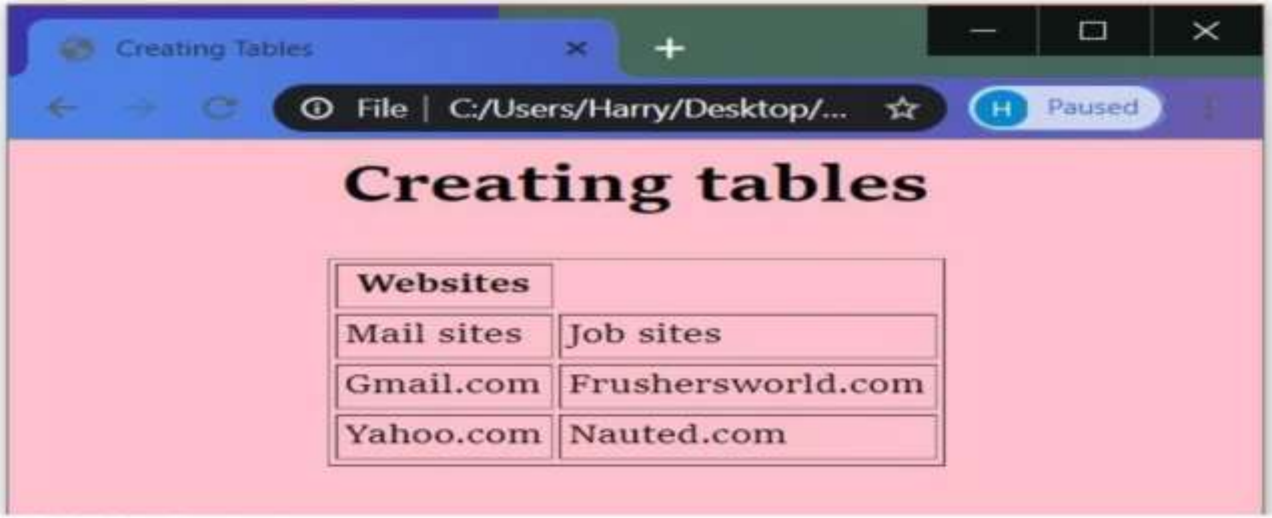
- align : specifies the horizontal alignment content in the table cells set to "left,center,right".
- bgcolor : sets the background color of the table cells.
- bordercolor : sets the external border color of the entire table.
- colspan : indicates the how many cell columns of the table this cell shouldspan.
- rowspan : indicates the how many cell rows of the table this cell shouldspan.

**Example**

```
<html>
 <head>
```

```
  <title>Creating Tables</title>
 </head>
 <body bgcolor="pink">
 <center><h1>Creating tables</h1>
 <table border=”1” cellpadding=”3” cellspacing=”3”>
      <tr>
            <th colspa=”2”>Websites</th>
      </tr>
       <tr>
              <td>Mail sites</td>
              <td>Job sites</td>
      </tr>
      <tr>
              <td>Gmail.com</td>
              <td>Frushersworld.com</td>
      </tr>
   <tr>
              <td>Yahoo.com</td>
              <td>Nauted.com</td>
  </tr>
</center>
</table>  </body></html>
```

**Output**



# Creating tables

| Websites | |
|---|---|
| Mail sites | Job sites |
| Gmail.com | Frushersworld.com |
| Yahoo.com | Nauted.com |

**Advanced Table elements :**

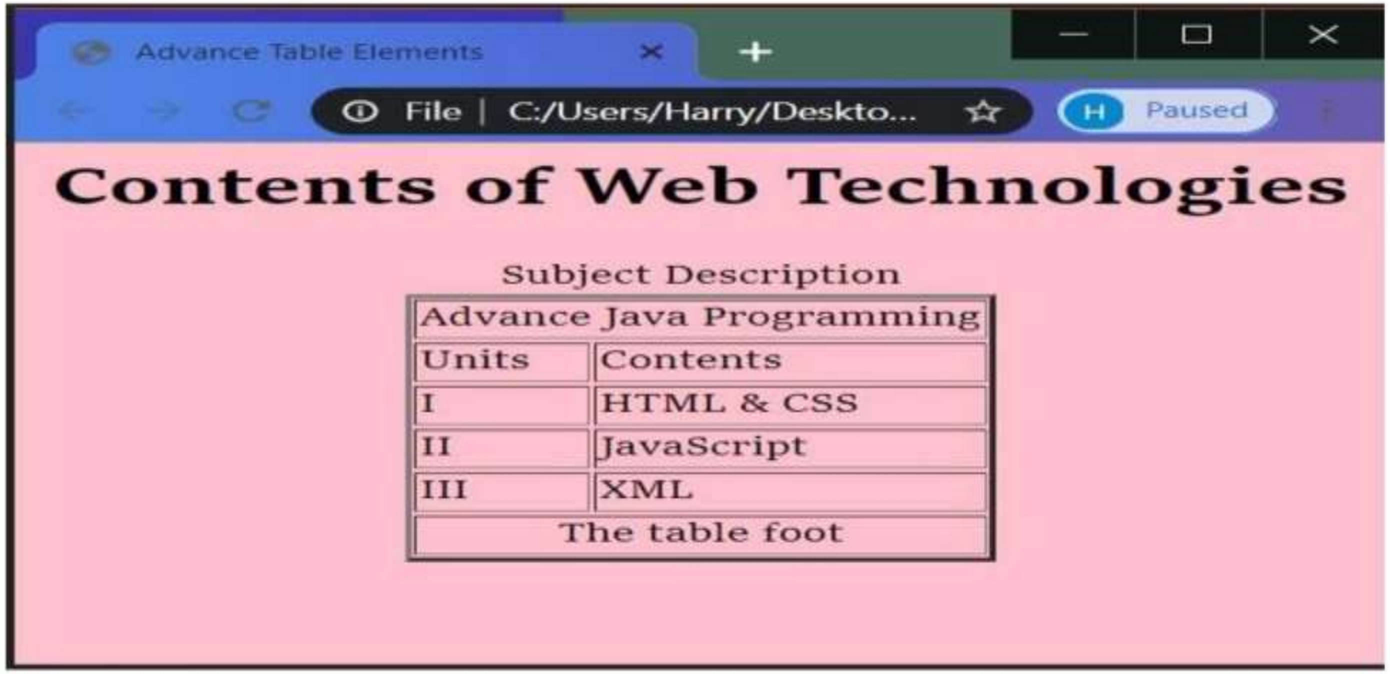
- <caption> : the element is an ptional element and it used to provide a string which describes,the contern of the table ,it must follow the table element.
- <thead> : The rows in a table can bc groudated one are more time we cancreate a table by using this <thead>.
- <tbody> : creates a table body when grouping rows.
- <tfoot> : Creates a table foot when groupin rows

