# CSS

Simple hand made css tutorial



Kaushal Bhatol

Learning sites :

<https://cssreference.io/>

<https://www.w3schools.com/cssref/>

<https://developer.mozilla.org/en-US/docs/Web/CSS>

## Rools

-> shortrhands always follow orderd // don’t change order

-> option: 20px 10px 0 8px; // are clock wise. top right bottom left

-> !important after value used for strickly use. // not-recomended

-> Inline style not recomanded

-> use classes insted of id

-> use id if needed.

## Units

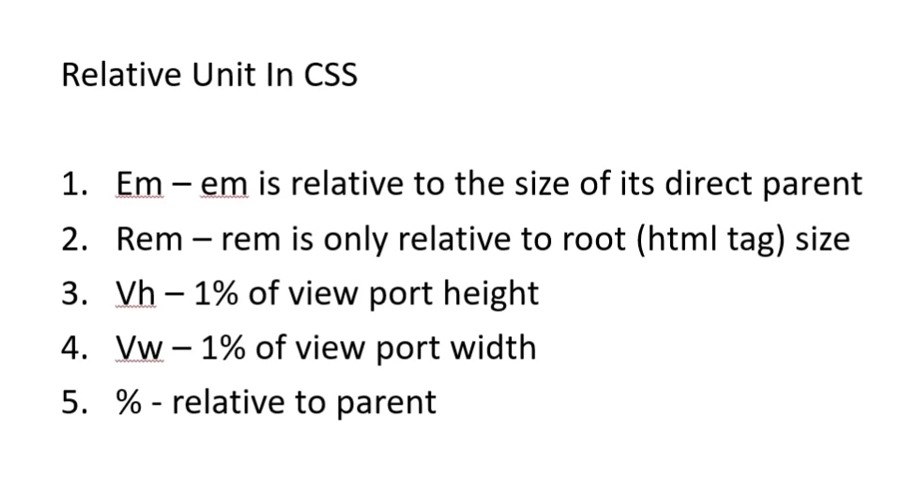
--> **there is two type of units in css**

1. Absolute
2. Relative

|  |  |
| --- | --- |
| **Absolute** | **Relative** |
| Cm | Em |
| Mm | Rem |
| Inch | Vh |
| Px (1Inch = 96px) | Vw |
| Pt (1Inch = 72pt) | % |
| Pc (1Pc = 12Pt) |  |

Absolute = Freedom

Relative = Pairent guidence



* 1em = perent actual size // 2em = 2 x perent size
* 1REm = html actual size // 2rem = 2 x root{html} size
* 100vh = jitni browser ki hight hai // Responsive
* 100vw = jitni broeser ki width hai // Responsive (Most Used)
* % = relative to parent

ex:

p{

width: 50vw; // 50% of screen width {responsive}

height: 50vh; // 50% off screen hight {responsive}

}

## Css adding Types

1. <script> .bgr { background-color: red; } </script> [internal css]
2. <link rel=”stylesheet” href=”style.css” /> [External css]
3. <p style=”color: blue;”> [Inline css]

## Css priority

1. inline
2. internal css
3. external css

\* jo last me hogi wahi mani jayegi pathhar ki lakir.

## Comments

css comments

exp :

/\* This is a comment \*/

## CSS Combinators

1. descendant selector (space)
2. child selector (>)
3. adjacent sibling selector (+)
4. general sibling selector (~)
5. Descendant Selector

The descendant selector matches all elements that are descendants of a specified element.

The following example selects all <p> elements inside <div> elements:

ex:

Html :

<div>

<span>

<p> </p>

</span>

</div>

css :

div > p {  
  background-color: yellow;  
}

div p {  
  background-color: yellow;  
}

1. Child Selector (>)

The child selector selects all elements that are the children of a specified element.

The following example selects all <p> elements that are children of a <div> element:

ex :

Html :

<div>

<p> </p>

</div>

css :

div > p {  
  background-color: yellow;  
}

1. adjacent sibling selector (+)

The adjacent sibling selector is used to select an element that is directly after another specific element.

Sibling elements must have the same parent element, and "adjacent" means "**immediately following**".

The following example selects the first <p> element that are placed immediately after <div> elements:

-> ex :

HTML:

<div>

<p>Paragraph 1 in the div.</p>

<p>Paragraph 2 in h the div.</p>

</div>

<p>Paragraph 3. After a div.</p>

<p>Paragraph 4. After a div.</p>

CSS:

div + p {

background-color: black;

color: white;

}

Output:

Paragraph 1 in the div.

