HTML hypertext markup language

 data presentation and UI designing (sign up, login, registration, search form…)

We can design static web pages

CSS Cascading Style Sheets

 Used to change look & feel of webpage (html elements)

It helps to present data more effectively, attractively & animations

JavaScript it provides logical support to html pages

 its works like as back-end for html/css

used to develop responsive web pages

Html, css & JavaScript are web technologies, these techs are used to develop **web applications**.

WHAT IS APPLICATION OR SOFTWARE?

Application is an Automation process of manual business operations (human being work) by using a programming language.

TYPES OF APPLICATIONS OR SOFTWARES

We can create an application or software in following flavors:

1) **Desktop:** The applications which are installable in local systems are called desktop applications.
2) **Mobile:** The applications which are installable in mobile phones or tablets downloaded from play store for android and apple store for ios.
3) **Web:** The applications which are deployable in any server and can be accessible from any location using a browser.

WHAT IS WEB APPLICATION?

Web applications are network enabled applications. We can deploy any web applications in servers and we can access them over the network using server ip address and application name.

In computing, a **web application** is a client–server software **application** which the client (or user interface) runs in a **web browser** and it contains web documents in the form electronic pages (web pages).

A web application typically contains following three layers:

Presentation layer is a user interface (views) which are accessible from any web browser.

Business layer is a server-side program which is nothing but automation of business rules. Client layer will interact with the business layer to persist data.

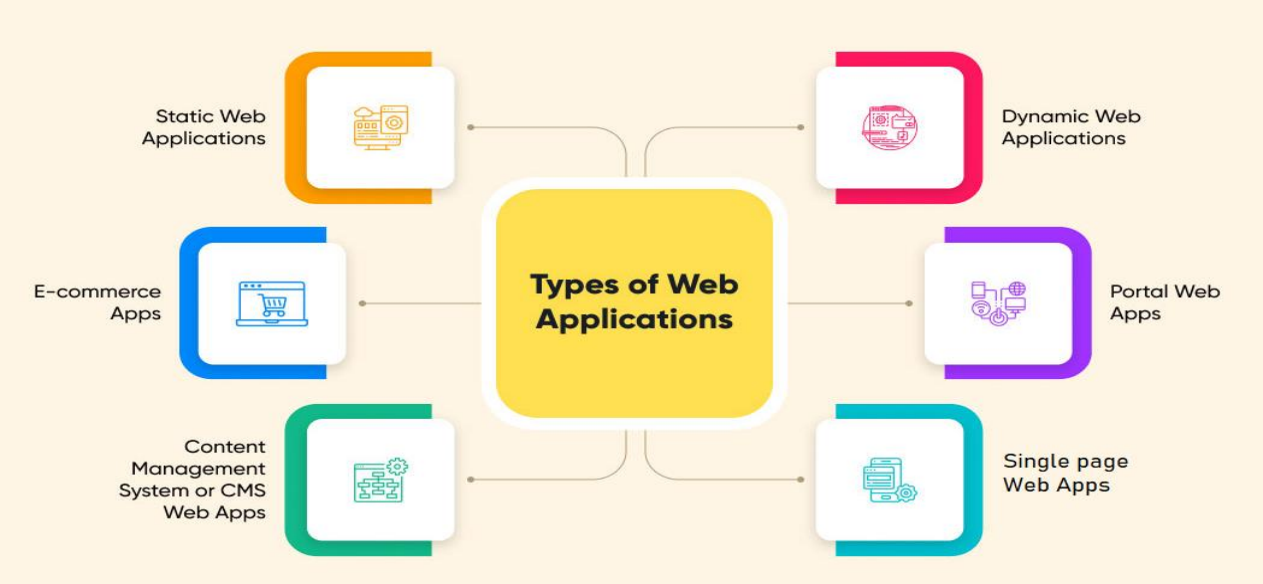
Data layer is database software where we can store client related data. Business layer will interact with the data layer.

HOW MANY TYPES OF WEB APPLICATIONS WE HAVE?

A webpage is an electronic page developed on HTML.

A web page is a group of elements.

Collection of WebPages or web documents is called web application (website).



STATIC WEB APPS:

- ⇨ The static web app directly delivers the content to the end user's browser without fetching any data from the server. Most static web apps are known to be simple and effortless to develop across the web.
- ⇨ HTML, CSS, and JavaScript are used to develop most static web applications.
- ⇨ You can also add GIFs, animations, videos in static web apps.

DYNAMIC WEB APPS:

- ⇨ A web application that generates the data in real-time based on the user's request and server response, is known as a dynamic web application.
  Eg: Facebook

SINGLE PAGE APPS:

- ⇨ A single-page application runs entirely within a browser and doesn't require page reloading.

  Eg: Gmail

E-COMMERCE APPS:

⇨ A web application that helps users electronically buy or sell goods over the internet is called an e-commerce web app.

Eg: Amazon

CMS APPS

A content management system (CMS) allows website owners to create, edit, and publish content, including images, text, and video, without involving a technical team. You can modify content through an admin panel without any knowledge of programming language.

Ex: Webflow, Wordpress

PORTAL WEB APPS:

⇨ A portal web application is a website that provides access to many different pages and links.

⇨ Portal web applications are often used for online shopping, news sites, blogs, and search engines.

⇨ A portal web application's main idea is to browse through different content without leaving the site.

⇨ **Eg:** telangana.gov.in

Network

Collection of computers interlinked together is called a network. First network name is **ARPANET** (Advanced Research Projects Agency Network). First protocol in the IT industry is FTP (File Transfer Protocol).

Internet

Internet stands for international networking.

The Internet is a network of connected computers. No company owns the Internet; it is a cooperative effort governed by a system of standards and rules. The purpose of connecting computers together, of course, is to share information.

A Brief History of the Web

The Web was born in a particle physics laboratory (CERN) in Geneva, Switzerland in 1989. There a computer specialist named **Tim Berners-Lee** first proposed a system of information management that used a "hypertext" process to link related documents over a network. He and his partner, **Robert Cailliau**, created a prototype and released it for review. For the first several years, web pages were text-only.

Tim Berners-Lee

- WEB (1989-1990)
- Html (HyperText Markup Lang)
- WWW
- W3C org

The World Wide Web Consortium

World Wide Web Consortium (called W3C) is the organization that oversees the development of web technologies. The group was founded in 1994 by TimBerners-Lee, the inventor of the Web, at the Massachusetts Institute of Technology (MIT).

Tim Berners-Lee (WWW/HTTP), Cerf & Kahn (TCP/IP), Baran, Davies, Klein rock & Roberts (packet networking), Bob Metcalfe (Ethernet).

Server

A **server** is a computer or system that provides resources, data, services, or programs to other machines, known as clients, over a network/inet.

In theory, whenever computers share resources with client machines, they are considered **servers.**

a **server** stores all the data associated with the websites that are hosted by it and shares that info with all computers and mobile devices (like yours) that need to access them.