**Example:**

```
<html>
  <head>
    <title>Advance Table Elements</title>
  </head>
  <body bgcolor="pink">
  <h1 align="center">Contents of Web Technologies</h1>
  <center>
     <table border="2">
```

```
          <caption>Subject Description</caption>
      <thead>
          <tr> <td colspan="2">Advance Java Programming</td>
      </thead>
      <tbody>
          <tr> <td>Units</td>
              <td>Contents</td>
          </tr>
          <tr> <td>I</td>
               <td>HTML & CSS</td>
          </tr>
          <tr> <td>II</td>
               <td>JavaScript</td>
          </tr>
          <tr> <td>III</td>
               <td>XML</td>
           </tr>
      </tbody>
      <tfoot align="center">
          <tr>
               <td colspan="2">The table foot</td>
          </tr>
</tfoot>
</table>
</center>
     </body>
    </html>
```

**Output**



# Contents of Web Technologies

Subject Description

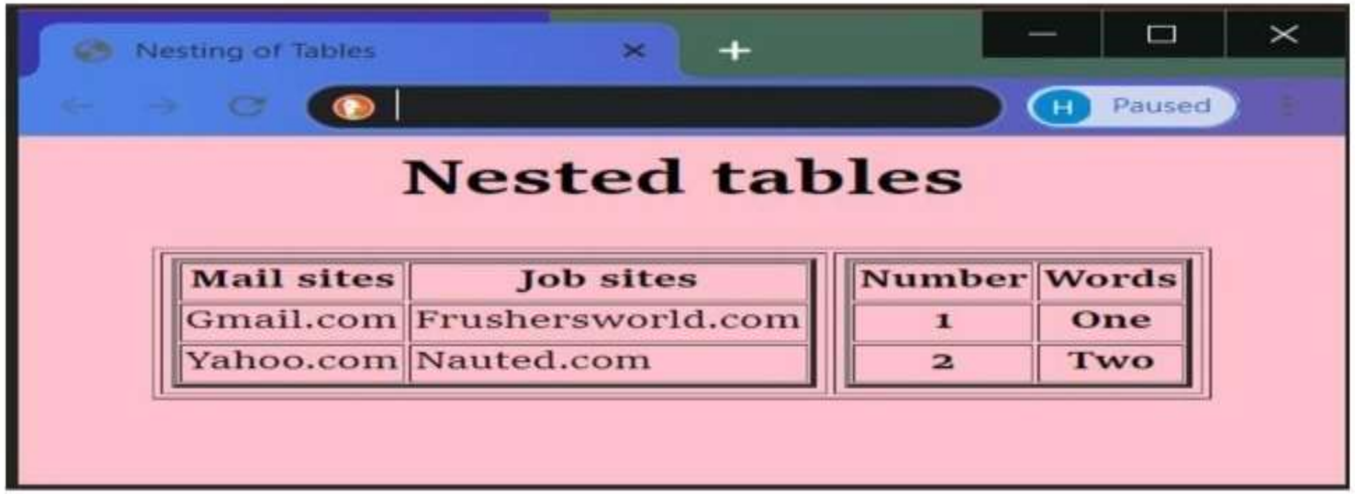
| Advance Java Programming | |
|---|---|
| Units | Contents |
| I | HTML & CSS |
| II | JavaScript |
| III | XML |
| The table foot | |

**Nesting of Tables:**

```
<html>
  <head>
    <title>Nesting of Tables</title>
  </head>
  <body bgcolor="pink">
  <center><h1>Nested tables</h1>
  <table border="1" cellpadding="3" cellspacing="3">
    <tr>
     <td>
       <table border="2">
        <tr>
               <th>Mail sites</th>
               <th>Job sites</th>
       </tr>
       <tr>
               <td>Gmail.com</td>
               <td>Frushersworld.com</td>
       </tr>
       <tr>
               <td>Yahoo.com</td>
               <td>Nauted.com</td>
       </tr>
      </table>
     </td>
    <td>
       <table border="2">
          <tr>
             <th>Number</th>
             <th>Words</th>
          </tr>
          <tr>
             <th>1</th>
             <th>One</th>
          </tr>
          <tr>
             <th>2</th>
             <th>Two</th>
          </tr>
        </table>
      </td>
    </tr>
</table>
</center>
 </body>
```

</html>

**Output**



Nested tables

| Mail sites | Job sites |
|---|---|
| Gmail.com | Frushersworld.com |
| Yahoo.com | Nauted.com |

| Number | Words |
|---|---|
| 1 | One |
| 2 | Two |

**Images in HMTL:**

In HTML we have the capability of displaying images in a webpage.This imagesmust be in a format that the web browser can handle , such as Graphics Interchange Format(GIF) , Joint Photograph Expert Group(JPEG) , and for somebrowser Portable Network Graphics(PNG) formats.

Displaying images in webpage is done by using <img> tag

**Format**

```
<img src=”URL of image source”>
```

**Attributes of <img> tag:**

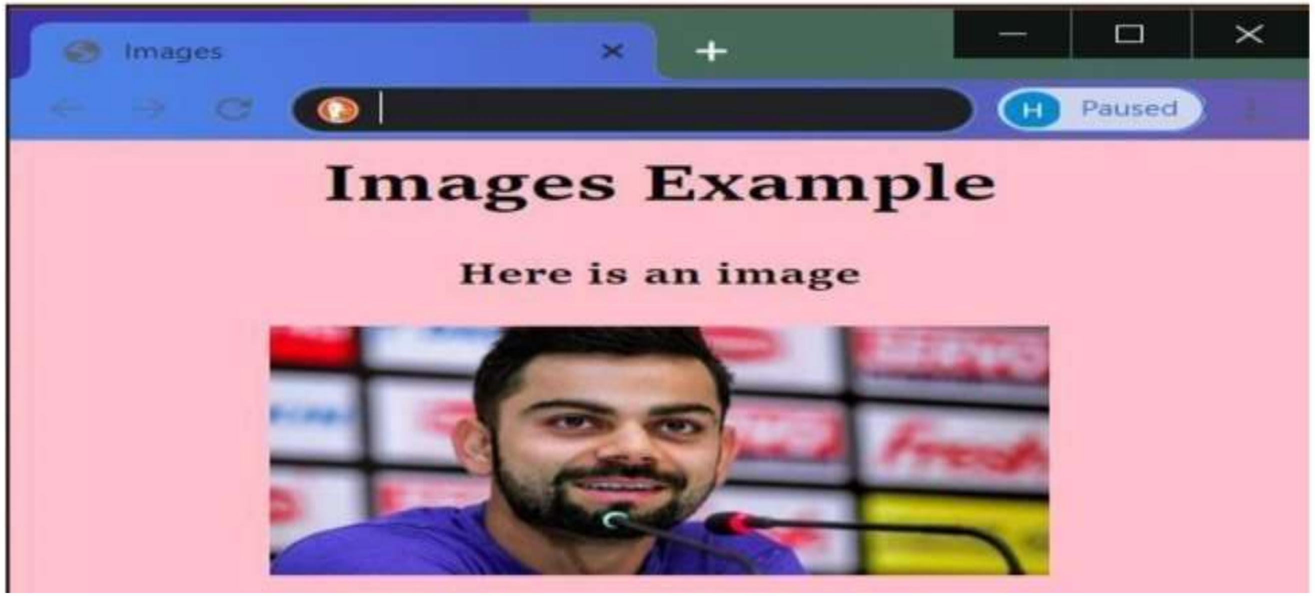
- alt : this attribute is used to specify text to be displayed in place of image forbrowser that cannot handle graphics.
- src : specifies the URL of the image to display.
- border : sets the border for the image.
- height : indicates the height of the image.
- width : indicates the width of the image.

