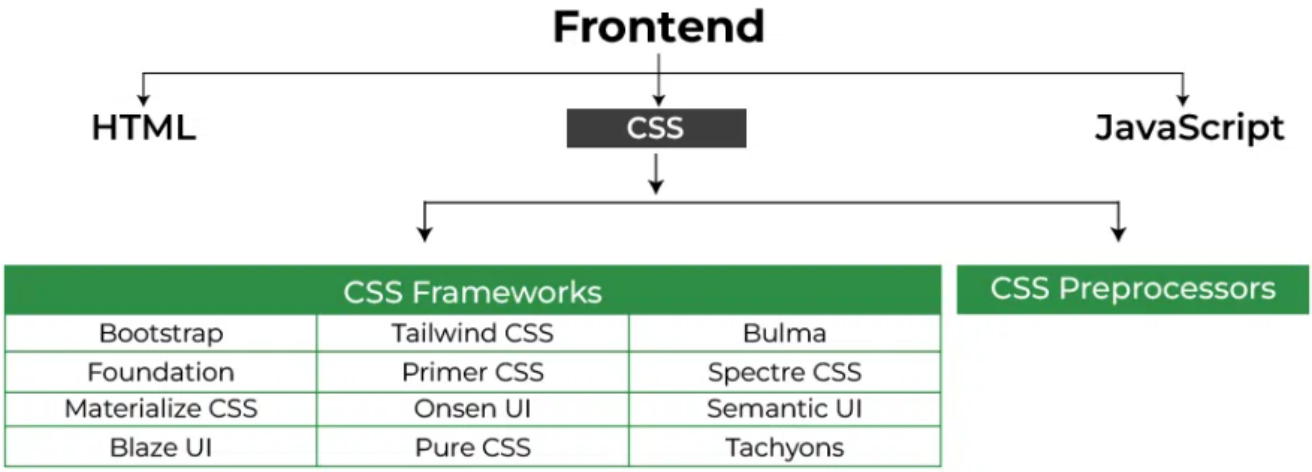
CSS is used to enhance the visual presentation of web pages. The cascade is the algorithm for solving conflicts where multiple CSS rules apply to an HTML element. Stylesheet is a set of CSS rules used to control the layout and design of a webpage or document.



Styling is an essential property for any website. It increases the standards and overall look of the website that makes it easier for the user to interact with it.

```
body {
    background-color: lightgray;
}
h1 {
    color: green;
    text-align: center;
}
p {
    font-family: sans-serif;
    font-size: 16px;
}
```

Inline css will override external css and interal css.

Types of CSS: There are three types of CSS which are given below.

- **Inline:** Inline CSS contains the CSS property in the body section attached with the element known as inline CSS.

```
<body>
  <p style="color: #009900; font-size:50px;
  font-style:italic; text-align:center;">
hello
</p>
</body>
```

- **Internal or Embedded:** The CSS ruleset should be within the HTML file in the head section i.e the CSS is embedded within the HTML file.

```
<head>
  <title>CSS</title>
  <style>
    .one {
        color: #009900;
        font-size: 50px;
        font-weight: bold;
    }
</style>
</head>
<body>
<div class="main">
    <div class="one">Hello</div>
</div>
</body>
```

- **External:** External CSS contains a separate CSS file that contains only style property with the help of tag attributes.

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="style.css">
```

- **Linking External Style Sheet**

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Bootstrap demo</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css"
  </head>
  <body>
    <h1>Hello, world!</h1>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.m
  </body>
</html>
```

- **Importing External Style Sheets**

<link> @import

```
<style>
@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@200&display=swap');
</style>
```

CSS rules to specify families

```
font-family: 'Poppins', sans-serif;
```

While HTML uses tags, CSS uses rulesets. A style rule set consists of a selector and declaration block.

1. **Selector:** A selector in CSS is used to target and select specific HTML elements to apply styles to.
2. **Declaration:** A declaration in CSS is a combination of a property and its corresponding value.