Client

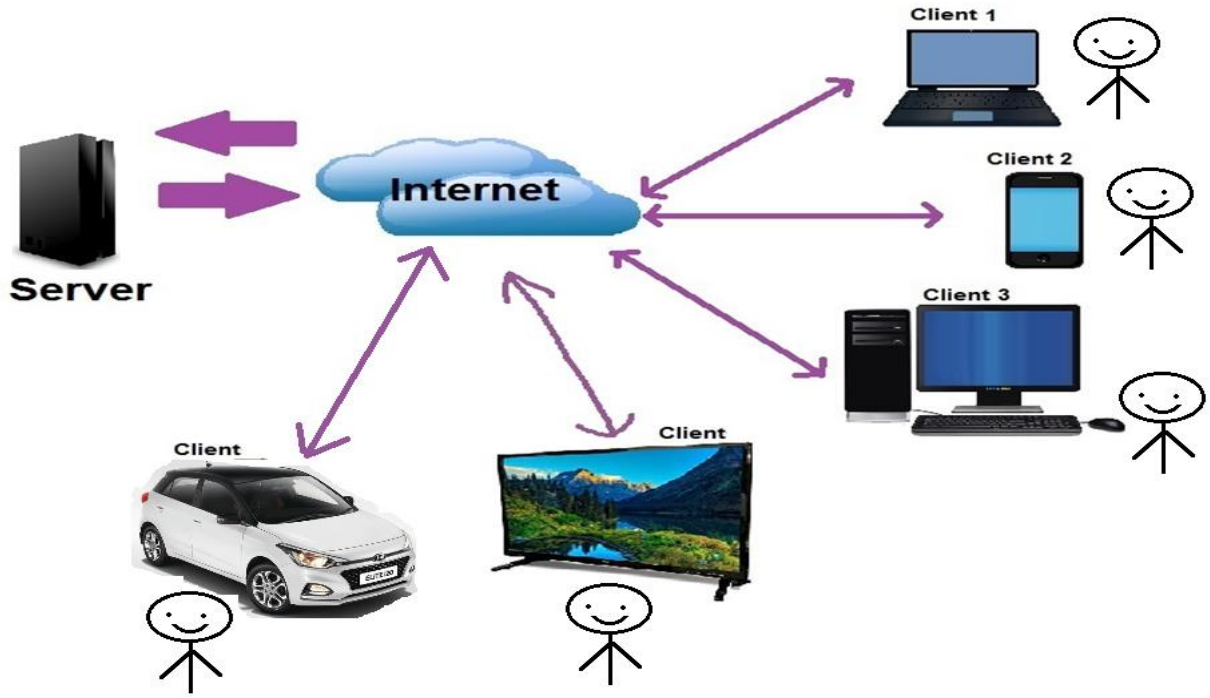
A client is a device that connects to and uses the resources of a remote computer, or server.

Clients may use a desktop or a laptop or a tablet or a mobile phone or a TV or a car etc.

The device which is used by the user is called a “Client”.

User

The person who is working on/operating a client machine is known as User or end-user.



What is web browser?

It is client-side lightweight software installed in client machine. It sends http request from client to server; it takes http response from server.

Browser provides navigation among web pages, and browsers executes html, css, JavaScript files and displays output to user.

List of Computer Browsers:

Internet Explorer(1995), Opera(1995), Mozilla Firefox(1998), Safari(2005), Google Chrome(2008) etc…

Email: Electronic mail services. It is a free service to communicate with other internet users. Email is invented by Shabeer Bhatia. Sabeer Bhatia is an Indian entrepreneur who founded the webmail company Hotmail.com.

SMTP: Simple Mail Transfer Protocol. It takes care of delivering emails from one server to another.

MIME: Multipurpose Internet Mail Extensions. It exchanges different kinds of data.

Blog: It is daily updating website or webpage. Every post displayed in reverse chronological order.

Forum: It is an online discussion website to exchange resources each other.

Http: It is a transfer protocol to exchange hypertext documents in the world wide web.

Http(s): Secured transfer protocol to exchange hypertext documents with the help of SSL (cipher text).

Cipher text is encrypted text. Plaintext is what you have before encryption, and cipher text is the encrypted result. The term cipher is sometimes used as a synonym for cipher text, but it more properly means the method of encryption rather than the result.

What is HTML?

It is specially designed hypertext for web browsers, with meaningful tags or elements in simple English language.

HTML Versions

From W3C organization there are fallowing versions released.

Version	Specification	Release Date
1.0	**N/A (HTML 1.0)**	**1993**
2.0	**HTML 2.0**	**24-Nov-1995**
3.2	**W3C: HTML 3.2**	**14-Jan-1997**
4.0	**W3C: HTML 4.0**	**24-Apr-1998**
4.1	**W3C: HTML 4.1**	**24-Dec-1999**
5.0	**WHATWG (Adv Markup Language For Mobiles)**	**28-Oct-2014**
5.1	**W3C: HTML 5.1 (Adv Markup Language For Small Electronic Devices)**	**-Nov-2016**
5.2	**W3C: HTML 5.2**	**14-Dec-2017**

HTML introduction

1. HTML was developed by “Tim-Berners-Lee”, released in 1993 and maintained by W3C Org.

GML ➔ SGML ➔ HTML

2. HTML stands for “Hypertext Markup Language”.

“**Hypertext**” means the text that can be transferred from internet server to internet client.

"Markup" means which syntax will be in the form of tags or you simply "markup" a text document with tags that tell a Web browser how to structure it to display.

“Language” is an interface between web developer and web browser

3. Technically, HTML is not a programming language, but rather a markup language.

4. HTML is used to design "static web pages", means HTML is used to create elements (such as headings, paragraphs, icons, menus, logos, images, textboxes, button etc) in the web pages.

static webpage means, that pages always showing same information.

5. HTML is very easy to understand (no pre-requisites).

6. HTML is “client side tech”. That means the html code executes on the client browser but not in server.

7. HTML is supported by all the browsers such as Google Chrome, Mozilla Firefox, Microsoft Internet Explorer, Safari, Opera and other browsers.

8. HTML is used in all real time web sites today; html is the only language available in world for designing Web pages.

9. The file extension either "filename.html" or "filename.htm"

10. HTML is an interpreter-based language. Browser interprets HTML code.
Translators: converting high level code (human) into machine level code (MP/OS) is called as translation. Who performs this operation those called translators.
Types: compiler, interpreter, assembler
interpreter ➔ it translates code line-by-line and executes line-by-line (interpretation)

interpreter
html code <==========> machine code
(English) (Binary code)

12. For working html no need installs any software, and browser is responsible for executing & producing output of html programs.

13. Html is error free programming.

14. HTML is not a case sensitive language that means you can write the html code in either upper case or lower case.

15. html is interpreter oriented language

Tag:

- A tag is a keyword, enclosed within "<" and ">" in HTML language.
- It is special kind of text placed between left angular brace and right angular brace(<tag_name>).
- Tag is predefined program, program is instructions / command to browser.
- Tag is used to display some specific output in the web page.
- Browser was not identified the tag; it shows blank page or it prints as text.
- tags also represented as elements.

types of tags:

in html we have **two** types tags, those are:

➔ **paired tags**

Contains open tag and closing tag.

opening tag specifies starting point of operation/output, closing tag specifies ending point of operation/output.

