

Language

It is a tool to communicate
Programming Language :-

It is a tool to communicate machine
It has three types.

- Low level understandable by machines only
- Assembly language / sorting of some keyword which given to assemble).
- High Level language / understandable by machine and developer).

Software

A Set of instruction to do a specific part. It has two types.

- System Software.
- Application Software.

System Software :-

The System SW is a SW used to control the whole device.

Application Software :-

The Application SW is a SW used to do a specific part.

Technology :-

Tools and Techniques in web

Internet :-

⇒ Internet is a interconnection of networks through worldwide to share data.

⇒ Internet follows packet Routing Network.

Packet :-

→ The messages send through Internet are converted into tiny packets.

⇒ Called packets.

Routing :-

⇒ It establishes connection to different networks.

Networks

⇒ It is a group of devices.

The Packet Routing network has two protocols.

⇒ Internet protocol address

⇒ Transfer Control protocol.

Client and Server Architecture :-

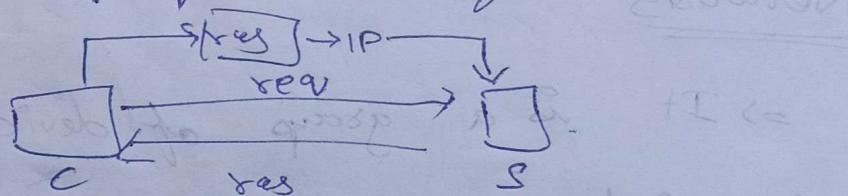
* If a client Requests of it will now go to Server and receive the response and display the result.

* The client has http and URL as medium to request

HTTP \Rightarrow HyperText Transfer Protocol

URL \Rightarrow Uniform Resource Locator

* The clients request was sent to Domain name Servers (DNS) and it creates IP Address and with the help of IP address it will reach the Server and Server will respond to the client with requested Information.



Browser:-

\rightarrow It is an Application that helps clients to interact with Server

Web:- \rightarrow It is a client side application

\rightarrow It is a area of collection of Information.

Website

A Collection of web Application and web pages

Web Application => A Collection of web pages

web pages => Is a single HTML Document

NOTE :- i) Internet and web are not same
ii) Internet is a Subset of web.

INTRODUCTION

HTML => TIM BERNERS - LEE 1993.

* Create web pages and Application.

* HTML 5 package to build. : websites and. Handle Presentation.

* HTML Stands for Hyper Text Markup Language
Hyper Links → Reference Link.

Text → Text in web pages.

Markup → Presentation of text

Language → Browser Language.

DECLARATION OF HTML

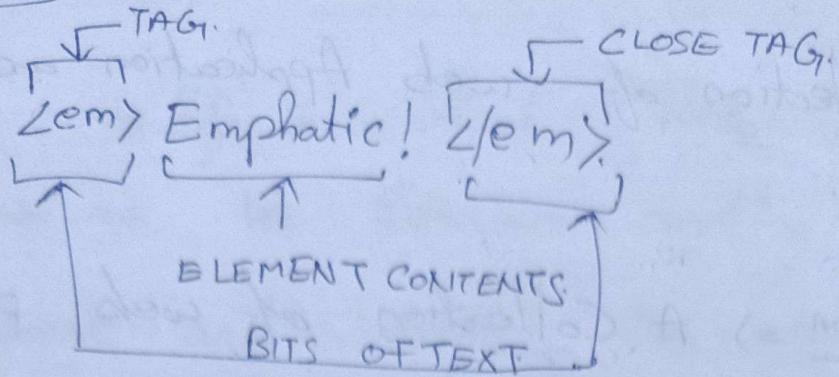
<!DOCTYPE html> refers to the version of the HTML.

<HTML> → Root of the HTML file.

<head> → It contains meta data.

<body> → It contains information for webpage.

TAGS



- * Tags will have some information.
- * Anything which is enclosed inside angular bracket is tag.
- * Paired tag or Container tag.
e.g.: `<html></html>`, `<head></head>`
- * Unpaired tag or Self closing or Empty tag.
e.g.: `
`, ``, `<bs>`

HEADING TAGS

Use HTML Heading tags only for headings.

`<H1>` Heading Tag `</H1>` maximum font size
`<H2>` " `</H2>`
`<H3>` " `</H3>`
`<H4>` " `</H4>`
`<H5>` " `</H5>`
`<H6>` " `</H6>` minimum font size.

TEXT FORMATTING TAGS

Formatting the text in web pages

Tags

Tag	Defines	Single page app
	Defines bold text	
<big>	Defines big text	
	Defines emphasized text	
<i>	Defines italic text	
<small>	Defines small text	
	Defines strong text	
<sub>	Defines subscripted text	
<sup>	Defines superscripted text	
<ins>	Defines inserted text	
	Defines deleted text	
<s>	Deprecated. Use instead.	
<strike>	Deprecated. Use instead.	
<u>	Deprecated. Use styles instead.	