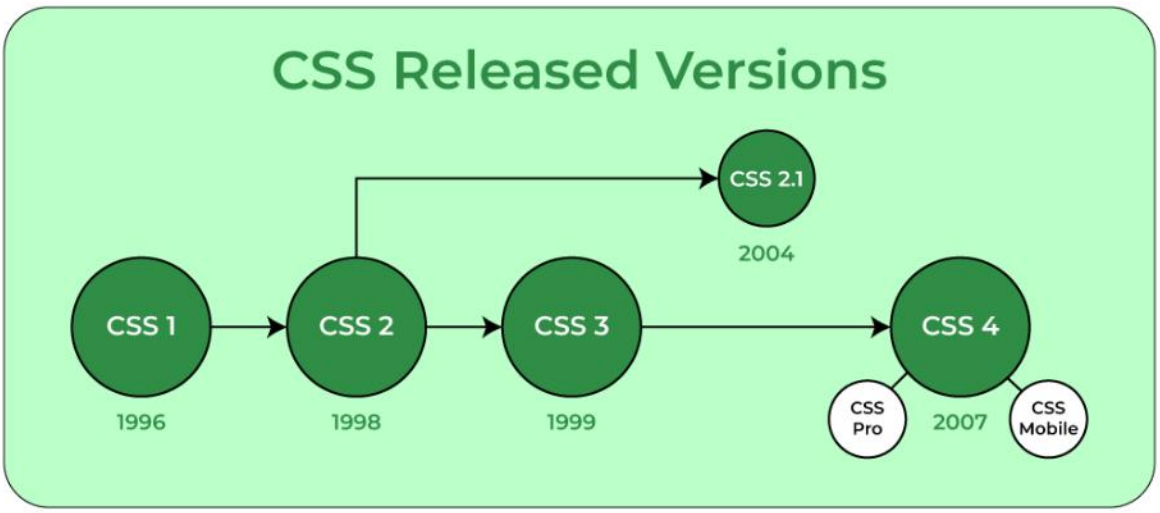
```
Selector -- h1
Declaration -- {color:blue;font size:12px;}
```

```
h1 { color: green; font-family:

 |      |      |        |

Selector Property Value   Property
```

CSS versions release years:



Comment=>

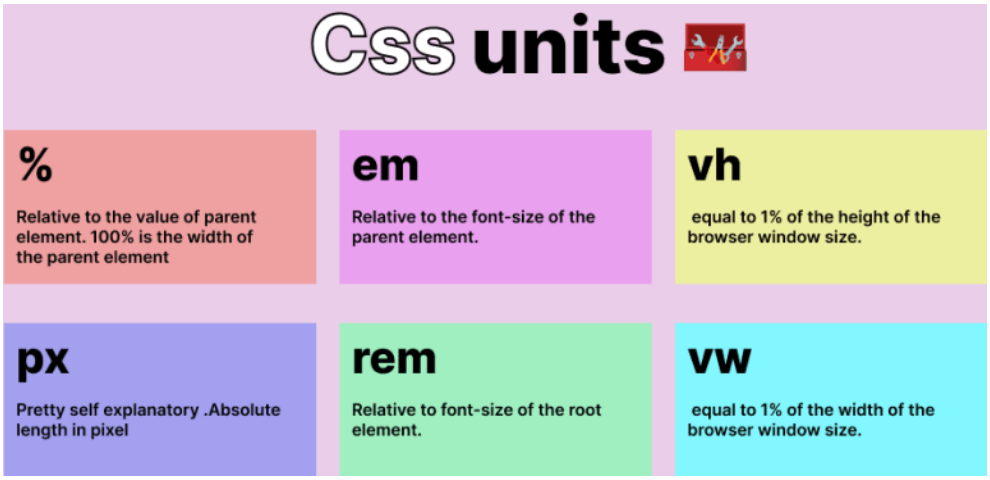
/* body{

background-color: blanchedalmond;