Syn: **<tagname>**something**</tagname>**

ex: <html> ... </html>
<head> ... </head>
<body> … </body>
<script> ... </script>
<style> ... </style>
 …

note: paired tags also called as body-full tags

➔ **unpaired tags**

contains only open tag.

VOID => ITS not RETURNING ANY VALUE

Syn: <tagname> or <tagname/>

ex:
 <img/><input/>
<hr>
<link>

note: Unpaired tags also called as body-less tags

Structure of HTML

as per **W3C** we have to follow the following structure to design web pages (but it's not comp).

```
<!DOCTYPE html>
<html lang="en">          ← web page/document designing starts here
    <head>
    </head>
    <body>
    </body>
</html>       ← web page/document designing ends here
```

Generally, html program contains three parts, those are:

> versioning section

> head section

> body section

versioning section

This is providing information to the browser which version we are using in the web page/program. So, browser is interpreting code and producing output as per given specification.

Syn:

```
<!DOCTYPE  html  version-url>
```

HTML4.0:

```
<!DOCTYPE html public "-//W3C//DTD HTML 4.0//EN"
"http://www.w3c.org/TR/html4/strict.dtd">
```

HTML5:

```
<!DOCTYPE  html>          it uses current version of html
```

strcit.dtd file (document type definition) contains definitions of tags and specifications.

html tag

The <html> tag represents the starting and ending of an html program. html tag contains two child/sub tags those are head tag and body tag.

head tag

head tag represents a non-content section (means not output) of the web page.

This information doesn't appear on the web page/in the browser (it's called as non-content), but it's used internally by the browser.

This tag is used to set icons, title, to provide some meta data (info about web app), css settings, java scripting etc...

head tag contains some child/sub tags, those are

<link>, <title>, <meta>, <style>, <script> and <base> tags

body tag

body tag represents content information (means output) of the web page.

this information appears on the web page/in the browser (it's called content).

This tag is used to design UI or to display output.

body tag contains so many child/sub tags.

some of tags: **p**, **img, h1, table, div, a, table, audio, video, input, button, form, ol, ul, li, hr tags …**

html is a collection of tags(elements) and every tag has some attributes.

how design & execute html programs

⇒open any text editor (sw) and type program.

notepad, editplus, notepad++, textpad, sublime, vs code, atom, coffee, ...

⇒save that program with any name (.html or .htm) and anywhere in the system.

⇒execution:

1st Approach: goto file location, then double click on file

2nd Approach: goto file location, then right click on file and click on open then select browser

3rd Approach: open any browser, then goto address bar and type filename with address.

for ex: d:/siva/test.html e:/test.html

comment lines

⇒ Comment lines are to provide some description about our program.

⇒Comments are not executed by browser.

Syn:

<!-- text -->

heading tags

These tags are used to print data/text in heading format.

html provides 6 heading tags, those are h1, h2, h3, h4, h5, h6.

These 6 tags are used to display headings in different sizes.

six tags are paired tags and block level elements.

Syn:

<h1> text </h1>

<h2> text </h2>

<h3> text </h3>

…

Note: inside the body section we can repeat any tag and no.of times.

p tag

> p stands for paragraph.

> this tag is used to display/print more lines of text (paragraph)

> its paired tag and block level.

> browser display an empty line(gap) between paragraphs

Syn:

** text or info **

Note:

>browser/html doesn't accept more than one space (space bar & tab key), means while designing the program we are given more space but browser prints only one space.

>browser/html doesn't accept enter key (line breaking), means while designing a program we use enter key but browser prints data without breaking line.

br tag

- Ø br stands for break line (enter key)
- Ø it moves the cursor to the beginning of the next line.
- Ø its un-paired

Syn: **
** or **
**

Html entities

=> Entities nothing but Special characters or html operators

=> Every operator is used to perform some task or to print some special text.

=> Operator is a English word

Syn: **&**operator**;**

Html hexa-decimal operators, these operators are starts with #

Hexa-dec base 16 ➔ 0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f

Syn: **&#**operator**;**

b tag or **strong**

> b stands for bold

> b & strong tags used to print text in bold format

> both are paired tags & inline tags

Syn:

 text

 text

I or **em** tag

>i stand for italic (inclined)

>i & em tags used to print text in italic format

>i is paired

Syn:

 text

 text

u tag

> u stands for underline

> u tag used to print text with underline (draws a line base of text)

> u is paired tag

Syn:

<u> text </u>

strikeout tag

> strikeout tag used to print text with line (draws a line middle of text)

>strikeout is paired tag

Syn:

<strike> text </strike>

superscript tag

>this tag used to display text top of upper line

> superscript is paired tag

Syn:

^{text}

subscript tag

>this tag used to display text bottom of baseline

> subscript is paired tag

Syn:

_{text}

All these tags are paired tags & inline tags

Span tag

>span tag used for small textual data, like as error message, mandatory specification.
> in continuity of text, if we want to highlight couple of word or **letters**, we use span tag
>its paired tag, inline tag
Syn: **** text ****

pre tag

> pre stands for pre-formatting (alignment)
> pre tag is used to print data/text, how we typed in same format
> pre is paired tag, block level
Syn:

<pre> text </pre>

Label tag

> label tag used for displaying prompting text.
> its paired tag, inline tag
Syn: <label> text </label>

HTML Attributes

⇒Attribute is a special feature/setting of a tag.

⇒Every tag they have attributes

⇒An HTML attribute is a piece of markup language used to adjust the behavior or display of an HTML element. For example, attributes can be used to change the color, size, or functionality of HTML elements.

⇒HTML Attribute is something that we use in the starting tag of HTML Elements or HTML Tags which provides extra information about those HTML Elements or HTML Tags.

Syn:

<tagname attribute="value" attribute='value' ...>

Note:

⇒Parameters should be enclosed within " " or ' ' or without quotes.

⇒Every attribute must be separated by a space

Types of attributes

⇒general attributes

These attributes are common for most tags (99% of tags). These attributes are used to adjust the behavior or display of an HTML element, to provide extra information about those HTML Elements to the browser.

those attributes are:

class, id, name, style, align, action, method, href, src, target, width, height, alt, title, lang, min, max, step, maxlength, type, checked, selected, value, readonly, placeholder etc…

⇒event attributes

An event is a notification that is triggered when something changes in the browser.

With event attributes these events are directed to JavaScript which then responds to the event.

These attributes are used to perform some logical operations.

logical operations we can perform by using JavaScript, these also called dynamic attributes.