Deprecated Tags

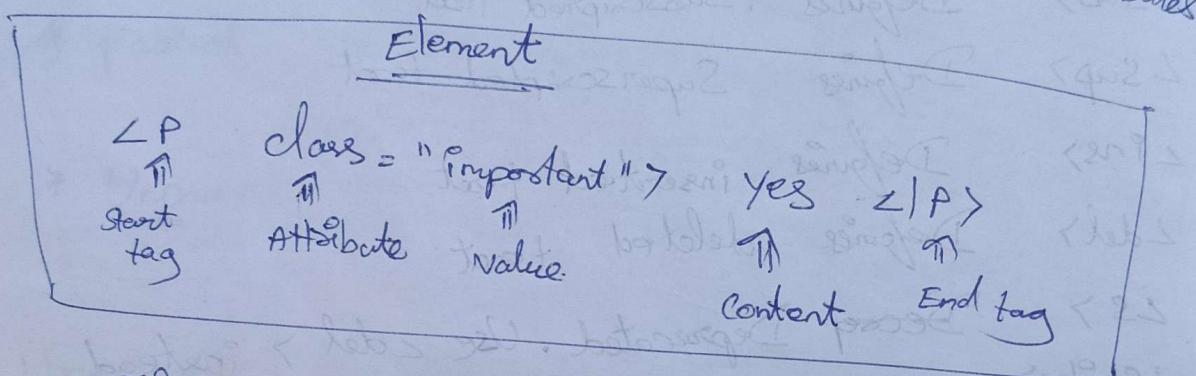
→ Tags which are not in use of they.
are better handled by

Cascading Style Sheet.

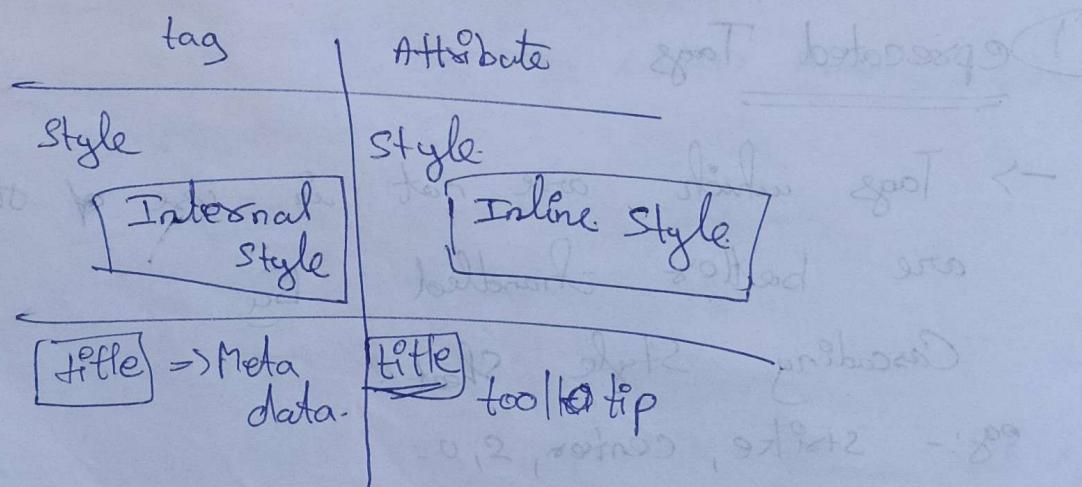
e.g.: - style, center, s, u.

Attributes

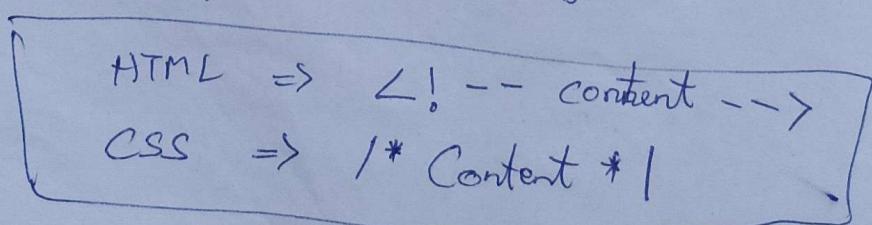
- Additional Information about a Element/Tag
 - Always Specified in Start Tag
 - It Should be with Name = "Value"
 - Case Attribute (Frequently Used Attributes)
 - 1) id → (Target Specific element) (Always unique)
 - 2) class → (Target Multiple elements)
 - 3) Style → Inline Style
 - 4) title → Tooltip
 - Each Tag will have their own.



\Rightarrow The group of elements ~~are~~ is known as elements.



Commands for comment file Content



Syntax

Selector {

Attribute: Value

}

Internal style :-

Style tag given inside a head tag

Grouping html element

We can group an html element in two ways.

* BLOCK LEVEL → Even though the Content is less it will occupy the 100% Space. (or) width
Ex:- Paragraph.

* INLINE LEVEL → It will occupy only the Content Space (or) width.

→ Both Block level and Inline level element are Content wrapper [It will wrap the Content]

ID name Selector:-

If we select something with the id then it is known as id name Selector.

Tag name Selector:-

If the element is selected with the tag name then it is known tag name Selector

Span tag :- It will occupy only the content space, it is used (as) also known as inline level element.