- hspace : sets the horizontal space around the image.
- vspace : sets the vertical space around the image.

**Example**

```
<html>
<head>
 <title>Images</title>
</head>
<body bgcolor="pink">
 <center>
  <h1>Images Example</h1>
  <h3>Here is an image</h3>
  <img src="one.jpg" alt="here is an image" width="300" height="150">
 </center>
</body>
</html>
```

**Output**



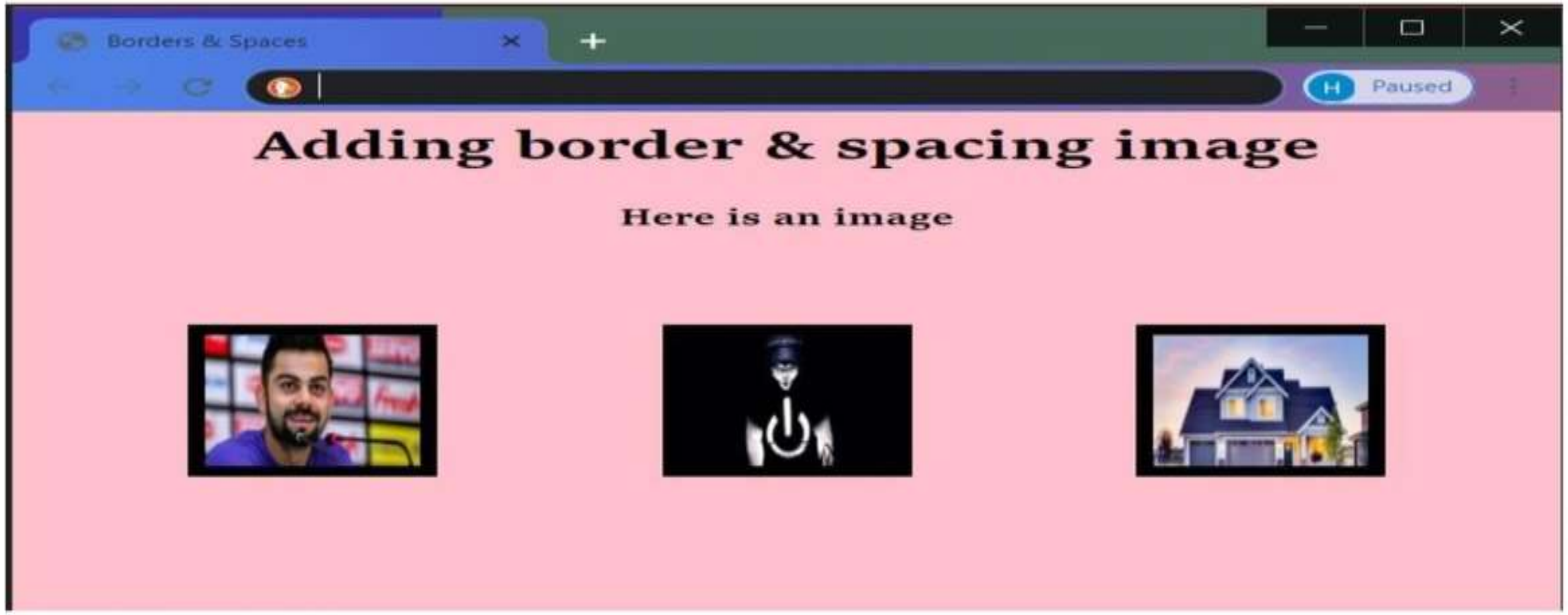
Images Example

Here is an image

**Adding borders and spaces around image:**

```
<html>
<head>
 <title>Borders & Spaces</title>
</head>
<body bgcolor="pink">
 <center>
  <h1>Adding border & spacing image</h1>
  <h3>Here is an image</h3>
  <img src="one.jpg" alt="here is an image" width="100" height="100"
hspace="50" vspace="50" border="8">
  <img src="two.png" alt="here is an image" width="100" height="100"
hspace="50" vspace="50" border="8">
  <img src="three.png" alt="here is an image" width="100" height="100"
hspace="50" vspace="50" border="8">
 </center>
</body>
</html>
```

**Output:**



Adding border & spacing image

Here is an image

**Creating HTML Forms:**

Form is a collection of various HTML control files , buttons ,checkboxes , radio buttons ,text fields et., and they use to send the data to the server. There are severalform elements.

- Button : <input type=”button”> :- are the standard clickable buttons.
- Checkbox : <input type=”checkbox”> :- displayed usually as a small boxwith a check mark in it. The use can toggle the checkbox on or off by checking the checkbox
- Customizable Buttons : <button> :- display images one other HMTL insideitself.
- File uploading controls : <input type=”file”> :- allow the user to upload filesto the server.
- Hidden controls : store data that is not visible to users unless they view theweb page source code.
- Image controls : <input type=”image”> :- are like submit buttons except thatthey are images the user can click.
- Password controls : <input type=”password”> :- are like text fields , buteach typed character displaying by an asterisk or instead any character.

- Radio buttons : <input type="radio"> :- displaying usually as a circle whichwhen selected displayed a dot in the middle. These controls are much like checkboxes except that they work int mutually exclusive at a time.
- Reset button : <input type="reset"> : - allow the user to clear all the data they has entered.When the user click reset button all controls in the form areremoved to that original state displaying the data they had when they first appeared.
- Selection : Works much like drop down list boxes also called select controlsFormat is:

```
<select>
        <option>Item1</option>
        <option>Item1</option>
        <option>Item1</option>
</select>
```

- Submit button : when we click the button all the data in the form will be sentto web server for processing.
- Text area : are two dimensional text fields allowing user to enter more thanone line of text.

  Format is: <textarea>
- Text fields : allow the user to enter one line of text also called a textboxFormat is : <input type="text">

In order to create form we use <form> tagFormat is :

```
<form>
 |
 |
</form>
```

**Attributes of <form> tag:**

- name : gives the name of the form so that we can return it in code . Set to analphanumeric string.
- target : indicates a named frame for browser to display the form results.

- method : indicates a method or protocol for sending data to the target actionURL.
- action : gives the URL that that will handle the form data.