By using event attributes From Html page we can tigger JavaScript or we can call JavaScript

attributes are:

onclick, **on**dblclick, **on**focus, **on**blur, **on**keypress, onkeyup, onkeydown, onsubmit, onchange, oninput, onreset, onselect, **on**mousemove, **on**mouseout, **on**mouseover, onwheel, **on**load, **on**submit, **on**change etc…

title tag

this tag used to set the title for a webpage, means every webpage
they have individual title.
its paired tag.
<title> is the sub tag of <head> tag.
Web site => 10 web pages =>Title 10 times

Syn:

<title> text **</title>**

Note: one web page/one title

Link tag

Link tag used to set the icon/logo for a webpage.
Un-paired tag.
<link> is the sub tag of <head> tag.

Syn: <link rel="icon" href="filename"/>

Relative => icon or stylesheet

Hyper reference =>.jpg .bmp **.png** **.jfif** .gif .tif **.ico**

images

> "img" tag is used to display images on webpage.
> in one webpages we can display any no.of images and any type of images.

> it is strongly recommended to place all images in side root folder or create sub folder with name images in root folder
>its un-paired tag, and its inline element
Syn:

<img attributes**/>**

.jfif .svg .jpg .bmp .gif .tif .png .webp

attributes:
src => to specify which img you want to display
width => width of image (pixel)
height => height of image (pixel)
title => it is used to specify tool tip. (whenever mouse pointer comes on top of image)
alt => alternative text, if image not loaded in webpage/not display, we want to display text message to user it called as alt

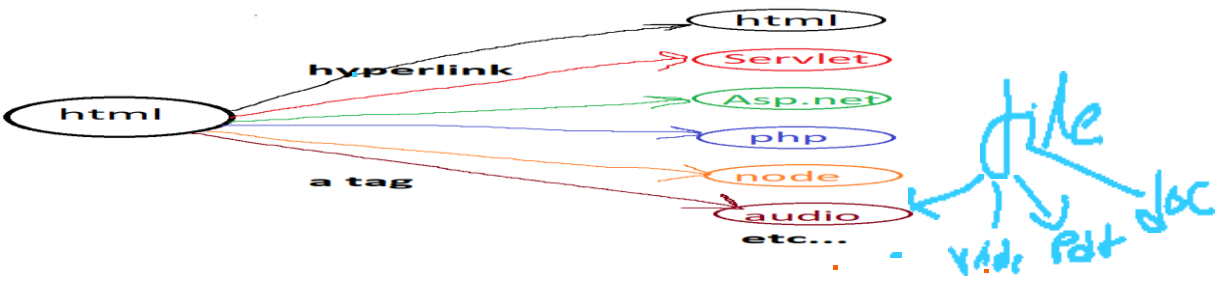
+

global attributes

opacity: 0.5;
filter: blur(5px);
brightness(125%)
contrast(135%)
grayscale(100%)
invert(100%)
hue-rotate(180deg)
saturate(8)
sepia(100%)
drop-shadow(8px 8px 10px green)

hyperlinks

> a stand for "anchor"
>"a" tag is used to create hyperlink, hyperlinks are used to move from one webpage to another webpage.
>whenever user clicks on the hyperlink, it moves to the specified page.
> destination page sometime within same application or other application.



>web application basically contains links to other pages, so it's very commonly used tag.

> by default, every browser provides built-in style for each hyperlink, i.e blue color+handsymbol+under line.

we can customize these styles by using CSS.

> its paired tag, and inline element

Syn:

<a attributes>Display Text

<a attributes> <img>

attributes:

href : hyper reference, used to specify the address of webpage or web site,

i.e whenever user clicks on this link, which page you want to open

url may be html page, server-side file, image, audio file, video, pdf file, documents etc...

href="url"

"https://www.abc.com/login.aspx"

"" ➔self-calling

"." ➔ home page of web site/home dir of web application

"#id" ➔ it creates book marks (moving within same page)

target : where you want open destination page

_blank ==> opens the link in a window/tab

_self ==> opens the link in current working tab/window (its default)

_parent ==> opens the link in parent frame

_top ==> opens the link in full body of window

framename ==> opens the link in specified frame

CSS

- ⇨ Cascading style sheets
- ⇨ Released & maintained by W3C org
- ⇨ Used to change look/feel of html elements (makeover)
- ⇨ Css provide only styles but not tags
- ⇨ Style is group of properties (attributes)

Where we can define styles?
We can define styles in places, thoese are:

- ➢ Inline styles
- ➢ Internal styles
- ➢ External styles

Different ways to implement css:

1st Approch (inline):

Html tag and css properties are defined Within the same line
<tag Style="property:value; property:value; …">

2nd Approch (internal):

Html tags and css styles are designed in the same program, but not in same line.

Internal css should be implements in Style tag, style tag must be sub tag head tag.

```
<style>
  selector{
        property:value;
        property:value;
        …
   }
   Selector{
        property:value;
        property:value;
        …
   }
   Etc…
</style>
```

3rdApproch (external)

Css styles are designed in separate file and should be save with ".css", and html code saved with separate file ".html"
Use link tag for mapping css file to html file
Syn: **<link rel="stylesheet" href="filename.css"/>**

note:

- css attributes we can't use in place of html attributes.
- html attributes we can't in place of css attributes.

html colors

html supports 3types of patterns, those are
> named colors
> RGB colors
> Hexadecimal colors

named colors:
>it supports to write direct color name
>we have some limited colors
ex: white, black, red, green etc...
> Color names are not case-sen

RGB colors:
>RGB model specifies that the composition of 3 basic colors (Red, Green, Blue)
>RGB produces 16millions colors.

Syn: **rgb**(red,green,blue)
red => 0 - 255
green => 0 - 255
blue => 0 – 255
ex: **rgb**(10, 45, 201) 401%255 ➔146

Hexadecimal number colors:
>Hexadecimal model is the shortcut for rgb model
>Hexadecimal system ranges from 0 - 15
0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f
Syn: **#**RRGGBB 1,2 red 3,4 green 5,6 blue
ex: #1a4b68

#RGB

ex: #3d7

Note: in realtime "Hexadecimal model" is recommended.

these colors we can use for foreground color, background color, border color etc..

for setting colors we have some attributes, those are

color ➔ to set/to change foreground color (text color)

background-color ➔ to set/to change background color

border-color ➔ to set/to change border color (line color)

box-shadow ➔ to set/to change shadow color

text-shadow ➔ to set/to change text shadow color

Note: all these are CSS attributes. Support by Most of html tags

Gradient colors

background: #FC466B; /* fallback for old browsers */

background: -webkit-linear-gradient(to bottom, #3F5EFB, #FC466B); ← Chrome 10-25, Safari 5.1-6

background: **linear-gradient**(to bottom, #3F5EFB, #FC466B); ←W3C, IE 10+/ Edge, Firefox 16+, Chrome 26+, Opera 12+, Safari 7+

linear-gradient(direction, color1,color2,…color-n)

dir: to left (r=>l)

to right (l=>r)

to top (b=>t)

to bottom (t=>b)

background: **linear-gradient**(to bottom, #3F5EFB 40%, #FC466B 60%);