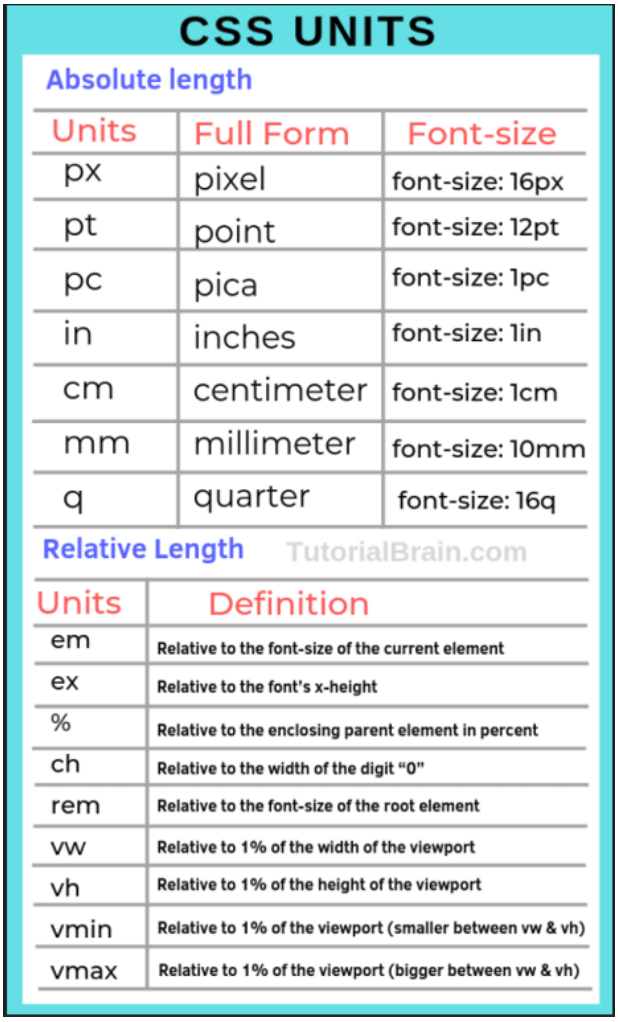
} */ single line and multiline comments are done in same way

CSS Units =>

100km here 100 is magnitude and km is unit

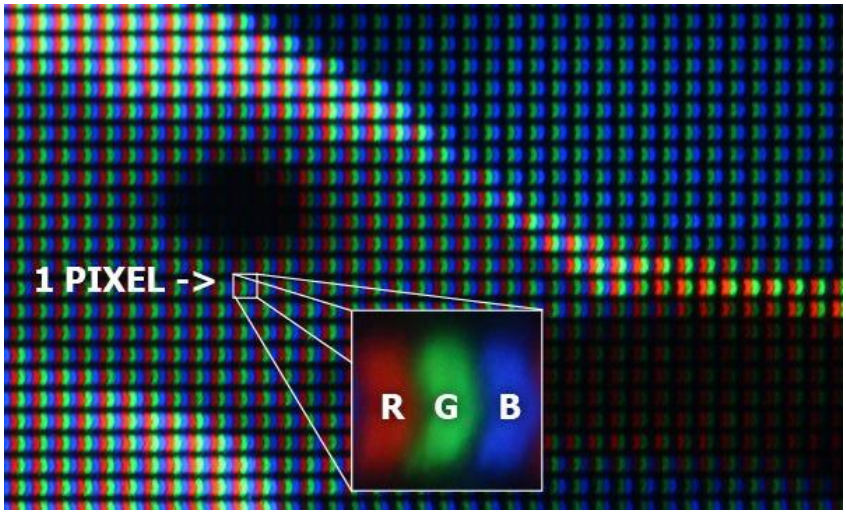


Absolute means those are fixed and relative means that unit which depends on other x factor.