**Example**

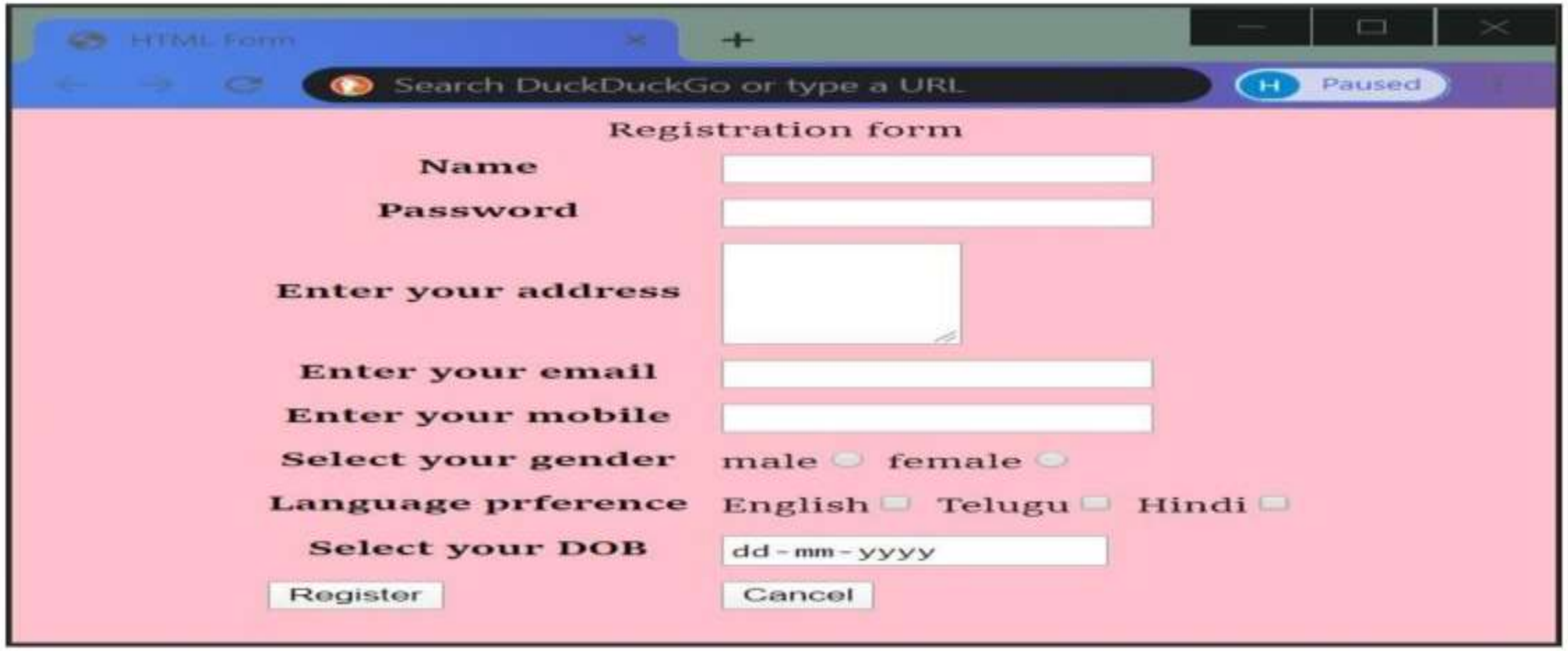
Registration.html

```
<html>
<head>
<title>HTML Form</title>
</head>
<body bgcolor="pink">
<center>
<form name="form1">
<table border="0" cellpadding="4" cellspacing="4">
<caption>Registration form</caption>
<tr>
<th>Name</th>
<td><input type="text" name="name" /></td>
</tr>
<tr>
<th>Password</th>
<td><input type="password"/></td>
</tr>
<tr>
<th>Enter your address</th>
<td><textarea rows="5" cols="10"></textarea></td>
</tr>
<tr>
<th>Enter your email</th>
<td><input type="email"/></td>
</tr>
<tr>
```

```
<th>Enter your mobile</th>
<td><input type="number"/></td>
</tr>
<tr>
<th>Select your gender</th>
<td>
male<input   type="radio"   name="g"   value="m"/>
female<input type="radio" name="g" value="f"/>
</td>
</tr>
<tr>
<th>Language prference</th>
<td>
English<input   type="checkbox"   value="   "/>
Telugu<input   type="checkbox"   value="   "/>
Hindi<input type="checkbox"  value=" "/>
</td>
</tr>
<tr>
<th>Select your DOB</th>
<td><input type="date"/></td>
</tr>
<tr>
<td ><input type="submit" value="Register"/></td>
<td><input type="reset" value="Cancel"/></td>
</td>
</tr>
</table>
</form>
</body>
```

```
</html>
```

**Output**



Registration form

| | |
|---|---|
| **Name** | |
| **Password** | |
| **Enter your address** | |
| **Enter your email** | |
| **Enter your mobile** | |
| **Select your gender** | male ○ female ○ |
| **Language prference** | English □ Telugu □ Hindi □ |
| **Select your DOB** | dd - mm - yyyy |
| Register | Cancel |

**Working with Frames:**

HMTL frames allow user to present documents in multiple views which may beindependent windows or sub windows. To divide a webpage into multiple parts and load different pages in a single web page we use the concept of frames. To dothis we use "<frameset>" tag which indicates the browser that the webpage window has a frame. We can divide it into rows and columns by using attributes such as 'rows' & 'cols'. In order to provide definition or each frame we use "<frame>" tag.

Format is :

```
 <frameset rows="30%,70%">
      <frame src="source page URL" name="frame name">
      <frame src="source page URL" name="frame name">
 </frameset>
```

- <frameset> element actually takes place of <body> tag.

**Attributes of <frameset> tag:**

- border : used in the outermost <frameset> tag to set the border thickness forframes.
- bordercolor : set the color of the borders for all frames in the frameset.
- frameborder : set whether or not border for all frames in the frameset.Can be set to 'yes' or 'no' or '1' or '0'.
- framespacing : set the pixel spacing between frames. set to the positiveintegers.
- cols : set the number of columns in the frameset. Separate the values assigned to this attribute with comma(,) each value represents width of acolumn. Can be set to pixel values,percentages.
- Rows : set the number of rows in the frameset. Separate the values assigned to this attribute with comma(,) each value represents width of a column. Can be set to pixel values,percentages.

**Attributes of <frame> tag:**

- bordercolor : set the color used for the frmae border. This setting overridesthe color specified in the surrounding <frameset> element.
- frameborder : sets whether or not border surround the frame. Can be set to'yes' or 'no' or '1' or '0'.
- name : sets the name of the frame we can use named frames as target for <a>tag.
- scrolling : determines scrolling possible values are : auto,yes or no.
- src : specifies the URL of the frame document. If we don't specify a URLthe frame will appear blank.

**Creating vertical frames:**

In order to display vertical frame we have use of 'cols' attribute.