Multiple page Application :-

* Entire application written on more than one single html document is created that is known as multi page application.

* If is also known as classical approach.

Disadvantages

i) It is stopped because, if taken more one it will reload again] To interconnect the web pages we use anchor tag `<a>`

Anchor tag :-

* If is paired tag
* Anchors will have 2 Attributes [href, target]

* Anchors will have their own attributes

* It will convert the text into hyper text

Syntax

` `
Hyper Reference

Reference Link:-

We can give the link in 2 ways

→ Absolute path.

→ Relative Path.

Hypertext:-

The text will have some reference line.

If we click the Hypertext it will navigate
to either webpage (or) website.

Absolute path:-

It will starts with parent node and
ends with child node.

Target:-

* By default target will be in
underscore self.

* If we want the page to be
displayed in next (or) new tab we need to use
blank.

→ id name selector

ph → pure height

vh → view height

. → class name selector

px → width of the border

Relative path:

- Starts from child node and ends with child node
- To achieve relative path (or) to ask Href Controller to go one step back we use (..)
- To step two steps back (.../)
- To step three steps back (.../.../)

-Top :- It Opens the Linked Documents in the full body of the window

-Parent :- It opens the linked document in the Parent frameset.

FrameName :- It opens the linked Document in the parent normal named frame

Block-level Elements :-

<P> <h1>
<div> <nav>

Inline Elements :

<u> <big> <big> <i>
<small> <sup>

Single page Application :-

- Entire Application written on one single html document
- It's an modern approach
- It takes less page reload time.

Image tag:-

It is an unpaired tag (self pair or empty tag)

Syntax:-

src:- Source (we will use both relative path and absolute path)

alt:- Alternative name. (if the image is not loading instead we can give [Image is loading])

Nested Element:-

Element inside a element

Eg:- <p> bold </p>

Basic tags in html.

html

head

body

b

hr

p

text formatting tag

heading tag

div

nav

img

title

style

anchor

span

Table

- HTML Table allow web developers to arrange data into rows and columns.
- Table is collection of rows and columns.
- HTML table is made up of cells
- Table Cell :- Each table cell is defined by `<td>` and `</td>`
- Table Rows :- Each table row starts with `<tr>` and ends with `</tr>` tag.
- Table Head :- Table Head starts with `<th>` and ends with `</th>`
- Table Foot :- Table Foot starts with `<tfoot>` and ends with `</tfoot>`

```
<table>
  <thead>
    <tr>
      <th> </th>
    </tr>
  </thead>
  <tbody>
    <tr> </tr>
  </tbody>
</table>
```

Column Span \Rightarrow Combining the Columns \Leftrightarrow

colspan="2"		

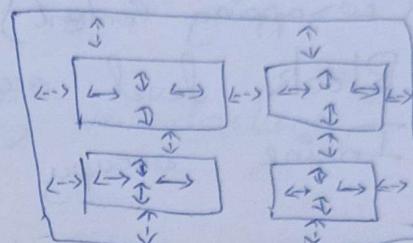
Row Span \Rightarrow Combining the Rows
Row Default = 1
 \Leftrightarrow

Rowspan="2"		

Table Cell - Padding

Space b/w Cell edge and Cell Content
Padding is 0 by default.

Ex:- th, td {
padding: 15px;
% default



\Leftrightarrow Cell Spacing
 \Leftrightarrow Cell padding

Table Cell - Spacing

Space b/w each cell is Cell Spacing
Spacing is 2px by default.

Ex:- table {

border-spacing: 30px;
}

ANCHOR TAG

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

By default, links will appear as follows in all browser

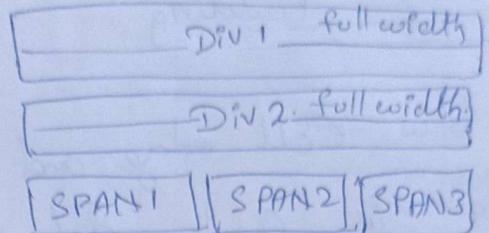
- An Unvisited link is underlined and blue.
- A Visited link is underlined and purple.
- An active link is underlined and red.

Target:-

-self
-blank
-parent
-top

Grouping Content

- <div> - Block Level.
- - Inline Level
- Both are Content wrapper (Wrapping Content)
- Block level occupy entire width of the Screen.
- Inline occupy only Content width.



