

## CSS -

→ Cascading Style Sheets

Cascade is like a waterfall with different levels at different heights

→ Style Sheet language specifies how things should look like on our webpage. Other types of style sheets are → SASS - Syntactically awesome style sheet and Less - Leaner CSS etc.

→ W3C - W3 consortium brings out standards.

→ Hakon Wium Lee → founder of CSS

→ 3 ways to add CSS →

1. inline - `<tag style=" " >`

2. internal - `<style> </style>`

3. external - `<link rel="stylesheet" href="styles.css" >`  
↳ relation to current file.

→ style is a global attribute.

→ CSS selection → To select HTML to apply CSS rules in between curly braces.

Element selector → selects all elements of that type.

`p{` → applies to all paragraphs.

`}`

Class selector → `.className{ }` used for grouping elements.

can have multiple tags with same class.

Id - selector → `#id{ }` only one element can have ~~with~~ a particular id in a file.

Attribute Selector  $\rightarrow p[\text{draggable}] \{ \}$   $\rightarrow$  select all paragraphs with draggable attribute.

$\hookrightarrow$  Selects both draggable : true & draggable : false set.

To be more specific  $\rightarrow p[\text{draggable} = "false"] \{ \}$

Universal Selector  $\rightarrow * \{ \}$  selects everything

$\rightarrow$  value attribute is used inside  $\langle li \rangle$  to denote its index in ordered list. value can only be integer even if we have type set to alphabet or roman. According to the type, value automatically matches the index.

$\langle ol \text{ type} = "a" \rangle$

$\langle li \rangle 12 \langle li \rangle$

$\rightarrow a. 12$

$\langle li \text{ value} = "3" \rangle 34 \langle li \rangle$

c. 34

$\langle li \rangle 45 \langle li \rangle$

d. 45

$\langle li \text{ value} = "6" \rangle 67 \langle li \rangle$

f. 67

$\langle ol \rangle$

$\rightarrow$  Named colors  $\rightarrow$  colors that can be accessed directly by their name.

colorhunt.co  $\rightarrow$  to find professionally designed color palette.

pixel

$\rightarrow 1 \text{ px} = 1/96^{\text{th}}$  of an inch = 0.26 mm wide & 0.26 mm tall

$1 \text{ pt} = 1/72^{\text{nd}}$  of an inch = 0.35 mm

$\hookrightarrow$  point is used in MS Word font size.

1 em = 100% of parent

$\hookrightarrow$  full width of letter m.

1 rem = 100% of root (relative to size of  $\langle \text{html} \rangle$ )

→ font-weight → normal, bold, lighter/bolder, 100-900  
 Parent = 100 ↑  
 ↓ relative to parent

→ font-family → Helvetica, sans-serif

↑ typeface generic font type if typeface cannot be displayed.  
 sans-serif → normal, without feet. f  
 serif → with feet F

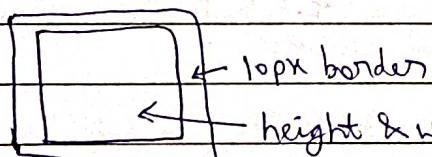
If typeface has multiple words → write inside double quotes.

font-family: "Times New Roman"

custom fonts can be found at fonts.google.com

CSS Box Model - Margin, Padding, Border, height & width.

border : 10px, solid, black.  
 thickness style color

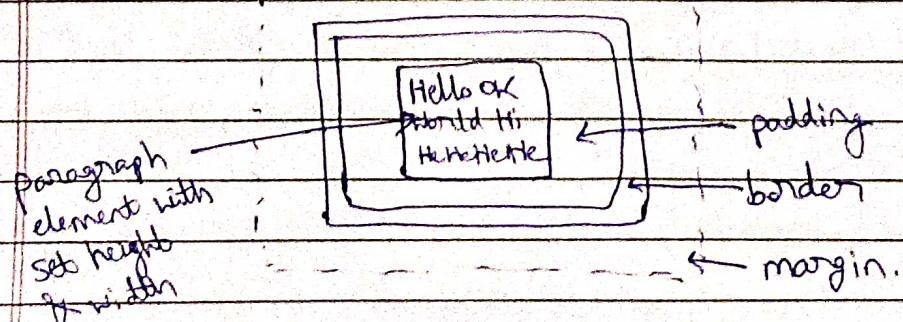


height & width of the element does not change.

border-width: a b c d;  
 ↑ ↓ ↓ ↓  
 Top right bottom left.

border-width: a b;  
 ↑  
 Top+bottom left+right

Padding pushes the border outside and height & width of element remains same.