PX => means pixels

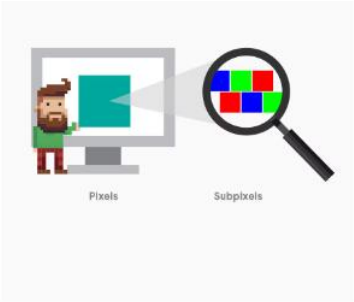
Px = 1/96 inch



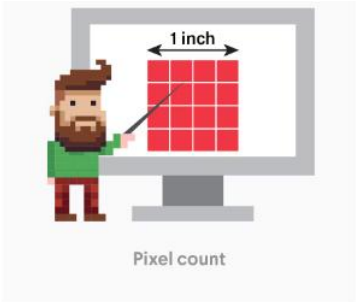
Difference between ppi and dpi => Pixels per inch(PPI) and Dots per Inch(DPI)

PPI describes the resolution in pixels of a digital image whereas DPI describes the amount of ink dots on a printed image.

Though PPI largely refers to screen display, it also affects the print size of your design and thus the quality of the output. DPI, on the other hand, has nothing to do with anything digital and primarily concerns print.

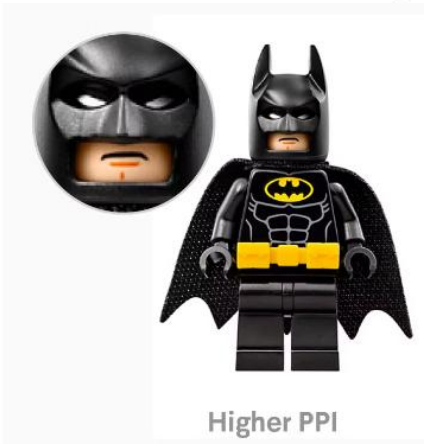


Each pixel is made up of RGB subpixels



Pixel count describe an image's dimensions based on the number of pixels

PPI, or pixel density, describes the amount of detail in an image based on the concentration of pixels




```
    <style>
        .box1{
            border: 1px solid black;
            font-size: 100px;
        }
    </style>
</head>
<body>
    <div class="box1">
        Love You sir
    </div>
```

Responsive =>

landscape(width more, height less) portrait(width less, height more)

main axis => x axis

cross axis => y axis

Unites =>

Px fixed

% How much of parent

vw, vh In the height nad width change according to screen size we are having

vmax,vmin means viewport maximum and viewport minimum

em, rem

In this we change parent(box) height then child(inner) height will not effect

```
 <style>
  *{
 margin: 0;
 padding: 0;
 box-sizing: border-box;
}
html, body{
 height: 100%;
 width: 100%;
}
#box{
 height: 300px;
 width: 300px;
 background-color: crimson;
}
 </style>
</head>
<body>
<div id="box">
</div>
</body>
</html>
```

here the box size is depend on screen size

```
    <style>
        *{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
html, body{
  height: 100%;
  width: 100%;
}
#box{
    height: 30%;
    width: 30%;       /*parent ka kitna persentage landscape screen also change it to px to change the effect*/
  background-color: crimson;
}
    </style>
</head>
<body>
<div id="box">
</div>
</body>
```

In this we change parent(box) height then child(inner) height will also change effect

```
html, body{
  height: 100%;
  width: 100%;
}
#box{
  height: 300px;
  width: 300px;
  background-color: crimson;
}
#inner{
  height: 50%;
  width: 50%;
  background-color: royalblue;
}
    </style>
</head>
<body>
<div id="box">
    <div id="inner"></div>
</div>
</body>
```

vw(viewport width) it will change when we change the width of screen, vh(viewport height) it will change when we change the height of screen.

```
}
#box{
  height: 300px;
  width: 300px;
  background-color: crimson;
}
#inner{
  height: 50%;
  width: 50vw;
  background-color: royalblue;
}
    </style>
</head>
<body>
<div id="box">
    <div id="inner"></div>
</div>
</body>
```