⇒ HTML Attributes

Attribute ⇒ Description.

alt ⇒ Specifies an alternative text for an image

disabled ⇒ Specifies an input element should be.
- disabled

href ⇒ Specifies the URL (web address) for a link

id ⇒ Specifies a unique id for an element

src ⇒ Specifies the URL (web address) for an image

style ⇒ Specifies an inline CSS style for an element

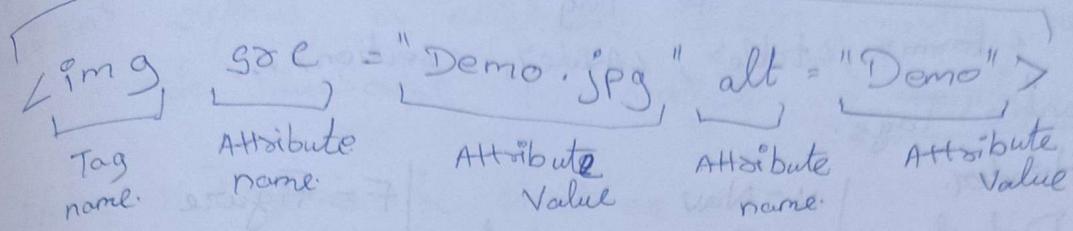
title ⇒ Specifies extra information about an element (displays as a tooltip)

Value ⇒ Specifies the value (text content) for an input element

Image tag

Import Image to your webpage.

HTML IMAGE TAG



LIST (Paired tags)

HTML can have OrderList, UnorderList, Description List

OrderList -

- first item
- second item
- third item

> Starts with

> each list item starts with
in the tag

> we can use attribute type.

eg. <ol type="1" or="1" or="a" or="A">

> Default type will be Number.

UnorderedList -

- first item
- second item
- third item

> Starts with

> each list item starts with in
the tag

> we can use attribute

type

eg. <ul type="disc" or="circle"
or="square" or="none".>

> Default type will be disc

Description List -

The first item.

Description of item.

> Description List starts with the tag <dl>

> Definition term starts with in the tag <dt>

> Definition details starts with the tag <dd>

Semantic Tags *→ feature of HTML5*

- * Semantic word meaning both developer and Because can understand the tag
e.g. `<form>`, `<table>`, `<aside>`, `<section>`...

Semantic Tags

H - Header.	N - Nav	F - Figure
D - Details	S - Summary & Section	F - Figcaption / Footer
A - Aside	T - Time	
M - Main / Main	A - Article	

HTML 4 vs HTML 5

<code><div id = "header"></code>	<code><header></code>
<code><div id = "nav"></code>	<code><nav></code>
<code><div id = "Content"></code>	<code><content></code>
<code><div id = "aside"></code>	<code><aside></code>
<code><div id = "footer"></code>	<code><footer></code>

HTML FORMS

- * HTML forms are Required to collect data from the user
- * HTML form is a Document which stores information of a user on a webserver.
- * Collecting information from the user through electronic formats

HTML Form elements

`<form>` defines an html form for user input
`<input>` defines an input Control (23) values
`<textarea>` defines a multiline input Control (text area)
`<label>` defines a label for an `<input>` element

- fieldset → Groups related elements in a form.
- select → Defines a Drop down list
- option → defines an option in a dropdown list
- button → defines a clickable button (mostly used in JavaScript to activate a script)
- checkbox → Defines a checkbox.
- color → Defines a color picker.
- date → Defines a date control (year, month, day (no time))
- datetime-local → Defines a date and time control (year, month, day, time (no time zone))
- email → Defines a field for an e-mail address.
- file → Defines a file-select field and a "Browse" button (for file uploads).
- hidden → Defines a hidden input field.
- image → Defines an image as the submit button.
- month → Defines a month and year control (no timezone)
- number → Defines a field for entering a number.
- password → Defines a password field.
- radio → Defines a Radio button.
- range → Defines a range control (like a slider control)
- reset → Defines a reset button.
- search → Defines a text field for entering a search string.
- submit → Defines a Submit button.
- tel → Defines a field for entering a telephone number
- text → Default. Defines a single-line text field
- time → Defines a control for entering a time (no timezone)
- url → Defines a field for entering a URL.
- week → Defines a week and year control (no timezone).

Forms Validation.

Multimedia Tags :- Audio, Video, Iframe

* <audio controls>

<source src = " " type = " " >

</audio>

Import audio and video to the web page.

* Audio Attributes :-

> Autoplay, Muted, loop

* <video attributes controls>

<source src = " " type = " " >

</video>

Relative path

* Video Attributes

> Autoplay, Muted, loop, controls

Iframe

* An HTML iframe is used to display ^{some other web page} in a web page

& iframe src = "#" Absolute (P-C) or Relative (C-C)

height = "200"