-webkit-linear-gradient(to left, #3F5EFB, #FC466B);

linear-gradient(to left, #3F5EFB, #FC466B);

background: radial-gradient(circle, rgba(2,0,36,1) 0%, rgba(38,38,162,1) 60%, rgba(0,212,255,1) 100%);

radial-gradient(shape, color1, color2, …color-n)

radial-gradient(circle, rgb(131,58,180) 0%, rgb(29,166,65) 50%, rgb(252,176,69) 100%);

radial-gradient(circle, rgba(166,29,142,1) 57%, rgba(100,180,111,1) 78%, rgba(69,252,96,1) 100%);
Note: while applying gradient colors we have to use "**background**" property in place of "**background-color**".

working with list tags

these tags are used to display data/info in points wise.
html supports three types of list, those are

Ordered list➔ numbering
Unorderedlist ➔ bulleting

ol tag

>ol stands for "Ordered List".
>it is used to display the text(names, colors, team names, course name...) with numbering.
>it supports 5types numbering, those are **1, A, a, i, I**. by default it displaying in number.
>by using "ol" tag we can create ordered list
>ol is paired tag & block level element

li tag

> li stands for "list item"
> li is sub tag of ol tag
> li tag is used to print text/data in points wise
> li is paired tag & block level element
Syn:

```
<ol attributes>
        <li> text </li>
        <li> text </li>
        <li> text </li>
    ...
</ol>
```

ol attributes:
type : which type numbering to display (Default is 1)
start : from where u want to start numbering (default is 1)

reversed : to displaying numbers in desc order

li attributes:
value : used for restarting numbering with specified value

ul tag

>ul stands for "Un-Ordered List".
>it is used to display the list of items(names, colors, team names, course name...) with bulleting.
>it supports 3types bulleting, those are **dot, circle, square**. by default, is dot.
>by using "ul" tag we can create un-ordered list items
> ul is paired tag
>"li" tag used for creating list items
Syn:

```
<ul type="dot/circle/square">
<li> text </li>
<li> text </li>
<li> text </li>
   ...
</ul>
```

dl tag

>dl stands for Definition list (since html5 description list)
>dl tag used for to display definitions/full forms (collection of definitions)
>its paired tag
> "dt" and "dd" are sub tags of "dl" tag
> "dt" stands for definition title, "dd" stands for definition data.
> dt & dd are paired
Syn:

```
<dl>
<dt>title/word</dt>
<dd>information</dd>
<dt>title/word</dt>
<dd>information</dd>
<dt>title/word</dt>
<dd>information</dd>
   ...
</dl>
```

fieldset tag

> this tag used for drawing a common border around elements/tags.
> its paired tag and block level
> we can draw any no.of borders

Syn: <**fieldset** attributes>
<**legend**>text</**legend**>
Sub elements
</**fieldtset**>

attributes:

align : align of elements, it supports 3 alignments center, left, right
left is default align
border : style of line, thickness of line, color of line
width : width of box (size in %)

legend tag

>legend tag used for set title/heading for fieldset
>legend is sub tag of fieldset tag
>its paired tag

Syn:<legend attributes>Heading</legend>

attributes:

align :align of elements, it supports 3 alignments center, left, right
left is default align
color :

div tag

>div is a container, means its grouping elements/controls/components of html.
>inside div tag we can place any content like normal text or images.
>div tag is used to divide web page as no.of subpages/parts, each part is rep as div.
>for better maintained, effective design of web page and simplifying css code.
>its paired tag, and block level element

Syn: <div attributes>
contents

>one webpage may contains any no.of div tags.

display:flex; <== it displaying all elements side-by-side row wise or column wise
flex-wrap:wrap; <== it align element to next line
flex-direction <== it used to specifiy direction (order) of flex elements
flex-direction:row|row-reverse|column|cloumn-reverse;
flex-flow <== it combination of felx-wrap & flex-direction attributes.
flex-flow: direction wrap;

display:grid; <== it displaying all elements in rowsXcols
grid-template-columns <== no.of columns to display (width of columns)
grid-template-columns:col1 col2 col3....;

:autoautoautoauto; <== 4columns

:300px 400px 250px; <== 3columns

:30% 30%; <== 2columns

:30% auto 400px;
grid-column-gap: Npx; <== it provides a gap between column to column
grid-row-gap: Npx; <== it provides a gap between row to row
grid-gap:Xpx; <== it provides a gap between row-row & col-col with same size

Note: its applicable on nested tags, means outer tag only we can apply grid

table tag

>table tag is used to display the data in form rows & cols in the web page.
> a table is a collection of rows, each row is collection of cells/col/field.

> a table is represented as <table> tag, a row represented as <tr> tag, a colheading is represented as <th> tag, data rep as <td> tag.
> table heading is represented as <caption> tag.
><thead> tag is rep of table head part, <tbody> tag is rep of table bodypart and <tfoot> tag Is rep of table footer part.

table➔ it just comb rows & cols
caption➔ main heading of table
tr➔ table row, used to draw a row
th➔ table heading (col heading)
td➔ table data (col data)
thead➔ table head section
tbody➔ table body section
tfoot➔ table footer section

> all these 8tags are paired tags
table, tr, caption, thead, tbody & tfoot are block level tags
th & td are inline tags

Syn:

```
<table>
    <tr>
        <th>heading</th> <th>heading</th>
    </tr>
    <tr>
        <td>data</td> <td>data</td>
    </tr>
  ...
</table>
```

NOte:
<th> and <td> are sub tags of <tr>
<tr> is sub tag of <table>

table attributes:
border : border of table (0 means no border, 1-n border req)
align : alignment of table
width : width of table (%)
...
th& td attributes:

colspan : specifies the no.of columns to merge/expend
rowspan : specifies the no.of rows to merge/expend
...

form elements

input tag

> input tag is used create a form elements

> these elements are used to take input/accept data from users.

> controls nothing but text field, button, radio button, file, color, password, etc...

> input is un-paired tag and inline element

Syn:

<input attributes/>

Note: by default, input tag creates “text field”.

attributes of input tag:

type ➔ it reps which type of control you want to create.

if we are not specified by default it creates a textbox.

input type="text|hidden|password|number|button|reset|submit|image|checkbox|radio|file|color|email|url|range|date|time|datetime-local|month|week..."

name ➔ it represents name of control, used in server-side programming (not unique)

id ➔ it represents id of control, used in client-side programming (unique)

value ➔ it used to initialize input elements (default value)

readonly ➔ this attribute not allowed to change the value of control

size ➔ this attribute specifies size of control (width of control)

disabled ➔ this attribute disabled the control

placeholder ➔ this attribute used to display prompting text within the element

maxlength ➔ this attribute used to set max limit of data (no.of chars)

required ➔ this attribute used to force the user to enter data (mandatory field)

autofocus ➔ this attribute set's cursor position (Default location)

tabindex ➔ this attribute controlling cursor movements (when we r pressing tab key, where cursor is moving)

min ➔ this attribute used to set minimum limit of number value (lower bound)

max ➔ this attribute used to set maximum limit of number value (upper bound)

step ➔ this attribute used to specify incremental & decrement value of number

text fields

>text field used for taking input(any type of data) from user or text box allows us to type data.