Paragraph 2 in h the div.

Paragraph 3. After a div.

Paragraph 4. After a div.

1. general sibling selector (~)

The general sibling selector selects all elements that are siblings of a specified element.

The following example selects all <p> elements that are siblings of <div> elements

ex:

css:

div ~ p {

background-color: yellow;

}

Html

<p>Paragraph 1.</p>

<div>

<p>Paragraph 2.</p>

</div>

<p>Paragraph 3.</p>

<code>Some code.</code>

<p>Paragraph 4.</p>

output :



## Color

.red {

color: red;

}

-> types of color

1. rgba
2. rgb
3. #ffffff
4. hsl(98, 69%, 59%)

## Background

body {

background-color: yellow;

background-image: url(‘image.png’);

background-repeat: no-repeat;

background-position: 0 0;

background-attachment: fixed;

background: yellow url(‘image.png’) no-repeat right top; //shortcut

}

# multipale background image

body {

background-image: url(‘image1.png’), url(‘image2.png’);

background-position: left top, right bottom;

}

-> background-repeat :

no-repeat // For not reppiting background

repeat // default

repeat-x // repeat horizontly

repeat-y // repeat vericaly

-> background-position:

200px 500px

30% 50%

center

left

right

right top

left top // default

right bottom

left bottom

-> background-attachment

fixed // not scroll with page

-> background-size:

cover // use 100%

50% 50% // y= 50%, x=50%

## hight & width

p {

hight: 10px;

width: 20px;

}

**MAX-WIDTH:**

p {

max-width: 8em; // vadhuma vatu 8em enathi vadhare nahi vadhe

}

note: min-width also same work. // you defnide smallest width, even then smaller not show.

## Border & outline

div {

boreder-width: 2px;

bordr-color: green;

border-style: solid;

border-radius: 5px;

border: 2px solid green; // shortcut

}

-> border-radius

top right bottom left

-> border-style

bordr-style: solid solid dashed solid;

top right bottom left // clockwise

or

1. border-top-style: dashed;
2. border-left-style: solid;

-> border-color

bordr-color: orange green red black;

top right bottom left // clockwise

or

1. border-top-color: green;
2. border-left-color: black;

-> outline

outline is same as border

it was out side of border.

### radius

#border {

border-radius: 8px;

}

-> border-radius :

50% // circle

10px 2px // 10px= top left, bottom right. 2px= bottom left, top right.

2px 10px 3px 8px // clock wise = top left, top right, bottom left, bottom right.

-> border-top-left-radius

-> border-top-right-radius

etc...

## css box model

## Screenshot from 2021-02-22 10-57-33

## Padding & Margin

div{

padding: 0px 2px 3px 4px; // inbox spacing

margin: 2px 2px 4px 3px; // outbox spacing

}

-> padding & margin:

auto // automatic match left and right margi. or horizontaly center

inherite // parent ki margin le leta hai, jo uske uper ki 0 to iski bhi 0.

margin-left: auto; // element dabi baju badhij jagya 6odi deshe, element right ma avi jase.

margin: auto ; // horizontally center, when width given

--> padding & margin collapes

-> jyare 2 box ni margine apva ma ave, ane banne ek bijana najik hoy tyare:

jeni margin vadhare hoy tetlij margin ave 6e.

ex: box 1 ni 30px, box 2 ni 50px

-> tyare bannena vacheni gep 80px ni jagyae 50 px j rahe 6e.

**note** : you cane give nagetive value in margine. {padding not worked}

## align, text-align, spacing, word-spacing, shadow, direction, line-hight, text-indend, white-space

### align

p {

text-align: center; // ta is shortcur in vs code

align-items: center; // vertically (Y), center

justify-content: center; // horizontal center

justify-items: center; // horizontal center

}

img {

vertical-align: top;

}

-> text-align:

left, right, center

justify // fill both sides and do’t zig-zag

-> vertical-align:

top

middle

**text- decoration**

a {

text-decoration: none; // textd is shortcut vs code

}

-> value:

none, underline, line-through, overline, etc..

**transform**

p {

text-transform: upprecase; // all text are upper case ex; THIS IS SOMTHING

}

-> value

upprecase

capitalize // first text is capital. ex This Is Somthing

**letter-spacing, word-spacing**

p{

letter-spacing: 3px; // space between latter.

