**­­CSS🡪Cascade Style Sheet**

Way to interlink

**>inline style(style as attribute)**

Inside the tab (<p style=” ”></p>)

**>internal style(style as tag)**

<head>

<style>

Selector{

Property:value;

}

</style>

</head>

**>external style**

Selector{

Property:value;

}

**SELECTOR**

1. Simple Selector

2. Combinator Selector

3. Attribute Selector

4. Pseudo-class Selector

5. Pseudo-Element Selector

**1. Simple Selector (if the help of id, class , tag universal )**

**a>Id name Selector** 1ST priority

**b>Class name Selector** 2nd priority

**c>Tag name Selector** 3rd priority

**d>Grouping Selector** (combination of id as a tag)

**e>Univrsal Selector** 5th or last Priority

**Common Mistakes (if CSS style not applied check these mistakes)**

1. check the selector

2. check the CSS link in HTML link tag inside the head tag

**2. Combinator Selector (based on element relationship it will target the tag)**

**1.Descended selector( selection direct and indirect child)**

div p{

Property : value;

}

**2. Child Selector**(selection of only direct child)

Div > p{

Property : value;

}

**3. Adjacent Selector**(selection of next to the one sibling)

Div + p{

Property : value;

}

**4.Genaral sibling selector**(selection of all sibling)(tilt~)

Div ~ p{

Property : value;

}

**3. Attribute Selector (with the help of attributes we will select the element)**

**1.Just with attribute**

Img[alt]{

Property:value;

}

**2. Attribute with value**

Img[alt = “value”]{

Property : value;

}

**3. ^🡪 check for prefix word**

Img[alt^ = “value”]🡪 (^🡪 check for prefix word)

{

Property:value;

}

**4. $🡪 check for suffix word**

Img[alt$ = “value”]🡪 ($🡪 check for suffix word)

{

Property:value;

}

**5. |🡪 check the value followed by hyphen (-)** 🡺(value should be in prefix)

Img[alt |= “value”]🡪 (|🡪 check for suffix word)

{

Property:value;

}

**Note**

1.if any alt value inside the double quotes (””) it will also consider as default hyphen present next to ex🡺(alt=”lamp”).

2.if we give space next to the value it w’ont consider as hyphen ex🡺( alt=”lamp “)

**6.check for the value followed by space**

Img[alt ~= “value”]🡪 (~🡪 check for both prefix and suffix words in the attribute value)

{

Property:value;

}

**7. Check for the value which given in selector value**

Img[alt \*= “value”]🡪 (\*🡪 check for letters present in the attribute value)

{

Property:value;

}

**4. Pseudo-class Selector (to manipulating the existing tag)**

Syntax

Selector:pseudo-class{

Properties:value;

}

**🡺If you want to change block level to inline or inline to block level Grouping we can use DISPLAY property**

**Display : block;**

**Display : inline;**

**Display : none;**

**Didplay:inline-block(min-width is mandayory)**

**🡺Image transformation we use Property OPACITY: ranges(0.1 highest transformation – 1.0 lowest transformation)**

**1. Anchor Pseudo class**

1. :link
2. :visited
3. :active
4. :hover

**2. UI Element Pseudo class (works on forms)**

1. :enabled
2. :disabled
3. :Focus

**3. A Structural Pseudo class**

1. :first-child
2. :last-child
3. :nth-child()
4. :first-of-type
5. :last-of-type

**5. Pseudo-Element Selector (will target the specific element)**

* **::first-letter**
* **::first-line**
* **::selection\***
* **::marker**
* **::before\*(content =” ” ) content property is important**
* **::after\***

**Position property(default static(absolute, relative) positioning the element**

* **Position : relative; (top, bottom, right, left)**
* **Position : absolute; (top, bottom, right, left)**
* **Border-bottom-style : none;**
* **Border-top-style : none;**

**🡺Imaginary line (starts from beginning of the page)**

**Properties**

**Text Property**

>color : red;(6 values)

>textalign: center;

>text-indent:100px;

>text-shadow:1px 1px black;

>text-decoration:underline or overline or line-through or none ……

>text-height:50px;

>wordspacing:20px;

>letter-spacing:5px;

>text-transform : capitalize or uppercase or lowercase;

>writtiy-mode:

>wordwrap:

>text-overflow:

**Background Property🡪color(mentioning height is mandatory) ,image**

Background-image:url(“ ”);

Background-repeat: repeat ;||no-repeat ;||repeat-x ;||repeat-y ;||round;

Background-size:contain;||cover;||100%||xy values(100% 100%)

Background-position:right;||top;||bottom;||left;||center;

Background-attachment:scroll;||fixed;

Background-clip:content;||border;||padding-box;

**Height and Width Property**

**Height Property**

* Min-height(if content is getting large it will also increase)
* Height(if content is getting large it will Overflow)**(Overflow:hidden or scroll)**
* Max-height(same as Height)

**Width property**

* Min-width(will be extend)
* Width(fixed)
* Max-width(fixed)