>text fields are used for typing of username, first name, pincode, phone, address etc...

>by using "**input**" tag we can create text fields

>user can type any no.of char's but single line, if u want to set a limit use "maxlength" attribute.

>text field allows you to type data in 1 line.

Syn:

<input type="text" attributes/>

Note:

id ➔ used by client-side programming (javascript)

=> unique (duplicate id are not)

name ➔ used by server-side programming (servlet/asp.net/php...)

=> either unique or not

hidden field

>an in-visible text field with default value is called a "hidden field".

>hidden fields are used to send some information about user/client to a server-side program without asking the user.

>to perform session tracking, we are using hidden fields.

>like normal text fields, hidden fields data is also sent to the server when we click the submit button.

Syn:

<input type="hidden" name="NAME" value="VAL" attributes>

password control

> password is a text field but the data is not visible.

> it's used for accepting password, pin, cvv, otp, verification code, passcode... from the user.

> password is displayed as * or dot

Syn:

<input type="password" attributes>

number control

>this control used to accept only numerical value (numbers only) from user

>it prevents typing of alphabets, special chars (in some browsers), except **minus**, **dot** and **e** alphabet.

>but some browser allows to type all these data but given error while submitting form

> some browsers displaying number fields with increment/decrement buttons

SYn:

<input type="number" attributes>

attributes:

min =>it rep starting value of number field (by default no min)

max =>it rep ending value of number field (by default no max)

step =>it rep increment/decrement value (by default +1/-1)

button control

reset button

>reset is one type of button, used for reset the html form/UI, meaning it clears all values of form.

>reset button must be part of the "form" tag.

> these button values are not sent to the server.

Syn:

<input type="reset">

submit button

>submit is one type of button, used to submit/send the html forms/page/UI to a server-side program.

>while submitting the html form, all input parameters (enter by user), hidden parameter (by programmer) are prepared as a "Query String"

ex:

https://www.redbus.in/search.aspx ? param-name=value&param-name=value &...

param-name => name of text box or name of password field or name of button...

value => value of text box or value of password field or value of button...

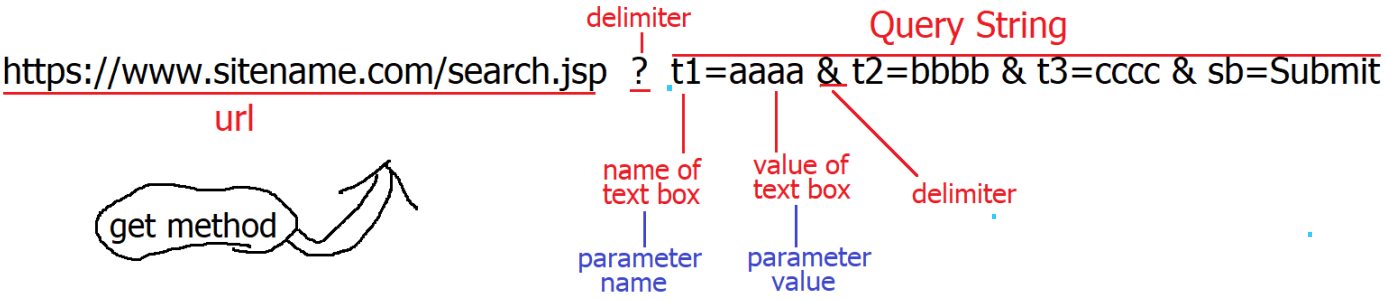
>outside the form, if we design anything that is not submitted to the server and without name attribute also not submittable.

>submit button value also sent to server

Syn:

<input type="submit" value="VAL" attributes>

<form action=”search.jsp”>



public String getParameter(String param)

String st = request.getParameter("t1")

post

https://www.sitename.com/search.jsp

image button

> "image submit button" is used to submit a form to the server.

> when the user clicks on the image button, browser is submitting data with x-co & y-co of image button

Syn:

<input type="image" src="filename" .../>

working with html forms

application forms => white paper

> form is collection/group of html input elements

> by using <form> tag we can create html form(s) (mean application form)

> web document/page it contains only one body, but a body can contain multiple forms.

> forms are used to collect info from users(ex: signup page, singing page, user registration, product delivery info etc...)

info we collected by using some fields (created by tag) like text field, password, checkbox, radio button, combo-box, list box, date, submit button, etc...

> form is used to submit input values (user's data) to a server-side program.

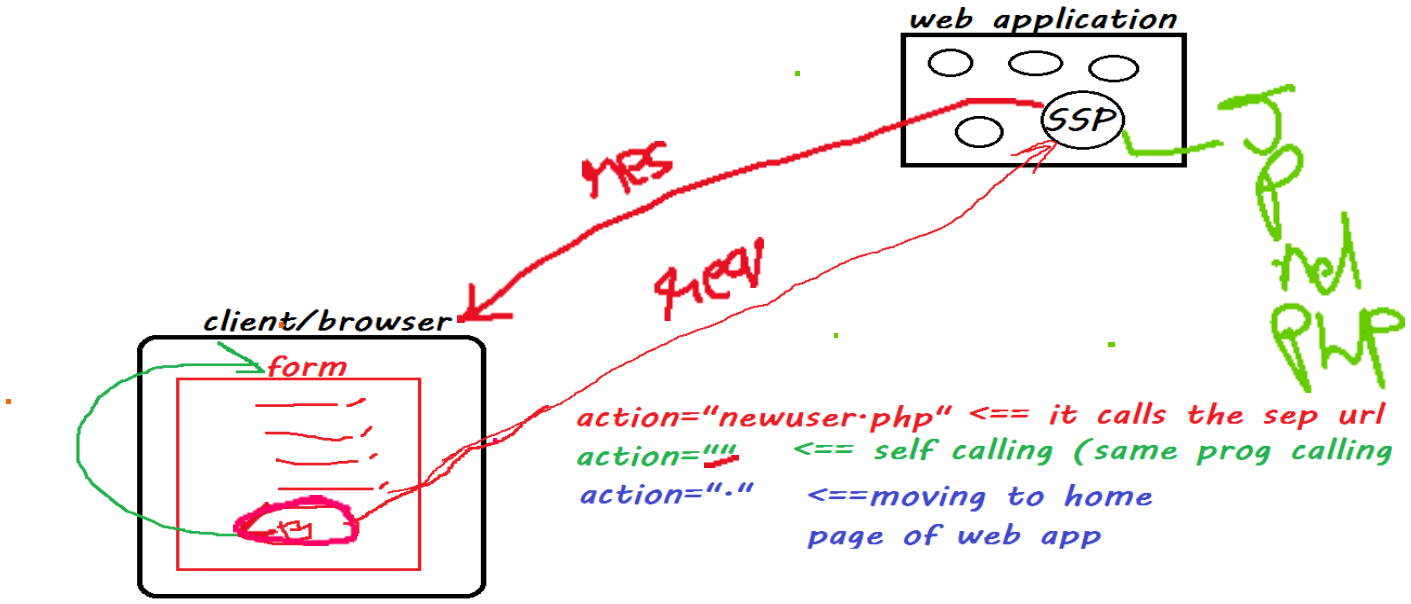
> form is a paired tag & block level

> form tag is sub tag of body tag

Syn: <form attributes>

UI designing

</form>



attributes:

1 action :url represents destination program address or

which program we want to call specify here

diff forms of urls:

https://www.irctc.co.in

https://www.sitename.com/login.class|login.aspx|login.php|login.ns|login.cgi|login.py

"" self-calling (its calling/sending data to current prog)

"." home of current application

2 method : it represents the way of sending data from client to server

it supports two ways, those are

https://www.redbus.in/search

get:

> it displaying data in address bar

https://www.redbus.in/search?fromCityName=Hyderabad&toCityName=Guntur&onward=10-Jan-2022&opId=0&busType=Any

> get method stores user inputs in browser history

> it's less secured

> get is a default method

> we can bookmark these pages

> get methods max data limit is 5.7kb

> get is faster than post

use-case's → search page, retrieving data from db, ...

Post:

> it not displaying any data in address bar

https://www.redbus.in/search

https://www.irctc.co.in/nget/booking/train-list

> post method doesn’t store user inputs in browser history

> it's more secured

> we can't bookmark these pages

> post method data no limit

> post method slower than the get

use-case's □ login page, sign-up page, registration pages, ...

3 target : its rep where to open destination page,

_self, _blank, parent, framename ..

default is _self

4 enctype : it rep in which format we are sending data to server

html support two types, those are

>application/x-www-form-urlencoded

If you want to send data to the server without attachment and file uploading use this method.

It is default option

>multipart/form-data

if u want send data to server with attachment and file uploading

5 autocomplete : it automatically saves data while typing in the UI (textbox, password, address, pincode, ...)

on/off

on is default

6 novalidate : while submitting html perform some basic validation, if u don't do those validations, switch off this.

check box

>check boxes are used to allow the user to select some options, for example product, class, color selection, sport selection, select branch, select collage etc.

>whenever we want to select more than one option use check box's

>if the checkbox is selected/checked it returns "on'' (true) value, if the checkbox is unchecked it returns "off" (false) value.

checked➔ on (SSP), true (CSP)
un checked ➔ off (SSP), false(CSP)

100cb => same purpose => 100cb name should be same

100cb =>20cb =>purpose games => 20 name should be games

30cb =>purpose sports => 30cb name is sports

50cb => name3

SYn:

<input type="checkbox" attributes>

Note: all check boxes should be created with the same name.

"checked" attribute of the check box makes the checkbox by default checked, while opening the page.

Syn: checked="checked" before html5 ver

checked since html5 ver

radio button

> RB is used to display two or more options to the user, but allows the user to select any one of them.

SYn:

<input type="radio" attributes>

Note:

> all RBs should be created with the same name.

> "checked" attribute of RB makes the RB by default selected, while opening the page.

file

>file used to upload or attachment

>we can upload or attach any type of file, but @time a only one file

Syn <input type="file" attributes>

attributes:

  multiple => it allows multiple to upload @time

  accept => filtering type of file

Note:

while uploading file method should be "post" and enctype is "multipart/form-data"

color

>used to select color by user, selected color we can apply on any control using JS

Syn: <input type="color">

date & time controls

date

>used to create a date box (date picker/popup calendar), where the user can select a date.

>the browser by default provides a built-in date picker.

Syn: <input type="date" attributes>

time

>used to create a time box, where the user can enter/select time (in the form of hours, minutes and seconds)

Syn:<input type="time" attributes>

datetime

>used to create a date-cum-time control

Syn: <input type="datetime-local" attributes>

month

>used to create a month box, where the user can select a month.

Syn: <input type="month" ...>

week

>used to create a week box, where the user can select a week.

Syn: <input type="week" ...>

email

>used to create an email textbox, where the user can enter a valid email id only.

>it displays an error message automatically (built-in validation), if the user enters other than email id (should contain @ and .).

Syn: <input type="email" attributes>

url

>used to create an url box, where the user can enter a valid url for downloading files or playing videos.

>it displays an error message automatically (built-in validation), if the user enters other than url.

Syn: <input type="url" attributes>

range

>used to create a slider bar, based on the specific range.

>this control req min value and max value, if we are not specified then browser takes default values.

Syn: <input type="range" attributes>

attributes

min => it sets min value of slider

max => it sets max value of slider

step => it sets increment value

value => sets indicator init position

appearance: slider-vertical;

search

> used to create a search box, where the user can enter some search text, it also displays a clear button to clear the text inside the search box.

Syn: <input type="search" ... >

dropdown (combo box) control

>using the "select" tag we can create a dropdown list/CB.

>dropdown list is used to display some options/items to the user and allows the user to select any one of them.

>"option" sub tag of "select" tag.

> "option" tag used to create/to add items/options to dropdown list.

> both are paired tags.

> select is inline element and option is block level element

Syn:<select attributes>

<option attributes> Text </option>

<option> Text </option>

...

</select>

new Syn:

<option> Text </option>

<option value="ws" > text </option>

Note: by default, DDB is displaying 1st added option/tiem

attributes:

selected : this attribute of "option" tag, used to change default selected option/item of DDB

list box

>using the "select" tag we can create a list box.

>list box is used to display some options/items to the user and allows the user to select any one of them (by default).

> "option" sub tag of "select" tag.

> "option" tag used to create/to add items/options to the list box.

> "size" attribute used to change DDB into List box.

Syn:<select size="N" attributes>

<option attributes> Text </option>

<option> Text </option>

<option> Text </option>

...

</select>

Note: by default DDB is displaying 1st added option

attributes:

selected : this attribute of "option" tag, used to change default selected option/item of DDB

multiple :it allows user to select more than one option @time

size :attribute used to change DDB into List box and no.of options to display @time

option groups

> "optgroup" tag is used to group-up some options/items inside the "select" tag.

> one "select" tag can contain many "optgroup" tags, the "optgroup" tag contains many "options".

>its paired tag

> "optgroup" tag is the sub tag of "select" tag.

Syn:

```
<select>
    <optgroup label="Text">
    <option> ..</option>
    <option> ..</option>
    <option> ..</option>
</optgroup>
<optgroup>
        ...
</optgroup>
</select>
```

textarea

>"textarea" tag is used to create a multi-line textbox.

use case: comments, address, feedback, delivery instr, ...

>its paired tag& inline tag

>it creates a multi-line text box, with default sizes (2row & 20 col)

Syn: <textarea attributes>

init value

</textarea>

attributes:

rows : it rep no.of lines to display @time, if more lines of data entered

automatically scrollbar is activated.

cols : it rep no.of chars per line

maxlength : it rep total no.of chars allowed in textbox.