}

ex: this is somthing

--> word-spacing is same value as latter spacing

-> it gives space beatween words

ex : this is somthing

**text-shadow & box-shadow**

p {

text-shadow: 2px 4px;

text-shadow: 2px 4px grey;

text-shadow: 2px 4px 10px grey;

x y blurr color

box-shadow: 2px 4px 6px 8px grey inset;

x y blurr sprad-radius color type

}

**multipal shadow**

p {

box-shadow: 2px 4px 6px grey, -2px 4px 6px red;

}

-> sprad-radius: kitna failega // howmuch width it get

-> shadow type :

inset // internal shadow

note: box-shadow values are same as text-shadow.

**direction**

p {

direction: rtl;

}

-> diction:

rtl // right to left

ltr // left to right

unset // default

initial //

inherite //

**line-height**

p {

line-height; 50px;

}

--> space betwen two lines hight

**text-indent**

**-->**first line start with some space before

p {

text-indent: 50px;

}

-> output:

loremfgk fjgk;fj

jdffsnknsksksmfkmsd;fm

sdfhjsdhfjjslkjfklsjfksdjkj

**white-space**

p {

white-space: pre;

}

-> value:

pre // as wroten in html file.

pre-line

pre-wrap

inherit

initial

unset // use parent value

nowrap

wrap // devide in new line if width are not enough

normal // default

**word-wrap**

p {

word-wrap: break-word;

}

-> value

break-word //if word is bigger than container, it will breack it

**writing-mode**

p {

writing-mode: vertical-lr; // wrote lines vretically(y)

}

-> value :

horizontal-tb

sideways-lr

sideways-rl

vertical-lr

vertical-rl

initial

unset

## Font

p {

font-family: serif, sans-serif, aria;

font-size: 20px;

font-size: 1.4vw; // view per width

font-weight: bold;

font-weight: 100; // value = 100, 200...900

font-style: bold; // italic

font-variant: small-caps; // small also capital but size small

}

-> vw get automaticly change font size by counting width, it was responsive.

-> **shorthand**

font: style varient font-weight size font-family;

font: italic small-caps 600 1.5em sans-serif;

**custom-font**

types :

ttf

otf

woff

-> download free font for web

-> put in assets/css/font

css:

@font-face{

font-family: my-ab;

src: url(‘font/aclonica.ttf’);

}

p {

font-family: my-ab;

}

**google-fonts**

fonts.google.com

-> select family

-> select another font

-> go to embed

-> copy <link>

-> paste in head

--> now, you can use custom font

p {

font-family: ‘poppins’, sans-serif;

}

## a tag

a {

text-decoration: none;

}

Pseudo-classes

a:hover // when mouse over a link.

a:links //unvisited link

a:visited // visited links

a:active // link in progress

## list

ul {

list-style-type: none; // circule is default value and none = no-style

}

## opacity

# for transparting elements, content

.color-box {

opacity: 0.5;

}

value : 0 to 1 // 0 = transparent, 1 = not-transparent

## Filter

img {

filter: blur(10px); // blurr image

}

-> value:

brightness(50%)

contrast(70%)

drop-shadow(8px 8px 2px gray) // x y blurr color. as same as box-shadow

grayscale(30%) // black and white

hue-rotate(90deg) // color wheel upto 360

invert(100%)

opacity(30%) // transparent

saturate(30%)

sepia(30%)

-> short hand // You can use multiple filters

ex.

filter: blurr(10%, sepia(30%), invert(100%);

## Image sprite

// adding one pic, use sapret pics

(x, y)



ex:

.spirite {

height: 128px;

background: url(‘img1.png’) no-repeat;

} // caneda

.spirite {

height: 128px;

background: url(‘img1.png’) no-repeat;

background-position: 0px -143px ; //x = 0 , y = -143px

} // us-flag

.spirite {

height: 128px;

background: url(‘img1.png’) no-repeat;

background-position: 0px -288px;

} // third flag

## Gradiant

// mixing two and more colors called gradiant

output:



Types of gradiant

1. linear-gradient // line type
2. radial gradiant // cercle type

ex1:

div {

background-image: linear-gradient(green, blue);

background-image: linear-gradient(green 30%, blue 70%); // ocupai color

background-image: linear-gradient(to right, green, blue, red); // give way

background-image: linear-gradient(to bottom right, green, blue, red);

background-image: linear-gradient(120deg, green, blue, red);

}

ex2:

div {

background-image: radial-gradient(green, blue);

background-image: radial-gradient(circle, green, blue, red); // give way

background-image: radial-gradient(to bottom right, green, blue, red);

background-image: radial-gradient(120deg, green, blue, red);

}

## Overflow

## Dispay propertce & visiblity

1. block level // ex div, body
2. inline level // ex. span

li {

display: inline; // display li content in one line

}

**visibility vs display none**

.hidden {

visiblity: hidden; // Not showing but space are not removed

display: none; // not showing and also removed space.