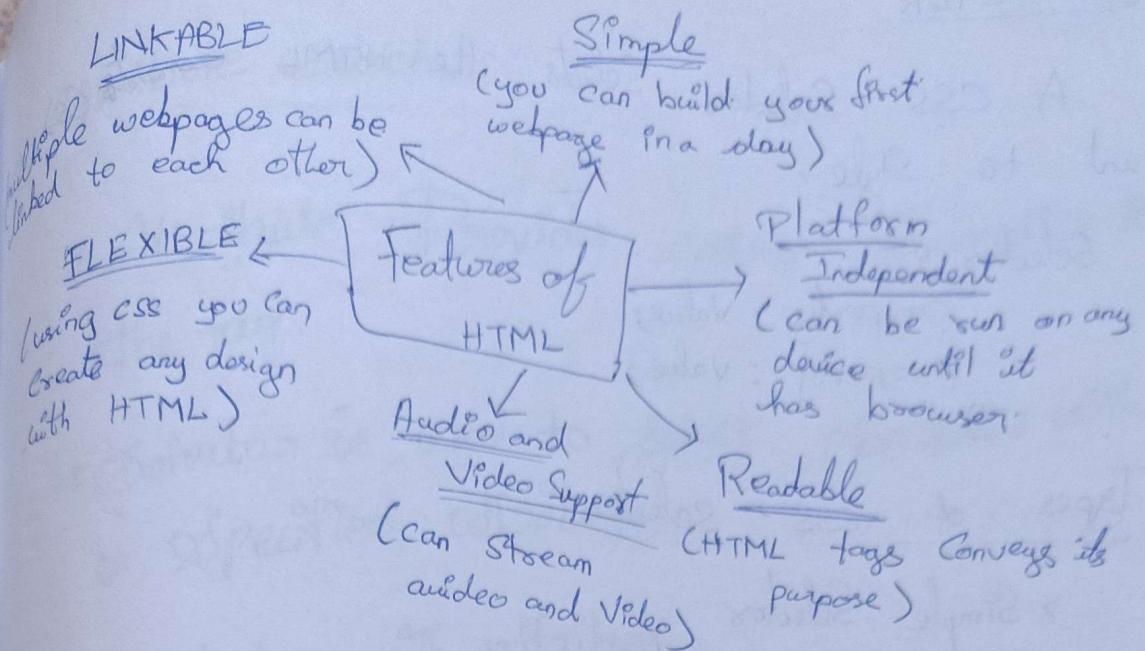
Relative (C-C)

width = "300"

0

title = " iframe example " >

</iframe>



Cascade Style Sheet → [CSS]

⇒ Ways to link CSS

- i) Inline CSS
- ii) Internal or Embedded CSS
- iii) External CSS

i) Inline CSS Style

e.g.: - <P style="color: red;"> Hello CSS </P>

ii) Internal or Embedded CSS Style

e.g.: <Style>

P { color: Value }

</Style>

iii) External CSS Style

* It is used to apply CSS on multiple Pages.

* Extension must be .css for CSS files

SELECTOR

A CSS Selector Selects The HTML elements (s) you want to style..

Selectors {

 Property : Value;

 Property : Value;

}

Types of css Selectors

> Simple Selector

> Combinator Selector

> Attribute Selector

> Pseudo class Selector

> Pseudo Element Selector

> Simple Selector:

* Id Selector (#)

* Class Selector (.)

* Universal Selector (*)

* Element Selector (tag)

* Grouping Selector (p, h1)

> Combinator Selector

> Descendent Selector (space) - direct and indirect children.

> Child Selector (>) - direct children.

> Adjacent Sibling Selector (+) - Next Sibling

> General Sibling Selector (~) - General Sibling

Attribute Selector

* Attribute Provides extra information to the tag.

* Selector is used to select elements with a specified attribute.

* Selector with attribute.

* Element selector [attribute]

* Example : `img [src]`

Attribute Selector

* Selector with attribute :- `img [alt]`

* Attribute with Value :- `img [alt = value]`

* Select the element with Value).

> Attribute with Caret ^ Symbol :-

`img [attribute ^= "Value"]`

(check for Prefix Value)

> Attribute with \$ Symbol :-

`img [attribute $= "Value"]`

(check for Suffix Value)

> Attribute with ~ Symbol :- `img [attribute ~= Val]`

Select Value Containing a Specified word followed by Space

Attribute Selectors

> Attribute with Vertical | symbol:-

`img [attribute] |= "Value"]`

(Selects Values followed by hyphen)

> Attribute with * Symbol:-

`img [attribute *|= "Value"]`

(Selects all Value)

Pseudo classes.

> It is a Selector that assist in the Selection of something that cannot be expressed by a simple Selector.

Selector : pseudo - class

Property : value;

Anchor Pseudo classes:

- * Link
- * Visited.
- * Active.
- * Hover.

VI elements Pseudo classes.

- * Enabled.
- * Disabled.
- * Focus.

Structural Pseudo classes:

- * First-child
- * Last-child
- * Nth-child()
- * First-of-type

- * Last-of-type

Pseudo Class

Pseudo class assist the selection, which can not be expressed by simple selector.

Syntax

Selector : P-C {

prop : value;

}

Pseudo Elements

> Styles Specified parts of an element

Selector :: Pseudo-element {

property : value;

}

> :: first-line

> :: first-letter

> :: before

> :: after

> :: marker

> :: selection

Pseudo element

→ Pseudo element target

Specified part of the element

Syntax

selector :: P-E {

prop : value;

}

Text Property

→ Text Formatting

* Colour

* Text-align

* Text-Transform

* Text-Shadow

* Text-decoration

* Letter-Spacing

* word-Spacing

* Text-indent

Text

* Text-overflow

* word-wrap

* word-break

* writing-mode

Background Property

- * Background - image : url (" .. /image.jpg");
- * Background - repeat : no-repeat | repeat-x | repeat-y | round;
- * Background - size : contain | cover | 100% | 100% | x y Value (100%, 100%)
- * Background - position : right | center | top | bottom | left.
- * Background - attachment : scroll | fixed.
- * Background - origin : content | border | padding - box

Height and width property

- max-height : px;
- height : px;
- min-height : px;
- max-width : px;
- width : px;
- min-width : px;

Colour

- * Colour : hex value = #e6fefef;
- * Colour : rgb - (255, 255, 0);
- * Color : hsl (hue, saturation, lightening)

hue (deg = 0) is red.
120 is green
240 is blue

Saturation	Lightening
↑ 100%	↓ 100%
0%	100%

- * colour : (rgba) (hsla);
a ⇒ opacity | transparency | alpha.