Note: user can resize the textarea, at runtime in the browser.

progress bar

> "progress" tag is used to display the progress of a task.

> to move progress bar dynamically, we have to use "JS"

> it is a paired tag.

SYn: <progress attributes></progress>

attributes:

min max value

label

> label tag is used to create heading/prompting messages for elements or controls.

> label providing description for controls, it gives an idea to user what we have to type.

> when the user clicks on the label, the cursor will appear in the associated control automatically.

>its paired tag.

SYn: <label attributes> text </label>

attributes

for : used to specify the id of the control that is associated with the control

Note: labels are not sent to the server while submitting the form.

HTML5 tags

Datalist

Syn:<datalist attributes>

<option>text</option>

OR

<option value=”text”>

….

</datalist>

Mapping: **<**input type=”text” list=”datalist-id”/>

output tag

> this tag used to print data/output value on webpage.

> it is paired & inline element.

Syn <output attributes> text </output>

details and summary

>details and summary tags are used to allow the user to expand/collapse some information, when the user clicks on the heading.

>both are paired& block level

> details tag is the main tag and summary is the sub tag of details tag.

Syn:

```
<details>
    <summary>Short info</summary>
  Detailed Information
</details>
```

header

>"header" tag represents header bar, which may include website logo, search box, main links, etc...

>it doesn’t provide any styles by default; we have to apply styles manually, using CSS.

>its paired tag.

Syn: <header>

place header content here

</header>

nav

>"nav" tag represents navigation bar, which may include top navigation menus.

>it doesn’t provide any styles by default; we have to apply styles manually, using CSS.

>its paired tag.

Syn: <nav>

place menus/links here

</nav>

section

>"section" tag represents a specific section of the page(box or container), which may include main-content/information.

>it doesn’t provide any styles by default; we have to apply styles manually, using CSS.

>its paired tag.

Syn: <section>

place main content here

</section>

footer

>"footer" tag represents the footer part of the web page, which may include information of contact, faqs, location, copyrights, etc...

>it doesn’t provide any styles by default; we have to apply styles manually, using CSS.

>its paired tag.

Syn: <footer>

place footer content here

</footer>

aside

>"aside" tag represents the "right-side" part of the web page, which may contain ads/other promotional information.

>it doesn’t provide any styles by default; we have to apply styles manually, using CSS.

>its paired tag.

Syn: <aside>

place ads/extra info here

</aside>

article

element specifies independent, self-contained content.

- Newspaper articles
- Forum posts
- Blog posts
- User comments

Syn: <article>

place ads/extra info here

</article>

figure

The <figure> element is used for indicating self-contained content. The tag can include images, diagrams, illustrations, code examples, etc.

figcaption

The <figcaption> element is used for adding signature or annotation to the <figure> tag.

Syn: <figure>

Image

<figcaption> caption/text </figcaption>

</figure>

mark

The <mark> element is used to mark a part of the text which has relevance. It can be used to highlight a text for showing emphasis, highlight search terms in search results to provide context, or distinguish a new content added by the user by showing it differently

Paired tag & inline

Syn: <mark> text </mark>

bdi

The <bdi> element is used to isolate bidirectional text when a language with a right-to-left directionality, such as Arabic or Hebrew, is used inline with left-to-right languages

Paired tag & inline

Syn: <bdi> text </bdi>

base tag

this tag is used to specify a base URI or URL to use for all relative links contained within an HTML document.

Only one “base” element can be specified within a document, and it must be placed within “head” element.

We can also specify how other links should open using the target attribute.

Its unpaired tag

Syn: <base href=”base-url” target=”target”/>

Audio tag

The <audio> is one of the HTML5 elements added to allow embedding audio files to a web page. Since not all browsers support all audio formats, the audio file is encoded using special codecs.

The **<source>** tag or the src attribute is used to indicate the variations of the same audio file. The path to an audio file can contain absolute or relative URLs.

Syn:-

<audio attributes></audio>

OR

<audio attributes>

<source src="filename" type="audio/type">

</audio>

Attribute Definition

src URL => Specifies the path to the audio file.

controls Displays the control panel (start button, scroll, volume control).

If the controls attribute is missing, the audio file will not be

displayed on the page.

autoplay Plays the audio file automatically after loading the page.

loop Repeat the audio file from the beginning after its completion.

muted Mutes the sound when the audio file is played.

Video tag

The <video> is one of the HTML5 elements added to allow embedding video files to a web page. Since not all browsers support all audio formats, the audio file is encoded using special codecs.

The <source> tag or the src attribute is used to indicate the variations of the same audio file. The path to an audio file can contain absolute or relative URLs.

Syn:-

```
<video attributes></video>
```

OR

```
<video attributes>
    <source src="filename" type="video/mp4">
    ...
```

</video>

Attribute Definition

src URL => Specifies the path to the video file.

controls Displays the control panel (start button, scroll, volume control).

If the controls attribute is missing, the video file will not be displayed on the page.

autoplay Plays the audio file automatically after loading the page.

loop Repeats continuously the audio file from the beginning after its completion.

muted Mutes the sound when the audio file is played.

width width of video player

height height of video player

poster to set wall poster/paper of video

frameset & frame

iframe

> "iframe" stand for inline-frame, html5 rel tag

> placing one webpage result within another webpage.

> its paired tag & inline tag

Syn:

<iframe attributes>

</iframe>

att:

src => which page we want to include

srcdoc => it displaying text

width =>

height =>

frameborder => disable/enable border

map tag

> this is used to map an img to other images or webpages.

> map tag used to explore internal details about par area of img

> area is sub tag map tag

> area tag used to create mapping co-ordi

> map tag should link img tag ny using "usemap" attribute of img

> map is paired and area is non-paired

Syn:

```
<imgsrc="image">
<map name="">
    <area shape="" href="" coords="" alt="">
    <area shape="" href="" coords="" alt="">
    ...
</map>
```

Shape: it rep mapping area shape.

it sup 3 shapes, those are

> circle

>rect

> poly

Hrefits destination url or image

coords shape x and y co-ords

https://www.image-map.net/

marquee tag

> used to move text/element in different directions

> its deprecated tag.

> its paired tag

Syn:

<marquee attributes>text|img| component </marquee>

attributes:

direction => down, up, left (default), right

loop => continues moving (default)

scrollamount => 6

scrolldelay => 84ms

behavior => alternate

meta

> data about data means we provide some info/details about a webpage.

> meta is unpaired

> meta is subtag of <head> tag

> by using meta we provide info to browser, search engines, users/programmers.

Define keywords for search engines:

```
<meta name="keywords" content="HTML, CSS, JavaScript">
```

Define a description of your web page:

<meta name="description" content="HTML and CSStutotrial">

Define the author of a page:

<meta name="author" content="SK">

Refresh document:

<meta name="refresh" content="45">

<meta http-equiv="refresh" content="time; url=URL"**>**

Setting the viewport:

<meta name="viewport" content="width=device-width, initial-scale=1.0">