}

## cursor

p {

cursor: ponter;

}

value:

auto //default

all-scroll

cell

col-resize

crosshair

e-resize

greb

help

not-allowed

ponter

progress

wait

zoom-in

zoom-out

## Position, float

**position:**

.box {

position: relative; // you can set any whare using relative position.

left: 10px;

}

-> value:

fixed // fix in the page as a view angle

absolute // free from any limitation, qualification, condition or restriction

relative // something dependent upon external conditions

**float :**

.first {

background-color: red;

height: 45px;

width: 50%;

display: inline-block;

float: right:

}

.second {

background-color: blue;

height: 45px;

width: 50%;

display: inline-block;

}

output:



## box-sizing

.container {

width: 100%;

height: 50px;

padding: 5px;

border: 3px solid black;

box-sizing: border-box; // remove horizontal scroll bar

}

.first {

background-color: red;

height: 45px;

width: 50%;

display: inline-block;

float: right:

}

.second {

background-color: blue;

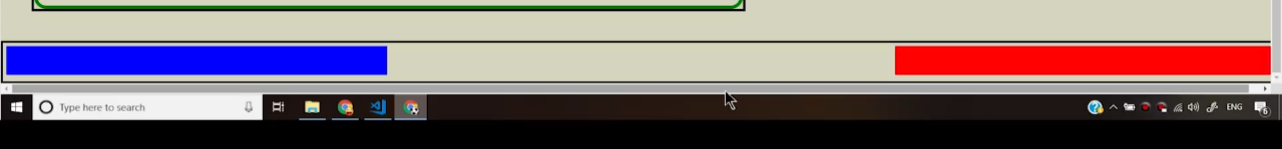
height: 45px;

width: 50%;

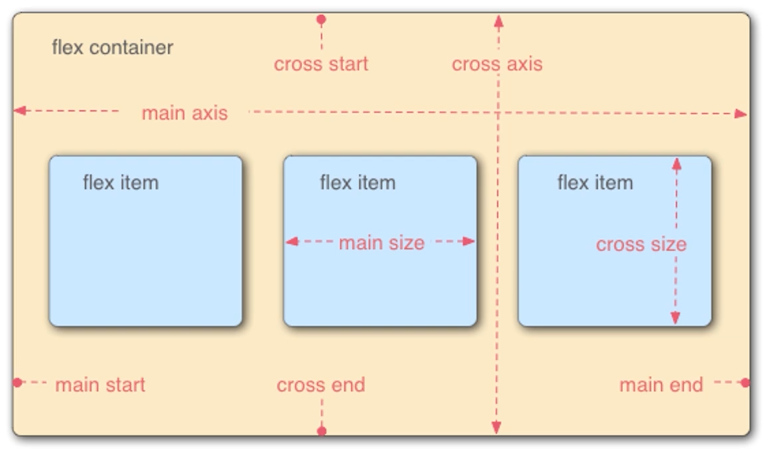
display: inline-block;

}

output:



# **Flax-box**



html:

<h2> This is my flex </h2>

<div class=”myflex”>

<div>1</div>

<div>2</div>

<div>3</div>

</div>

CSS:

.myflex {

display: flex;

flex-direction: row; // ada 3 booxes

}

.myflex > div {

background: red;

width: 80px;

height: 80px;

font-size: 25px;

margin: 8px;

}

Output:



-> flax-direction

column // vertical

column-reverse // ulta vertical. ex

3

2

1

row // horizontal

row-reverse // ulta horizotal. ex 3 2 1

-> flax-wrap : [valvu jyare badhu nanu hoy tyare]

.myflax {

flax-wrap: nowrap; // default, don’t come in second line

lax-wrap: wrap; // jagya nani thata biji line ma avshe

flax-wrap: wrap-reverse; // undhi rite biji line ma, ex 3,2 uper 1 nicheni line ma.