**If we use Display : inline-block we must use Width property**

**Box Model(if we want to give a space in and around for html element)**

1. **Margin🡪Space Between Imaginary line and element border**

**Margin:top;||bottom;||left;||right;**

1. **Padding🡪Space Between content and Element border**

**Padding: top;||bottom;||left;||right;**

1. **Border🡪Lines around html element**

**Border-color||width-||radius-||**

**Color or Background-color Property**

* **:direct color;**
* **:rgb(red,green, blue); 0-255**
* **:rgba(red,green,blue,alpha);0-255 ,alpha**🡪**opacity or Transparancy**
* **:hsl( hue,saturation,lightness);**
  + **hue**🡪**color in degree(0-359)(0**🡪**red , 120**🡪**green ,240**🡪**blue)**
  + **saturation**🡪 **Highest saturation=0% ,Lowest saturation=100%**
  + **lightness**🡪**Highest lightness=100% ,Lowest lightness=0%**
* **:hsla(hue,saturation,lightness,alpha);**
* **:hexvalues; 6-digits (combination of alpha numberic)** 
  + **#-->starts with ash # will have (a-e and 0-9)**

**Font Property**

**Font-size:large;||small;||medium;**

**Font-weight:bold;||bolder;||lighter;||normal;**

**Font-style:italic;||normal;**

**Font-family:serif;||san-serif;**

**Font-variant:small-caps;||normal;**

**Position Property(if we want to positioning the element any where on the webpage)**

* **Position:static;(default)**
* **Position:absolute; (Grandparent margin as 0px;)**
* **Position:relative; (Parent margin as 0px;)**
* **Position:sticky;**
* **Position:fixed;**
* **Top:**
* **Left:**
* **Right:**
* **Bottom:**

|  |  |
| --- | --- |
| Absolute | Relative |
| 1. Consider grandparent margin as 0px | 1. Consider Parent margin as 0px |
| 1. Absolute will share the space for next element | 1. Relative will not share the space for next element |

|  |  |
| --- | --- |
| Absolute | Fixed |
| 1. Can move to grandparent to parent | 1. Will be fixed the particular position |
|  | 1. Always consider imaginary line as 0px |

**Transform(If you want to move||scale||rotate||skew)**

**>2D(X,Y) >3D(X,Y,Z)**

* **Transform:translate(2d),translateX(250px)(will move in X axis),translateY(250px) (will move in Y axis)**
* **Transform:Rotate(25deg);|| RotateX(25deg);|| RotateY(25deg);|| RotateZ(25deg);**
* **positive degree value will rotate in clock wise**
* **Negative degree value will rotate in clock wise**
* **Transform:Scale(1,1.2)||scale(1.2,1)||scale(1.2,1.2)||scaleX(1.2)||**

**scaleY(1.2)**

* **Transform:skew(250deg,250deg)||skewX(25deg)||skewY(25deg)**

**Box-shadow: x y blur color;**

**Transition Property(smooth effect)(short hand property)**

/\* transition: height 10s ,width 5s; \*/

    transition-delay: 3s ;

    transition-duration: 6s;

    transition-timing-function: linear; same speed throughout the duration

:ease; start slowly,then fast,end slowly

:ease-in; start slowly

:ease-out;slow end

:ease-in-out;slow start and slow end

**Animation**

Without using hover we can make element to animate is known an animation.

**Properties of Animation**

   animation-name: Para2;

    animation-duration: 10s;

    animation-iteration-count: infinite;

    animation-direction: alternate-reverse;

    /\* animation-delay: 4s; \*/

     position: relative;

    animation-timing-function: linear;

**Two ways of Animation:-**

1.From and To

@keyframes Para1{

    from{

        background-color:yellow;

    }

    to{

        background-color: aqua;

    }

}

2.Staging

@keyframes Para2{

    0%{

        background-color: aqua;

        top:0px;

        left: 0px;

    }

    25%{

        background-color: purple;

        top: 0px;

        left: 300px;

    }

    50%{

        background-color:green ;

        top:300px;

        left: 300px;

    }

    75%{

        background-color: crimson;

        top:300px;

        left: 0px;

    }

    100%{

        background-color: magenta;

        top:0px;

        left: 0px;

    }

}

**Flex**

* **Layout Module(display:flex)**
* **Flex can align in only 1 dimention**
* **Responsive Webpage(based on the screen size it will align)**
* **Flex-item(Height and Width)(for the parent element we can give display:flex and inside that the childs are to be called as flex-item )**

**device<=576px extra small**

**device >=576px small device**

**device >=739px medium device**

**device >=1200px large device**

**device >1400 extra large**

**Properties of flex**

**display:flex;**

* **flex-warp:wrap;||nowarp;||wrap-reverse;**
* **flex-direction:row;||column;||row-reverse;||column-reverse;**
* **gap:30px;**
* **flex-basis:400px;only in pixel(it will not cover the space exactly)**
* **flex-grow:1; only in point(to cover the empty space)**
* **flex-shrink:2; (to compress)**
* **order:4;(depends on number of elements)**

**main axis or horizontal axis or x-axis(justify-content:flex-start(default))**

* **justify-content:flex-start;||center;||flex-end;||space-between;||space-around;||space-equally;**

**cross axis or vertical axis or y-axis(align-item:stretch(default))**

* **align-item:flex-start;||center;||flex-end;||baseline;**

**GRID**(when you trying to create template will go with a grid)

* **Layout Module(display:grid;)**
* **Grid can align in 2d dimentional**
* **Responsive Webpage(based on the screen size it will align)**
* **Grid-item🡪Templete’s**

Grid property

* Display:grid;
* Grid-templete-columns:1fr 1fr;(fractional unit)||50% 50%;||250px 250px;
* Grid-templete-rows:100px 100px 100px;|| only in pixels
* Gap🡪gatters
* Rowgap🡪gap between two rows
* Column🡪gap between two column
* Rowline🡪 lines between two rows(starts from top side of item 1)
* Columnline🡪lines between two columns(starts from left side of item 1)