→ Content division Elements → <div> </div>

↳ To group sections together.

→ We can horizontally center a div by giving it a width of 50% and margin-left of 25%.

→ text-align:center; affects all inline and inline-block elements within the div. Since, image by default is inline, so images also get center aligned.

→ text-transform: capitalize; → converts first letter of each word capital uppercase → all characters to uppercase. Similarly lowercase

→ 4 levels of importance to check which rule will be applied →

Specificity	Position
1	li{color: red;}
2	li{color: blue;}
3	.first-class{color: red;}
4	li[draggable]{color: green;}
5	#first-id{color: orange;}

### Type

<link rel="stylesheet" href=".style.css">

<style></style>

✓ <h1 style=""> </h1>

### Importance

color: red;

✓ color: green !important;

## Combining Selections -

1. Group →  $s1, s2, \dots \{ \}$  → apply css rule to all the selectors  $s1, s2, \dots$
2. Child →  $s1 > s2 \{ \}$  → applies to all  $s2$  only one level down (direct child)  
 ↳ to only direct child.
3. Descendant →  $s1 s2 \rightarrow$  all  $s2$  inside  $s1$  any level deep.
4. Chaining →  $s1 s2 s3 \rightarrow$  apply only where all selectors are true.
5. Combining →  $s1 s2 s3$

CSS Positioning → 4 types =

1. Relative
2. Absolute
3. Fixed
4. Static

1. Static → HTML default flow
2. Relative → relative to original position
3. Absolute → relative to nearest positioned ancestor on top-left corner of webpage.

moves with webpage when scrolling.

ancestor having position property explicitly added. the value of position can be anything → static, relative, absolute, fixed or sticky.

z-index → determines which element goes on top of others.

Higher z value is above lower. Works only in absolute positioning because it brings element out of the flow.

Default z-index is 0.

4. Fixed → relative to top-left corner of browser window.  
 Scroll kرنے pe bhi wahi rahega.

→ To make a circle → border-radius: 50%;

## CSS Display →

→ `<span>` has a default display property on inline. So, it is used to differently style a part of a sentence.

→ `display: inline;` → lets other elements to come in same line.  
`display: block;` → takes full width.

→ inline elements cannot have custom height and width, their size is same as the content inside it.

→ `inline-block` → allows custom size along with same line.

→ `display: none;` → element disappears.

CSS Float → used to wrap text around a particular element.

`img { float: left; }`

To make an element not wrap → `clear: left; / right / both`

→ use float to only wrap text around images.

## Responsive Websites →

4 ways - 1. Media Queries

2. CSS Grid

3. CSS Flexbox

4. External Frameworks like Bootstrap.

breakpoint  
1. Media-query →  $\text{@media}(\downarrow \uparrow) \{ \}$   
means when screen size is 600px then apply these rules.

We can also combine breakpoints →  $\text{@media}(\text{min-width: 600px}) \text{ and } (\text{max-width: 900px}) \{ \}$

→ General Screen Sizes → Mobile → 319px - 480px

iPads & Tablets → 481px - 1200px

Laptops → 1201px - 1600px

Desktops → 1601px & more

→ text-align: left → aligns text to left, similarly for right.  
justify → stretches lines so that each line has same width.  
inherit → inherit from parent.

→ object-fit → specify how an `<img>` or `<video>` should be resized to fit its container.

object-fit: fill; → default, image will be stretched or squished to fit.

contain → resized to fit but aspect ratio maintained.

cover → instead of resizing, image is clipped and maintains aspect ratio.

none → image not resized.