Font property

- * font-size : large | small | medium
- * font-weight : bold | bolder | lighter | normal
- * font-style : italic | normal
- * font-family : serif | sans-serif
- * font-variant : small-caps | normal

Box Model

- ⇒ Box model allow us to design a border around html element and give space b/w elements
- ⇒ It consists of Padding, margin and border.

Padding :- Space b/w Content and element border

Margin :- Space b/w Imaginary line and element border

Border :- lines around html element

Properties

> Margin-style > Padding-style > Border-style

Margin:
: top
: bottom
: left
: right

Padding:
: left
: right
: top
: bottom

Border:
: border
: width
: radius

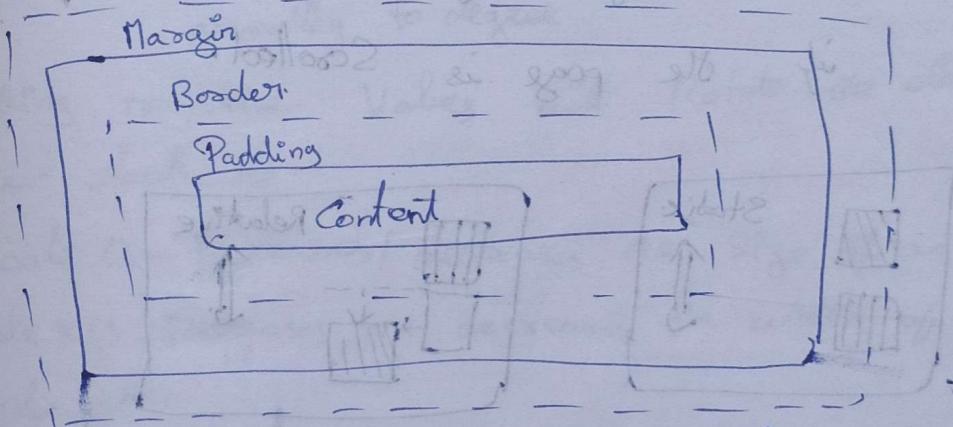


fig - Box-Model.

short hand Properties

transform
transition
animation

padding
margin

Position Property

> The Position Property Specifies the type of Positioning method used for Element.

Static :- Elements are positioned Static by default.

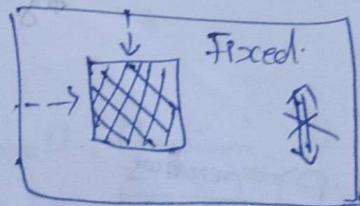
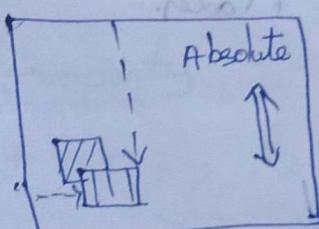
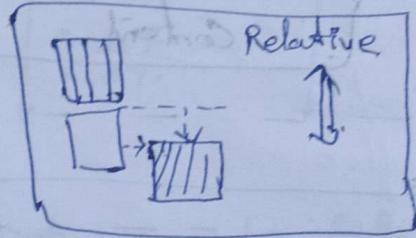
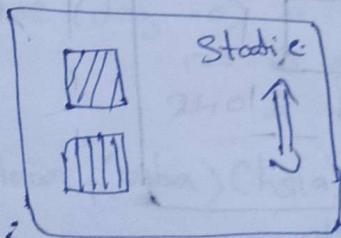
:-> Static Positioned elements are not affected by the top, bottom, left and right properties.

Relative :- Positioned relative to its normal position.

Absolute :- Positioned Relative to the nearest positioned ancestor.

Sticky : Positioned Based on the user's scroll position.

Fixed :- It always stays in the same place even if the page is scrolled.

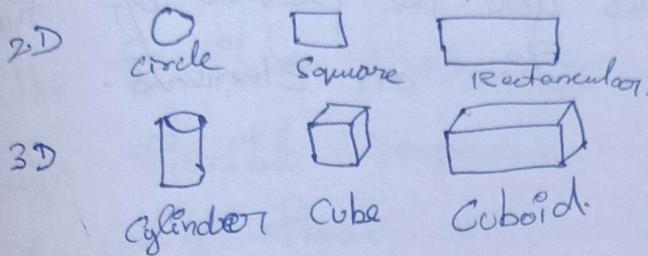


2D and 3D Transforms

Transforms allow you to move, rotate, scale, the element

Difference:

Composing shapes.