**Example**

```
<html>
<head>
<title>Vertical Frames</title>
</head>
```

```
<frameset cols="30%,70%">
        <frame src=frame1.html>
        <frame src=frame2.html>
</frameset>
</html>
```

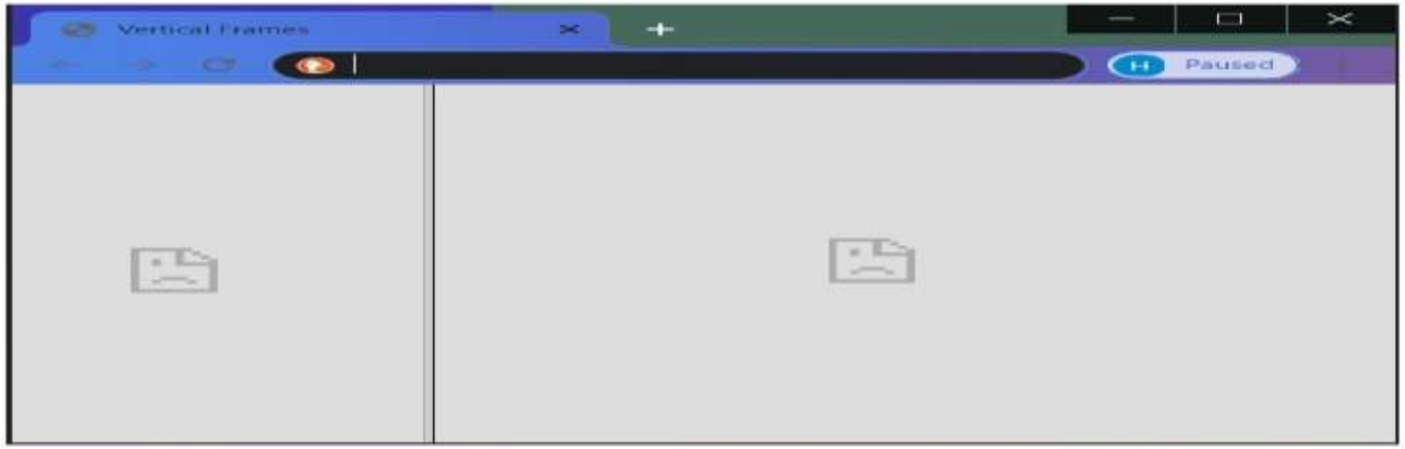
**Frame1.html**

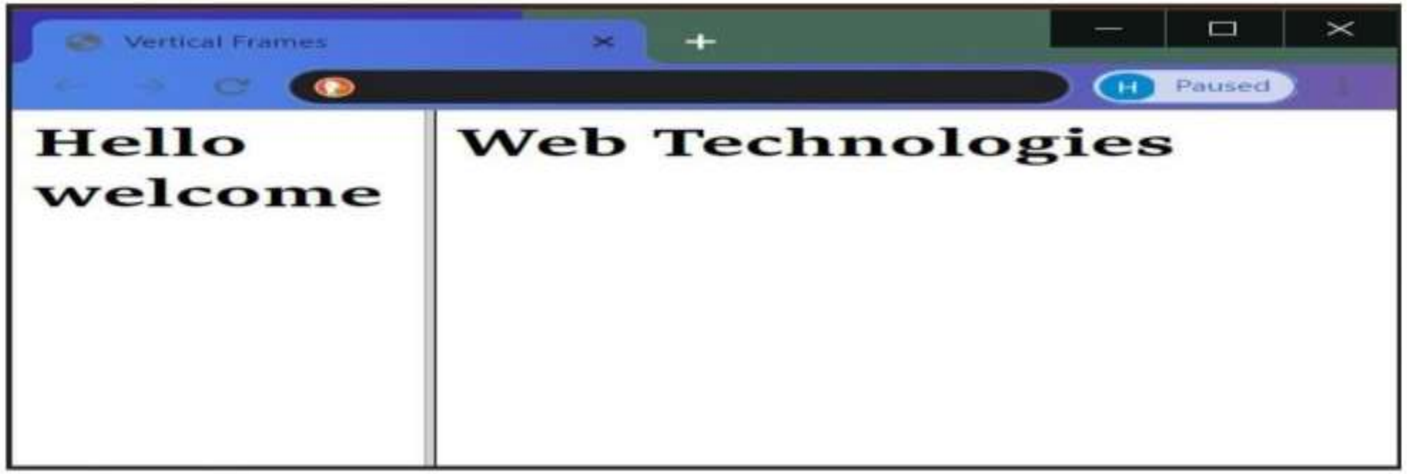
```
<html>
<head>
        <title>page1</title>
</head>
<body>
 <h1>Web Technologies</h1>
</body>
</html>
```

**Frame2.html**

```
<html>
<head>
        <title>page2</title>
</head>
<body>
 <h1>Web Technologies</h1>
</body>
</html>
```

**Output**





Hello welcome

Web Technologies

**Creating horizontal frames:**

In order to display horizontal frame we have use of ‘cols’ attribute.

**Example**

```
<html>
<head>
        <title>Horizontal Frames</title>
</head>
<frameset cols="50%,50%">
        <frame src=page1.html>
        <frame src=page2.html>
</frameset>
</html>
```

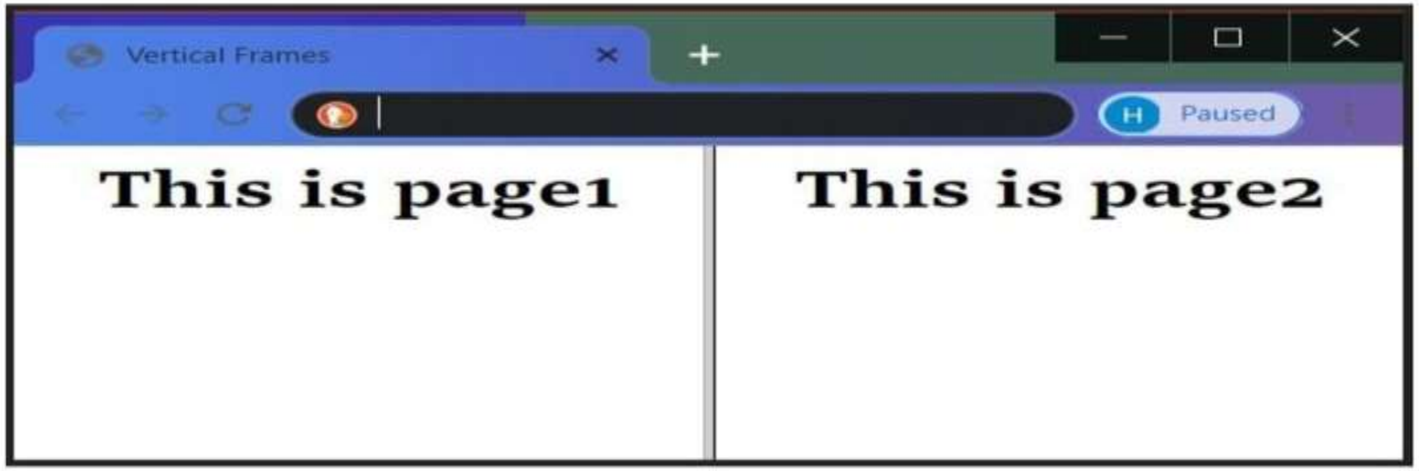
**page1.html**

```
<html>
<head>
        <title>page1</title>
</head>
<body>
 <h1 align="center">This is page1</h1>
</body>
</html>
```

**page2.html**

```
<html>
<head>
        <title>page2</title>
</head>
<body>
 <h1 align="center">This is page2</h1>
</body>
</html>
```