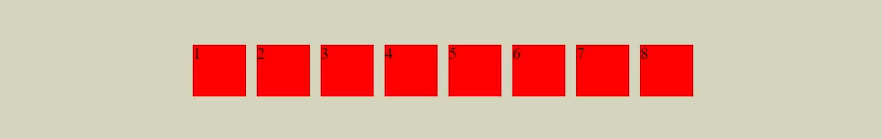
}

## Justify

.myflax {

justify-content:center;

}

output:

-> vertically center

.myflax {

align-items: center;

}

-> justify-content & align-items:

center // in flex center

flax-start // start where the flax is started. ex. our situation it goes left

flax-end // start where the flax is started. ex. our situation it goes right

alig-items: baseline // in comman line

**exp**

css :

.continer {

display: flex;

}

## Properties for flex container

-> **flex-direction:**

row //default Horizontal

row-reverse

column // vertical

column-reverse

-> **flex-wrap:**

nowrap //default

wrap // responsive (come in new line if width is short)

-> **flex-flow:** {combination of direction and wrap}

ex:

flex-flow: column wrap;

-> **justify-content:**

center // all box in center

space-between // take a eqal space between box and use maximum width

space-evenly // take a eqal space between box and border

space-around // equal space for each box.

-> **align-items:**

center // vertically center

flex-end

flex-start // default

stretch // creat own space when wrapped

## Propertice for flex items

-> **order:** (jeno order moto ae 6elle jay)

2

ex:

.items-1 {

order: 2;

}

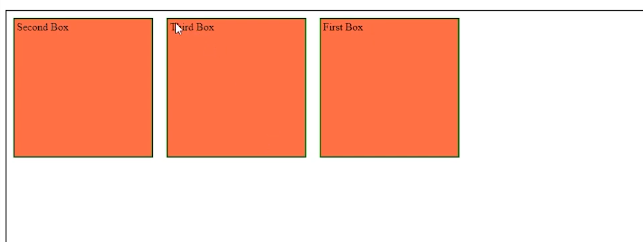
.items-2 {

}

.items-3 {

}

output:



-> **flex-grow: 2;**

-> bigger width then othr. if width is samll all are same.

**-> flex-shrink: 2;** //oposie of grow

-> **flex-basis: 120px; // similer to width, height and flex-grow**

> when flex-direction set to row flex-basis will control width

> when flex-direction set to column flex-basis will control hight

-> **shothand**

flex: flex-grow flex-shrink flex-basis ;

flex: 2 2 230px;

-\*> **align-self:**

flex-end //that solo box go to end of flex-end

flex-start // oposite of flex-end

center // vertically and horizontally center

## a : priority and how to write

default :

a:link {}

a:visited {}

a:hover {}

a:link {}

Note :