2D Transforms

- 1.) Translate() - moves an element from its current position.
- 2.) rotate() - rotate an element clockwise / Anticlockwise according to degree
 - * Using negative values will Rotate the element Counter-clockwise.
- 3.) scale() - increases / decreases the size of an element
- 4.) scaleX() - increases or decreased the width of an element.
- 5.) scaleY() - increases or decreases the height of an element.
- 6.) skew() - Skew an element along the X and Y axis by the given angles.
- 7.) skewX() - Skews an element along the X axis by the given angle [degree 0 - 360]
- 8.) skewY() - Skews an element along the Y axis by the given angle.

3D Transforms

- > rotateX() - rotates an element around its X-axis
- > rotateY() - rotates an element around its Y-axis
- > rotateZ() - rotates an element around its Z-axis

Transition

=> Transition Gives you the period of time to make some action on Elements.

Transition Property

- > Transition
- > Transition-delay
- > Transition-duration
- > Transition-timing-function

Animation

=> CSS Allows animation of HTML elements.
=> An animation lets an elements Gradually change from one Style to another

Animation Property

- > @keyframes
- > animation-name
- > animation-duration
- > animation-delay
- > animation-iteration-count
- > animation-timing-function
- > animation-direction

float, Display & Overflow:

> Float : right \ left \ none.

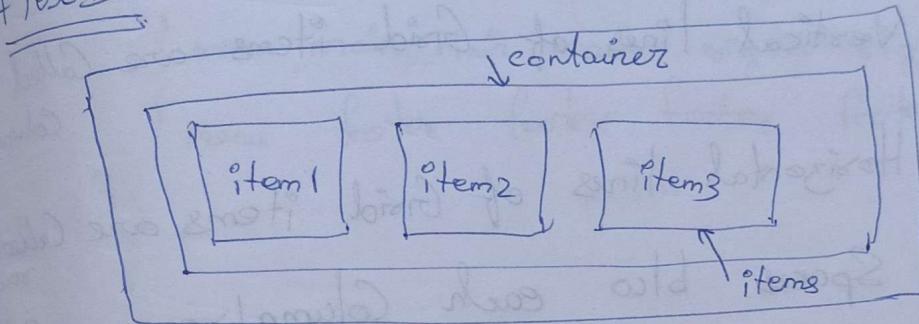
> Display : none, block, inline, inline-block

> Overflow : Hidden, Visible, Scroll.

Flex:

The Flexible Box Layout, makes it easier to design flexible responsive Layout structure without using float or positioning.

Flex Box



Flex Properties

> Display : flex

> Flex-direction : row / column.

> Flex-wrap : wrap / no-wrap / wrap-reverse

> justify-content : flex-start / flex-end / center / space-around / space-between / space-evenly

> align-items : flex-start / flex-end / center / stretch

> flex-basis : length

> align-self : flex-start / end / center.

> flex-grow : <number> 0

> flex-shrink : <number> 1

> order : <integer>

GRID

→ CSS Grid Layout Module Provides Grid based layouts with rows and columns.

→ It is to design web pages without having to use floats and Positioning

GRID rows and columns

> The Vertical Lines of Grid items are called columns

> The Horizontal Lines of Grid items are called rows

> The Spaces b/w each column/rows are called gaps

GRID Lines

> The lines b/w columns are called Column lines

> The lines b/w rows are called row lines

GRID Property

> display : grid;

> grid-template-columns : auto auto auto auto;

> grid-template-rows : auto auto auto auto;

> justify-items ; space-around - evenly-around.

- 7 Column - gap : 20px;
- 7 Row - gap : 20px;
- 7 Gap : even space for column and row.
- 7 Grid - row & Grid - Column
- 7 Grid - area.

item1 {

grid-area: header;

}

grid-Container {

grid-template-areas:

' header header header header header header'

' menu main main main right right'

' menu footer footer footer footer footer';

}

8/12/23

JavaScript / Ecma Script-6

Father of JS - Brendon Eich - 1995

(in 1955 ~~Brendon~~ \Rightarrow Netscape) market strategies of JS

- To make Web page Dynamic we use JS
- European Company Manufacture Association introduced Standard JS to use by everyone.

Introduction

- * Introduced By Brendon Eich in 1995
- * It became an ECMA Standard in 1997
- * JavaScript and Java are Completely different Languages , both in Concept and design.
- * JS is the world's most popular Programming Language which is used in Browser to make Web Page Dynamic

Difference between Java and JavaScript

Java

- Programming Language
- Strictly typed
eg:- `char c = "A";`
necessary to mention the Datatypes
- Java Runs On Virtual Machines
- More Memory Used

JavaScript

- Scripting Language [functionality in the front end]
- Weakly typed Language
eg:- `let identifier = "e"`
Not Necessary of mentioning Datatypes.
- JS will runs on all the Browsers.
- Less Memory Used.
[Light weight language]