**Output**



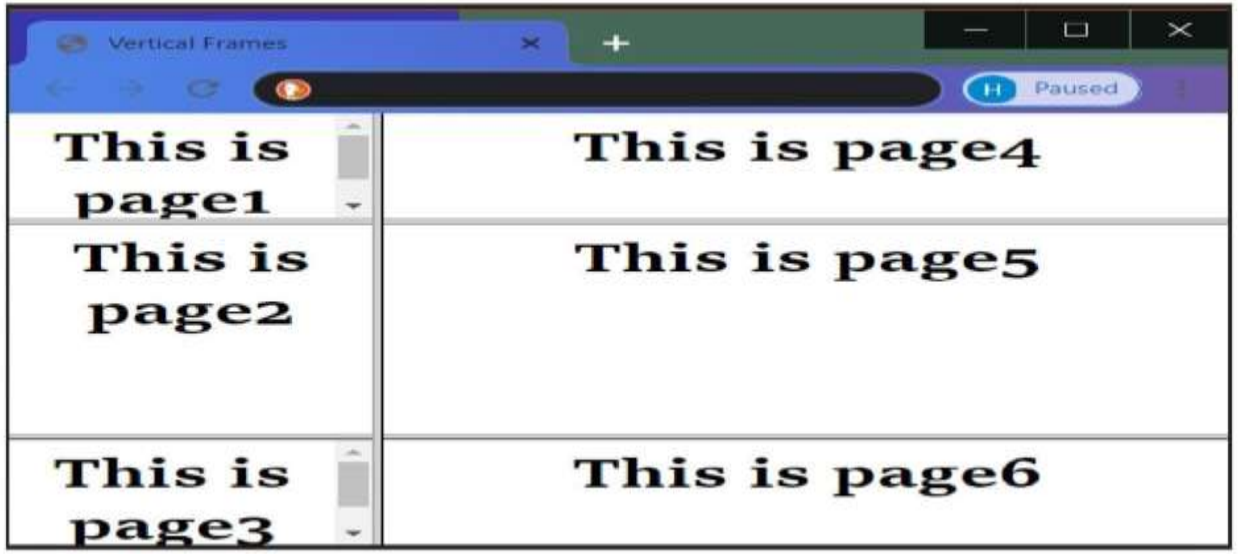
This is page1

This is page2

**Creating horizontal & vertical frames :**

```
<html>
<head>
          <title>Horizontal Frames</title>
</head>
<frameset cols="30%,70%">
     <frameset cols="25%,50%,25%">
          <frame src=page1.html>
          <frame src=page2.html>
          <frame src=page3.html>
     </frameset>
     <frameset cols="25%,50%,25%">
          <frame src=frame1.html>
          <frame src=frame2.html>
          <frame src=frame3.html>
     </frameset>
</frameset>  </html>
```

**Output**



This is page1

This is page4

This is page2

This is page5

This is page3

This is page6

**<noframes> tag:**

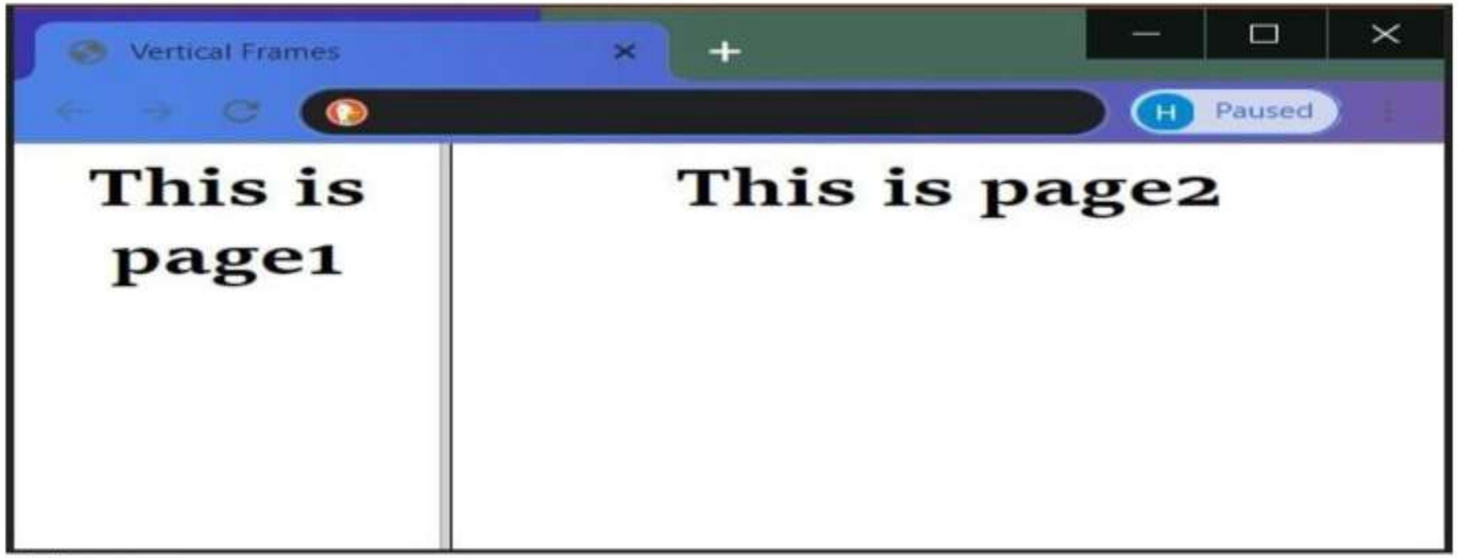
When the browser does not support frameset use <noframes> element to indicateto users that the browser doesnot support frames.The <noframes> element is ignored that handle frames.

**Example**

```
<html>
<head>
        <title>Vertical Frames</title>
</head>
<frameset cols="30%,70%">
        <noframes>Your browser does not support frames...</noframes>
        <frame src=on.html>
        <frame src=two.html>
</frameset>
```

```
</html>
```

**Output**



This is page1

This is page2

**Named frames:**

One important aspect of working with frames is using named frames. When wegive frmae a name , we can use as a target to load new page into the frame.

