# HTML

HTML tags

\*heading tags (h1 to h6)

-from different size

-for creating multiple lines ##h${content-$}\*6

\*paragraph tag (p)

-for paragraph generate

\*div tag

-to create division or section

-we can create multiple div one in one

\*anchor tag (a)

-reference any file or link

-ex <a href="path">name</a>

-target attribute=use to open same tab or another tab

#image tag (img)

-self closing or void element

-use to add image by path(local)or by link

-ex <image src="path" alt="">

\*list tag(ul)(ol)

- use li for input list items

\*pre tag

-show file as it is you write in code with same spacing and style

\*sub(sub) and super(sup) script tags

\*table tag

-format=<table>

<thead>(table Head)

<tr>(table row)

<th>Heading</th>(table Heading)

</tr>

</thead>

<tbody>(table body)

<tr>

<td>Data</td>(table data)

</tr>

</tbody>

</table>

- row span and column span use to marge row and column

\*input tags(input)

-use type attribute for change type of text (text, number, checkbox, etc)

-if multiple radio have same name then only one is select.

-Select tag is use for Selectbox {

<select name="" id="">

<option value="">India</option>

<option value="">China</option>

</select>

}

-Textarea is use to select fixed size of area for text

\*Form tag

-use to create form which combine all details

\*name attribute act as variable which store data

\*Method get is use to show data by url

\*video, audio tag use to add video, audio in page

\*iframe is use to show some part of any website on our page

//ssl is licence which is use for https

//difference b/n href and src

- in href it create a reference link

- in src it take file to the page

//difference b/n b tag and strong

- <b> = just bold styling (no extra meaning).

- <strong> = bold + conveys importance/priority.

//Block level Element and Inline Element

-block level element take full space like p tag

-block level element take full space(width) also when content is not available

-Inline Element tale limited space on page for display

//Sematic tags

-use to create layout in code

-ex header ,footer, main, selection, aside, article, strong

//Void Element

-Self closing element

-ex hr,br etc

# CSS

\*CSS version 3

\*There are three types of CSS

-1 Inline CSS 2 Internal CSS 3 External CSS

\*Inline CSS

-use style attribute

\*Internal CSS

-use style tag in head tag

-use for only one(working) page

\*External CSS

-Need to create new file with .CSS

-use link-css tag for connect

-use for multiple pages

//For select particular element there are three main method (selector)

-1 id(use # id name in style tag)

-2 class(use same css on multiple element of same class)

(use . class name to select)

-3 tag-name (use for particular tag )

-4 group selector(use comma(,) for select multiple selector)

-5 descendant selector(target parent to children(or grandchild etc))

-6 child selector (only target child using (>) )

-7 Universal selector(work on whole page use (\*))

-8 Sibling selectors(Adjacent Sibling Selector (+) target next sibling)

(General Sibling Selector (~) target all sibling)

-9 Pseudo selector(Pseudo classes(:) -focus, hover, link, active, visited,nth-child, lase-child, first-child, empty)

(Pseudo element(::) -before, after, placeholder, selection, first-line, first-latter)

//CSS Prop

\*Hight Prop=give hight , Width Prop= give Width

\*background-color prop=give background color to part

\*Font Prop

-font-size

-font-weight(bold text)

-font-family(change font)

-font-style

\*Text Prop

-Text-align(position in tag)

-Text-decoration

-Text-Shadow(x y blur color)

-color(#, rgb)

-border(size style color)

\*padding, border, margin

\*justify-content

\*flax-direction

\*flax-wrap

\*align-item(positions items inside each line.)(horizontally)

\*align-content(positions the lines themselves when there are multiple lines)

\*child prop

-order

-align-self

-flex-grow

\*grid

-grid-template-columns: 1fr 1fr 1fr

-grid-template-rows: 200px 300px 400px;

-grid-row

-grid-column

\*Overflow

\*transform (translate, rotate, scale, skew)

-transition-duration

-transition-timing-function

-transition-delay

-transform-origin

\*@keyframes

\*animation(keyframes name, duration, timing-function, delay, iteration-count, direction, fillmode)

\*breakpoints(@media screen and{min-width :xpx})

//position prop (default value =static)

-static

-absolute

-relative

-fixed

-sticky

(supportive prop)-top,bottom,left,right,z-index

$if we give parent relative position and give child absolute position them child move inside the parent only

//If we use same id name for different element then css will work fine but it cause issue in js