-> always write below style

-> active always last

# Pseudo

## Pseudo-Classes

|  |  |  |
| --- | --- | --- |
| **Selector** | **Example** | **Example description** |
| [:active](https://www.w3schools.com/cssref/sel_active.asp) | a:active | Selects the active link |
| [:checked](https://www.w3schools.com/cssref/sel_checked.asp) | input:checked | Selects every checked <input> element |
| [:disabled](https://www.w3schools.com/cssref/sel_disabled.asp) | input:disabled | Selects every disabled <input> element |
| [:empty](https://www.w3schools.com/cssref/sel_empty.asp) | p:empty | Selects every <p> element that has no children |
| [:enabled](https://www.w3schools.com/cssref/sel_enabled.asp) | input:enabled | Selects every enabled <input> element |
| [:first-child](https://www.w3schools.com/cssref/sel_firstchild.asp) | p:first-child | Selects every <p> elements that is the first child of its parent |
| [:first-of-type](https://www.w3schools.com/cssref/sel_first-of-type.asp) | p:first-of-type | Selects every <p> element that is the first <p> element of its parent |
| [:focus](https://www.w3schools.com/cssref/sel_focus.asp) | input:focus | Selects the <input> element that has focus |
| [:hover](https://www.w3schools.com/cssref/sel_hover.asp) | a:hover | Selects links on mouse over |
| [:in-range](https://www.w3schools.com/cssref/sel_in-range.asp) | input:in-range | Selects <input> elements with a value within a specified range |
| [:invalid](https://www.w3schools.com/cssref/sel_invalid.asp) | input:invalid | Selects all <input> elements with an invalid value |
| [:lang(](https://www.w3schools.com/cssref/sel_lang.asp)*[language](https://www.w3schools.com/cssref/sel_lang.asp)*[)](https://www.w3schools.com/cssref/sel_lang.asp) | p:lang(it) | Selects every <p> element with a lang attribute value starting with "it" |
| [:last-child](https://www.w3schools.com/cssref/sel_last-child.asp) | p:last-child | Selects every <p> elements that is the last child of its parent |
| [:last-of-type](https://www.w3schools.com/cssref/sel_last-of-type.asp) | p:last-of-type | Selects every <p> element that is the last <p> element of its parent |
| [:link](https://www.w3schools.com/cssref/sel_link.asp) | a:link | Selects all unvisited links |
| [:not(selector)](https://www.w3schools.com/cssref/sel_not.asp) | :not(p) | Selects every element that is not a <p> element |
| [:nth-child(n)](https://www.w3schools.com/cssref/sel_nth-child.asp) | p:nth-child(2) | Selects every <p> element that is the second child of its parent |
| [:nth-last-child(n)](https://www.w3schools.com/cssref/sel_nth-last-child.asp) | p:nth-last-child(2) | Selects every <p> element that is the second child of its parent, counting from the last child |
| [:nth-last-of-type(n)](https://www.w3schools.com/cssref/sel_nth-last-of-type.asp) | p:nth-last-of-type(2) | Selects every <p> element that is the second <p> element of its parent, counting from the last child |
| [:nth-of-type(n)](https://www.w3schools.com/cssref/sel_nth-of-type.asp) | p:nth-of-type(2) | Selects every <p> element that is the second <p> element of its parent |
| [:only-of-type](https://www.w3schools.com/cssref/sel_only-of-type.asp) | p:only-of-type | Selects every <p> element that is the only <p> element of its parent |
| [:only-child](https://www.w3schools.com/cssref/sel_only-child.asp) | p:only-child | Selects every <p> element that is the only child of its parent |
| [:optional](https://www.w3schools.com/cssref/sel_optional.asp) | input:optional | Selects <input> elements with no "required" attribute |
| [:out-of-range](https://www.w3schools.com/cssref/sel_out-of-range.asp) | input:out-of-range | Selects <input> elements with a value outside a specified range |
| [:read-only](https://www.w3schools.com/cssref/sel_read-only.asp) | input:read-only | Selects <input> elements with a "readonly" attribute specified |
| [:read-write](https://www.w3schools.com/cssref/sel_read-write.asp) | input:read-write | Selects <input> elements with no "readonly" attribute |
| [:required](https://www.w3schools.com/cssref/sel_required.asp) | input:required | Selects <input> elements with a "required" attribute specified |
| [:root](https://www.w3schools.com/cssref/sel_root.asp) | root | Selects the document's root element |
| [:target](https://www.w3schools.com/cssref/sel_target.asp) | #news:target | Selects the current active #news element (clicked on a URL containing that anchor name) |
| [:valid](https://www.w3schools.com/cssref/sel_valid.asp) | input:valid | Selects all <input> elements with a valid value |
| [:visited](https://www.w3schools.com/cssref/sel_visited.asp) | a:visited | Selects all visited links |

**note: active gose last.**

## Pseudo-elements

**usage:**

Style the first letter, or line, of an element

Insert content before, or after, the content of an element

Syntax:

selector::pseudo-element {  
  property: value;

content: "Read this -"  
}

|  |  |  |
| --- | --- | --- |
| **Selector** | **Example** | **Example description** |
| [::after](https://www.w3schools.com/cssref/sel_after.asp) | p::after | Insert something after the content of each <p> element |
| [::before](https://www.w3schools.com/cssref/sel_before.asp) | p::before | Insert something before the content of each <p> element |
| [::first-letter](https://www.w3schools.com/cssref/sel_firstletter.asp) | p::first-letter | Selects the first letter of each <p> element |
| [::first-line](https://www.w3schools.com/cssref/sel_firstline.asp) | p::first-line | Selects the first line of each <p> element |
| [::marker](https://www.w3schools.com/cssref/sel_marker.asp) | ::marker | Selects the markers of list items |
| [::selection](https://www.w3schools.com/cssref/sel_selection.asp) | p::selection | Selects the portion of an element that is selected by a user |