**Example**

```
<html>
<head>
        <title>Vertical Frames</title>
</head>
<frameset cols="40%,60%">
        <frame src=menu.html>
        <frame src=default.html name="display">
</frameset>
</html>
```

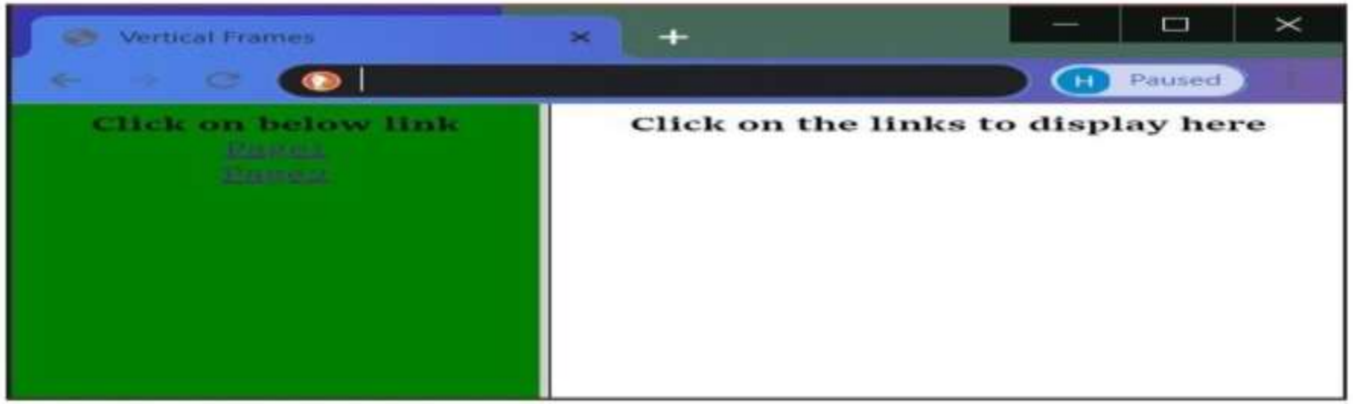
**menu.html**

```
<html>
<head>
        <title>Menu</title>
</head>
<body bgcolor="green">
  <center><b>Click on below link</b>
        <br><a href="page1.html" target="display">Page1
        <br><a href="page2.html" target="display">Page2
  </center>
</body>
</html>
```

**Defual.html**

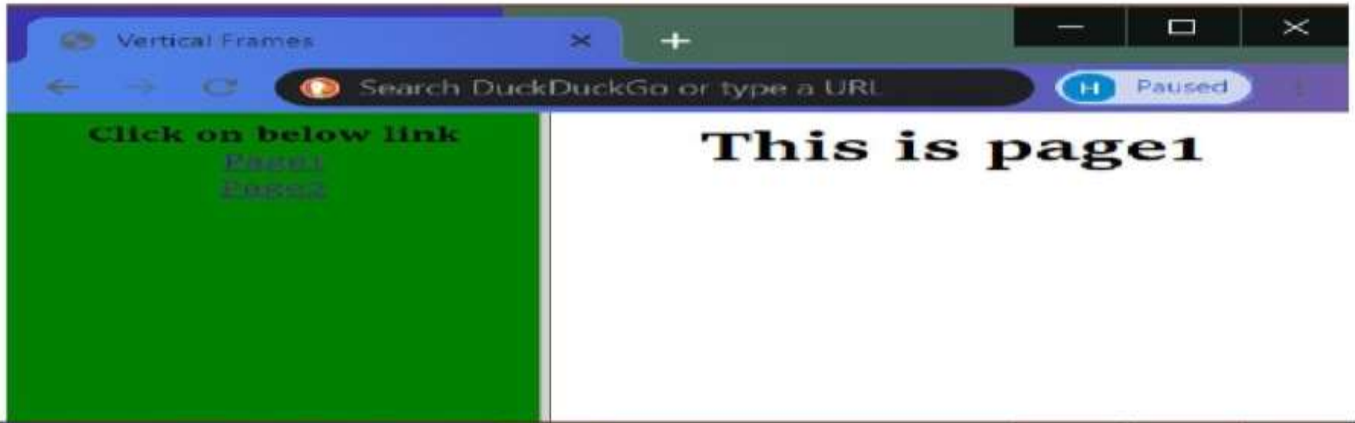
```
<html>
<head>
        <title>Vertical Frames</title>
</head>
<frameset cols="40%,60%">
        <frame src=menu.html>
        <frame src=default.html name="display">
</frameset>
</html>
```