VW VH applying on font the size of font depends on vh and vw

```
html, body{
  height: 100%;
  width: 100%;
}
h1{
font-size: 5vw;  /* change the width and see the
effect and also see by applying px and %*/
}
</style>
</head>
<body>
    <h1> hello how are u</h1>
</body>
</html>
```

vmax(viewport maximum), vmin(viewport minimum) =>

Take example of phone and laptop and see which is maximum height and width. Why we are using vmax and vmin becz when we use vw and reduce the screen size then at point we are not able to see what is written on screen.

When we use vmax and vmin and we reduce the screen size at some point out width will become lesser then height at that point our font size will stop to reduce. Vmax means the thing which we max take that at reference like if the width is max taken take width reference if height is max then take height as reference but at a point when the max this is become less in compare of other at that point the size of font stop reducing and start wrapping. Vmin means take the reference of what is minimum width or height.

In the case of vmax the font size will change which is maximum or greater in between height and width according to device

In the case of vmax and vmin when we reduce width of device then font size will change but as we know at apont our with size is less then height size at that point due to vmax vw change into vh.

```
html, body{
  height: 100%;
  width: 100%;
}
h1{
font-size: 5vmax;
}
</style>
</head>
<body>
    <h1> hello how are u</h1>
</body>
```

Em => this unit is use when we want to give the font size to our font on the base of parent font size so for such case first we have to give font size to parent also.

By default font size is 16px and if we say child is 1em so it means 16px.

Em(element) is a unit of measurement which indicates size relative to the size of the font. 1em means "equal to the current font size", and 2em means 2 times the current font size.

```
html, body{
  height: 100%;
  width: 100%;
}
#parent{
background-color: #dadada;
width: 40%;
}
h1{
font-size: 50px;   /*In thie when we want to change the font size of any heading we have to do it manually and if we have so many headings then it will be hectic work for all of us so in such case we use em*/
}
h2{
font-size: 40px;
}
h3{
font-size: 50px;
}
</style>
</head>
<body>
    <div id="parent">
  <h1>hello how are u</h1>
  <h2>i am good what about u</h2>
  <h3>lets have fun while learning</h3>
    </div>
  </body>
```

```
#parent{
background-color: #dadada;
width: 40%;
font-size: 30px;    /*chnage the font size of parent
and see the changes*/
}
h1{
font-size: 2em;
}
h2{
font-size: 1em;
} q
h3{
font-size: 1.5em;
}
</style>
```

Rem => In this the font size of element depends on root element font size. The value of 1 rem is 16px. R means root and here root is HTML tag.

```
html, body{
  height: 100%;
  width: 100%;
}
#parent{
background-color: #dadada;
width: 40%;
font-size: 30px;    /*comment parent font size and
see the changes*/
}
h1{
font-size: 2rem;
}
h2{
font-size: 1rem;
}
h3{
font-size: 1.5rem;
}
</style>
</head>
```

We can group selectors-

h1,h2,div{

}

We can use element.class as selector-

p.one{

color:red;

}

```
 <meta name="viewport" content="width=device-width
 initial-scale=1.0">
 <link rel="stylesheet" href="style.css">
 <title>Document</title>
</head>
<body>
 <div class="container">
<p class="a">Lorem ipsum dolor sit amet.</p>

<p class="b">Lorem ipsum dolor sit amet.</p>

<p class="c">Lorem ipsum dolor sit amet.</p>

<p class="d">Lorem ipsum dolor sit amet.</p>

<p class="e">Lorem ipsum dolor sit amet.</p>
</div>
</body>
</html>
```

```
*{
    margin: 10px;
    padding: 0px;
    box-sizing: border-box;
}
html{
    width: 100%;
    height: 100%;
    font-size: 20px;
}
p.a{
    color: red;
}
p.b{
    color: aqua;
}
p.c{
    color: greenyellow;
}
p.d{
    color: pink;
}
p.e{
    color: darkcyan;
}
```

We can use universal selector

*{

Margin:0;

Padding:0;

}

Attribute =>

Id attribute- To use an ID selector in CSS, you simply write a hashtag (#) followed by the ID of the element.