//CSS Priority Order

-!important (highest)

-Inline styles (style="")

-ID selectors (#id)

-Class selector(.class)

-Tag-name selector

-Universal selector \* & inherited styles

-Source order (last one wins if same specificity)

-In Internal and External the last one get higher Priority

//Inline v/s Inline-block

**-inline** → fits inside text, no size control.

**-inline-block** → stays inline **but allows box-like styling** (width, height, margin, padding).

//Bg color occurred in content + padding + border

Margin

Margin collapse work only vertical

Size unit

Px fixed size

Vh according to screen size (height)

Vw according to screen size (width)

% according to parent (like body is parent)

Em relative to the font-size of its parent element.

Rem relative to the root element

//Order of use Pseudo Class Selector

-link

-visited0

-hover

-active

//there are two types of variable in css

1-globle variable(accessible in whole page)

2-local variable(accessible in particular element)

\*to define variable we do

--variable\_name-property

\*to use variable

var(variable\_name)

# JS

JS version 5

\*var,let,const,ES6

\*primitive datatype- number , string, boolen, undefined, symbol, null

\*ref type - array, objects, date

\*Operators

+,-,\*,/,%,\*\*

=,+=,-=,\*=,/=,%=,\*\*=

==,===,!=,!==,<,>,<=,>=

&&,||,!

&,|,^,<<,>>

\*Id -snake case(use \_ eg-first\_name) - env

-camel case(use first latter in capital eg-firstName)-variables & functions

-pascal case(use first latter of first word capital eg-FirstName) - react components

\*Control flows

conditional statements - If else

- switch case

functions -function functionname (parameter){function body}

-const fun1= function (parameter){function body}

-arrow function - const fun1= (parameter)=>{function body}

loops -for loop (for in(index) -use on array, for of(element)-use on array, object)

-while loop

-do while loop

\*find, findindex, some, every, foreach, sort, reverse, flat, join

\*String methods

chartAt, indexOc, lastindexOf, includes

trim, split, slice, toUppercase, tolowercase

\*Math, Number, Date

-Math=>sqrt, random, ceil, floor, tound, pow, MIN,MAX

-Number=>toFixed, Parseint, ParseFloat,isNaN,isInterger

-Date=>getFull, getMonth, getday, getdate, gethour, getminutes, getseconds, getTime, Now(), todatestring, totimestring ,(Date.now())

\*Type conversion(explicit conversion) and type coercion(implicit conversion)

\*Scope

-Global Scope

-Functional Scope/Local Scope

-Block Scope (ES6)

{

var- Global Scope variable

let,const- block Scope variable

}

//difference b/n var, let, const

-let,const come in ES6 but var is present before

-the variable declared with var can be redeclared and can be reassigned

-the variable declared with let cannot be redeclared and can be reassigned

-the variable declared with const cannot be redeclared and cannot be reassigned

-var is Global Scope variable and let,const is local Scope/block Scope

//type of null is object

//array is linear data structure which store different type of datatype in contiguous memory location(type of array is object)

//object is collection of multiple key values

//"==" only compare the value but "===" compare value as well as datatype

//short circuit evolution

0, "", false, undefine, null => are falsie rest of all are true

//Object.keys(obj)

//callback function vs higher order function

//array methods

-push,pop,shift,unshift,map,filter,reduce,slice,splice,split,indexof,lastIndexof,concat,includes,find,findindex,some,every,foreach,sort,reverse,flat

console.log(arr.push(60));//return new length

console.log(arr.pop());//return the element which removed from starting

console.log(arr.shift());//return the element which removed from starting

console.log(arr.unshift(24));//return new length

//difference b/n Object.seal anf Object.freeze

-Seal only stop changing object name

-freeze stop Changin anything

//Spread Operators , Rest Operators

//Destructuring and Rest

//hoisiting

-if we access any variable of function before its declairation

-var support hoisiting

-normal function also support hoisiting but other two do not work

//DOM(Doc obj model) Manipulation

//InnerHTML(Use as html content), innerText(use as text data), textContent(use as text data, visible on ui)

//setInterval,setTimeout,clearTimeout

//promise= It is an object which hold feature value

-it has three states panding fulfilled rejected

result

.then((res)=>{})//fulfilled (store result in res)

.catch(()=>{})//rejected (give error)

priority system = sysnc

asyns

promise

setTimeout

//this keyword refrences on different places

//prototype and \_\_proto\_\_