**Output**



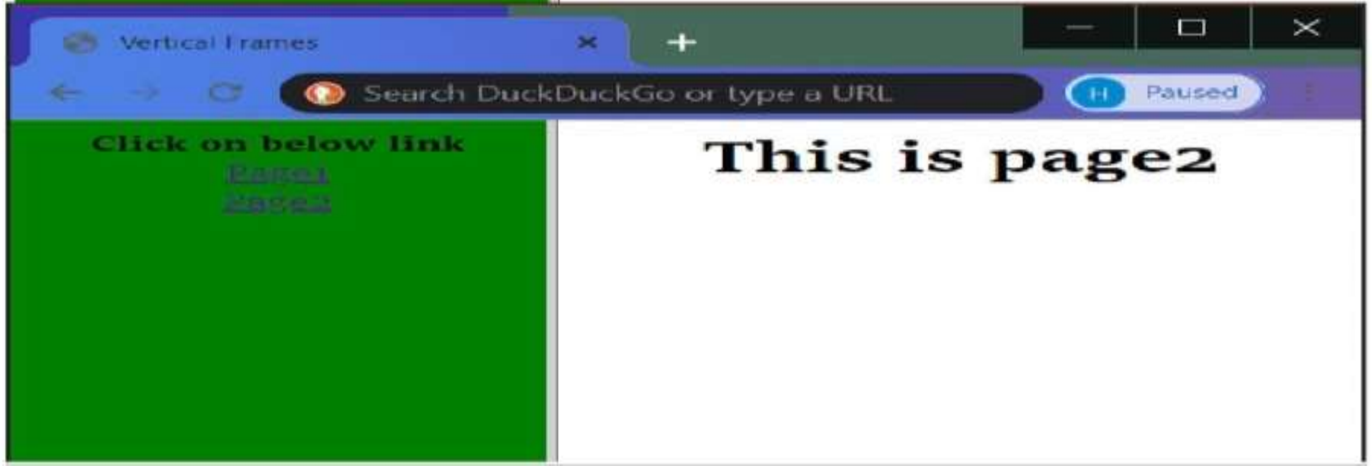
Click on below link
Page1
Page2

Click on the links to display here



Click on below link
Page1
Page2

This is page1



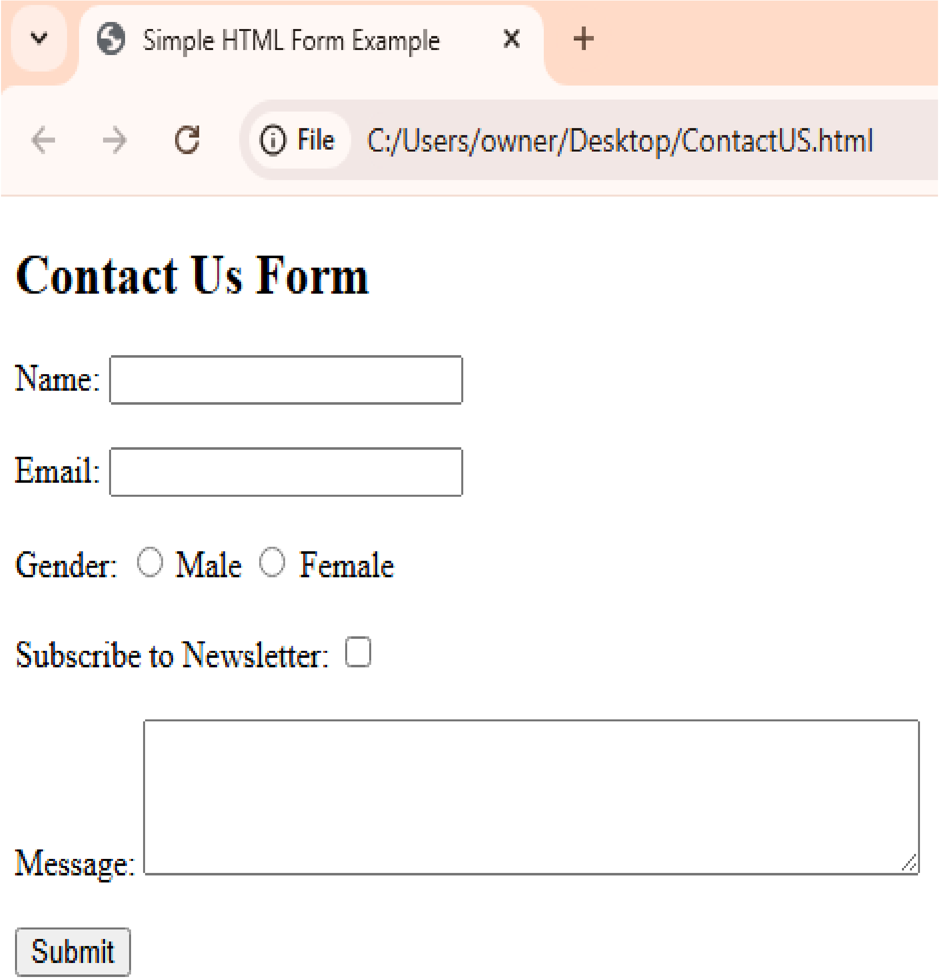
Click on below link
Page1
Page2

This is page2

**<form> tag  -** In HTML, the <form> tag defines an HTML form for user input, acting as a container for various form elements like text fields, checkboxes, and submit buttons, enabling data collection and submission to a server

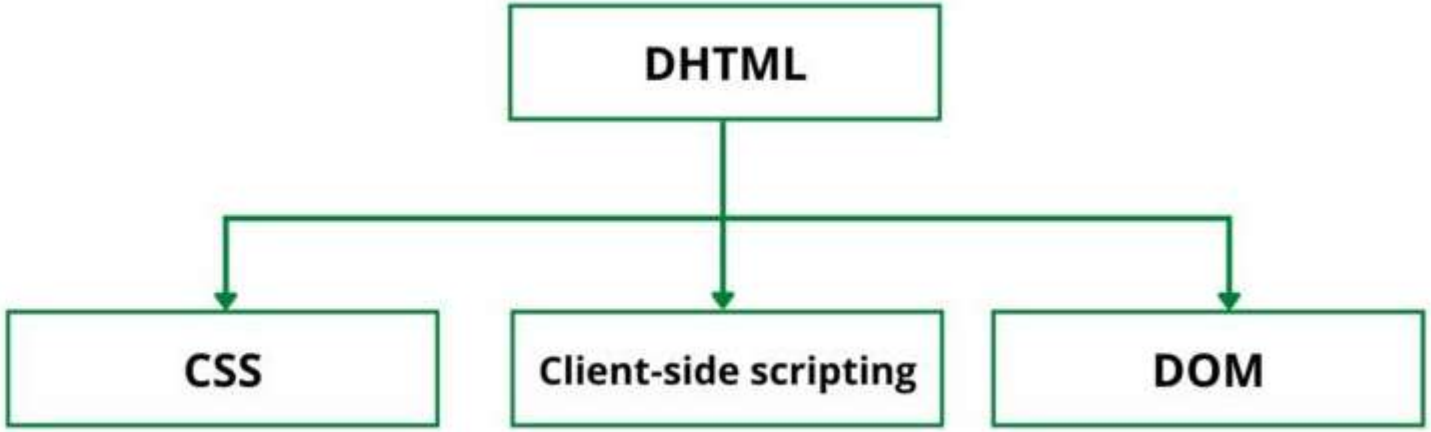
**Purpose**:
The primary function of the <form> tag is to create a structure for users to input data, which can then be sent to a server for processing.
**Container**:
The <form> tag acts as a container for various form elements, such as <input>, <textarea>, <select>, <button>, and <label>.
**Attributes**:
**action**: Specifies the URL where the form data will be sent for processing.
**method**: Specifies the HTTP method (GET or POST) used to submit the form data.
**name**: Specifies a name for the form, which can be used for scripting purposes.
**target**: Specifies where to display the response that is received after submitting the form.

```
 <!DOCTYPE html>
<html lang="en">
<head> <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Simple HTML Form Example</title>
</head>
<body>
<h2>Contact Us Form</h2>
<form action="/submit_form" method="POST">
    <!-- Name field -->
    <label for="name">Name:</label>
    <input type="text" id="name" name="name" required>
    <br><br>
    <label for="email">Email:</label>
    <input type="email" id="email" name="email" required>
    <br><br>
    <label>Gender:</label>
    <input type="radio" id="male" name="gender" value="male">
    <label for="male">Male</label>
    <input type="radio" id="female" name="gender" value="female">
    <label for="female">Female</label>
    <br><br>
<label for="newsletter">Subscribe to Newsletter:</label>
    <input type="checkbox" id="newsletter" name="newsletter" value="yes">
    <br><br>
      <label for="message">Message:</label>
    <textarea id="message" name="message" rows="4" cols="50" required></textarea>
    <br><br>
    <input type="submit" value="Submit">
</form></body></html>
```



**DHTML DHTML, or Dynamic HTML**, is a technology that differs from traditional HTML. DHTML combines HTML, CSS, JavaScript, and the Document Object Model (DOM) to create dynamic content. It uses the Dynamic Object Model to modify settings, properties, and methods. Scripting is also an essential component of DHTML, which was part of earlier computing trends. It is supported by some versions of Netscape Navigator and Internet Explorer 4.0 and higher.

1. DHTML stands for Dynamic Hyper Text Markup Language.
2. DHTML is a combination of technologies used to create dynamic web pages.
3. DHTML means a combination of HTML, CSS, DOM and JavaScript.
4. DHTML is designed to enhance the user experience on web.
5. DHTML includes the following features :
    a. Dynamic content, which allows the user to dynamically change web page content.
    b. Dynamic positioning of web page elements.
    c. Dynamic style, which allows the user to change the web page's colour, font, size or content.



DHTML

CSS

Client-side scripting

DOM

## Difference between HTML and DHTML

| S. No. | HTML | DHTML |
|---|---|---|
| 1 | HTML is used to create static web pages. | DHTML is used to create dynamic web pages. |
| 2 | HTML consists of simple HTML tags. | DHTML is made up of HTML tags, Cascading Style Sheets (CSS) and JavaScript. |
| 3 | HTML does not allow to alter the text and graphics on the web page unless web page gets changed | DHTML allows to alter the text and graphics of the web page without changing the entire web page. |
| 4 | HTML web pages are simple but less interactive. | DHTML web pages are complex but more interactive. |