#id {

// CSS property

}

```
<head>
  <title>CSS</title>
  <style>
        #g1 {
          color: green;
          text-align: center;
        }
        </style>
<body>
</head>
  <h1 id="g1">I am ID</h1>
</body>
```

Class attribute-

***.class* selector** is used to select all elements which belong to a particular class attribute. In order to select the elements with a particular class, use the period (.) character specifying the class name ie., it will match the HTML element based on the contents of their class attribute.

.class {

// CSS property

}

```
<head>
  <title>CSS</title>
  <style>
          .gs {
        color: green;
    }

        </style>
<body>
</head>
  <h1 class="gs">I am ID</h1>
</body>
</html>
```

Color property-

p{

Color:red;

}

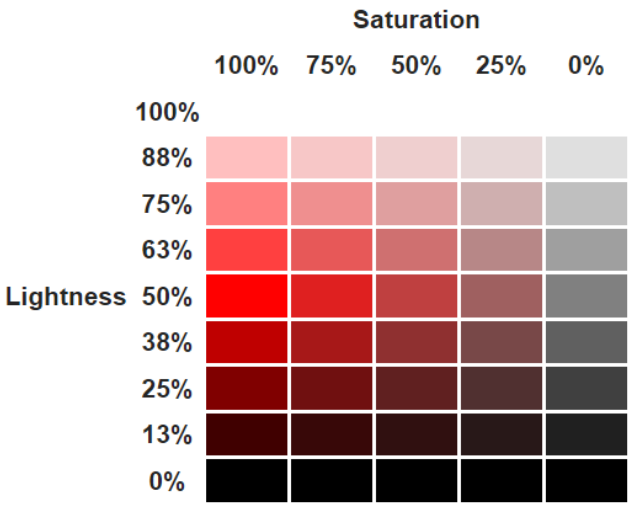
Types of color values-

rgb(100,78,98), rgpa- a stands for Aplha.

hex code - #ff7180

hsl(Hue-Saturation-Lightness)- Hue is represented as an angle of the color circle (i.e. the rainbow represented in a circle). This angle is so typically measured in degrees. Saturation is represented as percentages. 100% is full saturation, and 0% is a shade of gray. Lightness is represented as percentages. 0% lightness is black, 100% lightness is white, and 50% lightness is “normal”.

Hue	hsl(H,S,L)	Color	Color Name
0	hsl(0, 100%, 50%)		red
60	hsl(60, 100%, 50%)		
120	hsl(120, 100%, 50%)		green
180	hsl(180, 100%, 50%)		
240	hsl(240, 100%, 50%)		blue
300	hsl(300, 100%, 50%)		
360	hsl(360, 100%, 50%)		red




```
<body>
<div class="demo">
  <p class="named">Named Color</p> <!-- aqua -->
  <p class="hex">Hex Color</p> <!-- #00FFFF -->
  <p class="rgba">RGBa Color</p> <!-- rgba(0, 255, 255, .5) -->
  <p class="hsla">HSLa Color</p> <!-- hsla(180, 100%, 50%, .5) -->
  </div>
</body>
```

```
.named { color: aqua; }
.hex { color: #00FFFF; }
.rgba { color: rgba(0, 255, 255, .5); }
.hsla { color: hsla(180, 100%, 50%, .5); }
/* Styling for Pen, unrelated to color */
body {
  font-family: 'Abril Fatface', serif;
  font-size: 2.5em;
}
p {
  min-width: 50%;
  text-align: center;
  margin: .5em 0;
  text-transform: uppercase;
}
.demo {
  width: 50%;
  margin: 0 auto;
}
```