Java

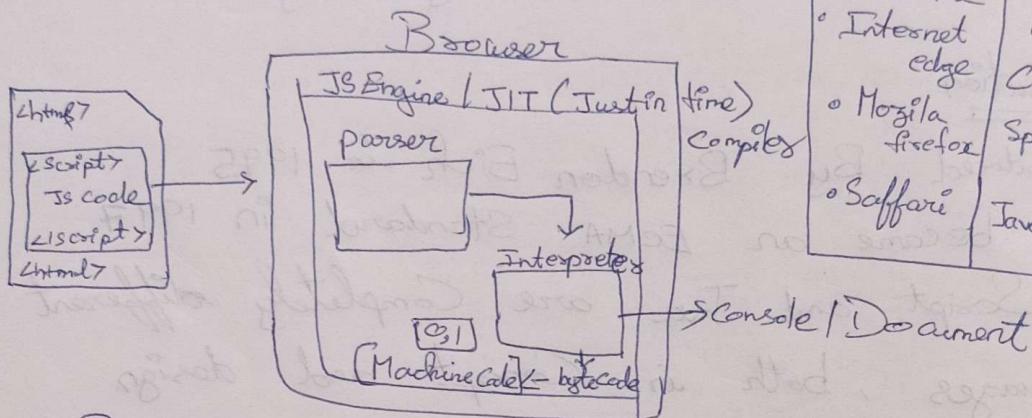
→ Independent Language.

Without support of any other programming language Java can execute.

JavaScript

→ Executed Along with HTML

How JS Works?



Parser is a method which present inside the JS Engine to convert source code to byte code.

Environment of JS

→ Browser

→ Node.js

i) Node.js is combination of a bundle of Chrome V8 engine and built-in methods in C++

ii) It serves as an environment to run JS outside the browser

iii) This invitation helped JS to gain its popularity in usage as a backend language

Characteristics of JS

1 → Purely Object Oriented and object Based

Primitive Datatype

- inbuilt datatypes
- fixed in MemorySize
- String
- Number
- BigInt
- Boolean.

Non-Primitive Datatype

- will be developed by developers
- Not fixed in MemorySize
 - array
 - class
 - function.

new String

new Number

new Array

new function

new date

new Math

new Boolean

Many more inbuilt objects

⇒ Every Object has State and Behaviours of that Object

Let str = "String" ✓

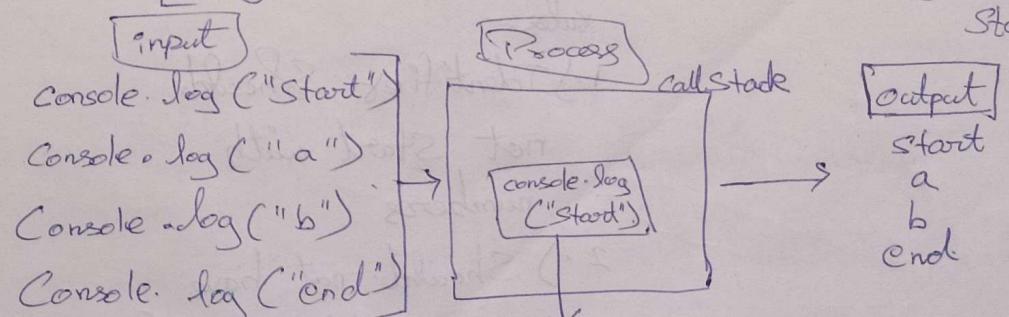
Let str = 'String' ✓

Variable identifies Literals

Document.write()
Object member

2. Synchronous in Nature

⇒ [Single threads It will have Single Call Stack]



at a time only one statement
can be processed.

3.) Case-Sensitive.

3 types of Variables

- * Var
- * Let
- * Const

4.) Interpreted Language.

Converting line-by-line to the machine code.

and Gives the Output (JS `engine.parse()` method)

e.g:- (JavaScript, Ruby, Python)

TOKENS (Smallest unit of every programming language.)
kID

Keywords Identifiers Data Literals

Redefined Words (63) Name Given to the Component by the developers
eg; var, let.....

rules:-

1.) Identifiers Should not start with numbers

2.) Should not have space b/w identifiers.

3.) use Alphabets [A-Z, a-z]

Normal identifier → camelCase

Object identifier → PascalCase

Special identifiers → SNAKE_CASE

- * Keywords Should not be an Identifiers
- * Identifiers Should not be as a keywords.

Data Literals (Fixed Values) ← Literals

String
Number
Boolean
Bigint
null
undefined
Object

Primitive
datatype

Non-Primitive
datatype.

Variable Values

Var str = "Smith"
Console.log(typeof str)

- 1.) It is a keyword
- 2.) It is a Special Operator in js (unary Operators)
- 3.) It will check for the data type.

String

Can declare by 3 ways * " " * , ,

introduced in ES6 version Back ticks. * [] => Template String

let var = ' this is Smith's Phone' X
uncought errors

let var = " this is Smith's Phone" ✓

→ Advantages of using template String

→ Call expression inside Backticks by using
\$ { }

→ no need to use regular expression like "In" instead Enter Button.

Number

Primitive - datatype.

Number - Any Number can will Consider as a

Number datatype in JS