



# Learning CSS (Cascading Style Sheet)

---

**Praveen Nair**

# Cascading Style Sheet (CSS-3)

CSS describes how an HTML file should be display.

CSS is used to change the layout of an HTML page.

CSS consist of following:

1. Selector & Declaration Block
2. Property & Value

.....

H1 = {color:blue}

# Inserting inline CSS

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Document</title>
  </head>
  <body>
    <p style="color:blue">
      This is a dummy text and can be replaced.
    </p>
  </body>
</html>
```

# Inserting internal CSS

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Document</title>
    <style>
      p {
        color: blue;
      }
    </style>
  </head>
  <body>

    <p>Paragraph 1 - This is a dummy text and can be replaced</p>
    <p>Paragraph 2 - This is a dummy text and can be replaced</p>

  </body>
</html>
```

# Inserting external CSS

```
<html lang="en">
  <head>
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <p>This is a dummy text and can be replaced.</p>
  </body>
</html>
```

.....

styles.css

```
p {
  color: red;
}
```

# Order of precedence

1. Inline style
2. Internal style
3. External style
4. Browser default

Note: If external stylesheet is referred after internal block then external style will take precedence

# Select based on tag, id or class

```
p {color: red;}
```

```
#p1{color: blue;}
```

```
.c1 {color: green;}
```

```
.c2 {color: yellow;}
```

```
p.c1 {color: maroon;}
```

# Refer more than one class

```
.c1 {color: green;}
```

```
.c2 {text-align: center;}
```

```
#p1{color: blue;}
```

```
p {color: red;}
```

```
.....
```

```
<p class="c1 c2" id=p1>This is dummy text</p>
```

```
.....
```

Id selector takes the first precedence and tag is last precedence



# Universal – affects every element

```
. * {  
  text-align: center;  
  color: blue;  
}
```

# Grouping selectors

```
p, h1, div {  
  color: red;  
}
```

# CSS Comments

```
/* This is a single-line comment */  
H1 {  
  color: blue;  
}
```

# Colors and Backgrounds

```
div {  
  background-color: lightblue;  
  color: green;  
}
```

# Borders and Margins

```
div {  
    border: dashed 10px blue /*shorthand*/  
}
```

```
div {  
    width: 100px;  
    border: 5px blue solid;  
    margin: 40px 40px 40px 40px; /* try auto and margin-top */  
}
```

# Text, Fonts

- `div {`
- `font-family: Arial;`
- `font-size: 14px;`
- `}`

# Position fixed

```
div {  
    position: fixed;  
    bottom: 0;  
    right: 0;  
    width: 500px;  
    border: 10px solid blue;  
}
```

# Position sticky

```
div {  
    position: sticky;  
    top: 0;  
    background-color: green;  
    border: 2px solid #4caf50;  
}
```



# Padding

```
div {  
    border:5px blue solid;  
    padding: 10px 20px 30px 40px;  
}
```

.....

Try with just padding:10px and  
Padding-top, padding-bottom, etc

# Background-image

```
body {  
    background-image: url("colors.png");  
    background-repeat: no-repeat; /* repeat */  
    background-position: center; /* right top, left top */  
    background-attachment: scroll; /*fixed*/  
    background-size: cover;  
}
```

```
.....  
body {background: #eeeeee url("colors.png") no-repeat right top;}
```

```
.....  
background-color, background-image, background-repeat, background-attachment,  
background-position
```

# Flex box

```
<!DOCTYPE html>
<html>
<head>
<style>
.container {
  display: flex;
  height: 100px;
  /*flex-direction: row;*/ /* try column */
  background-color: silver;
  /* flex-wrap: wrap; */
  /* flex-flow: row wrap; */
  /* justify-content: right; center left flex-start flex-end space-around space-between */
  /* align-items: center; flex-start flex-end stretch(default) */
  /* align-content: flex-start; */
}

.container > div {
  background-color: orange;
  width: 100px;
  margin: 10px;
  text-align: center;
  line-height: 75px;
  font-size: 20px;
}
</style>
</head>
<body>
<div class="container">
  <div>HTML</div>
  <div>CSS</div>
  <div>JavaScript</div>
</div>
</body>
</html>
```

# Grid

```
<!DOCTYPE html>
<html>
<head>
<style>
.container {
  display: grid; /*inline-grid */
  background-color: silver;
  grid-template-columns: auto auto auto;
  column-gap: 20px;
  row-gap: 30px;
  /* gap: 20px 30px; for both row and col */
}
.container > div {
  background-color: orange;
  width: 100px;
  margin: 10px;
  text-align: center;
  line-height: 75px;
  font-size: 20px;
}
</style>
</head>
<body>
<div class="container">
  <div>HTML</div>
  <div>CSS</div>
  <div>JavaScript</div>
  <div>HTML</div>
  <div>CSS</div>
  <div>JavaScript</div>
  <div>HTML</div>
  <div>CSS</div>
  <div>JavaScript</div>
</div>
</body>
</html>
```

# Styling Lists

```
<style>
  li {border: solid;}
  ul {
    border: solid;
    list-style-type: square; /* try circle upper-roman lower-alpha */
    /* list-style-image: url('colors.png'); */
    list-style-position: outside; /* try inside */
    margin: 20px;
    padding: 30px;}
</style>
```

# Box model

```
div {  
  width: 300px;  
  border: 15px solid green;  
  padding: 50px;  
  margin: 20px;  
}
```

Content > padding > border > margin

# Border Boxing

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      .c1 {
        width: 100px;
        border: 1px solid black;
        /* box-sizing: border-box; */
      }
      .c2 {
        width: 100px;
        padding: 10px; /*width becomes 120px */
        border: 1px solid orange;
        /* box-sizing: border-box; */
      }
    </style>
  </head>
  <body>
    <div class="c1">Hello World</div>
    <div class="c2">Hello World</div>
  </body>
</html>
```

# Pseudo class

`a:link {color: green;}`

`a:visited {color: red;}`

`a:hover {color: orange;}`

`a:active {color: blue;}`

`div:hover {display: block; background-color: yellow;}`

`input:focus {display: block; background-color: pink;}`

`input:in-range {border: 2px solid red;}`



# Form styling

```
input {  
  width: 100%;  
}
```

```
input[type=text] {  
  width: 100%;  
  padding: 12px 20px;  
}
```

```
input[type=button], input[type=submit], input[type=reset] {  
  background-color: #04AA6D;  
  border: none;  
  color: white;  
}
```

# Wildcard Selectors (\*, ^ and \$)

The asterisk (\*) matches any character, the caret (^) matches the start, and the dollar sign (\$) matches the end of an attribute value.

```
.....  
<style>  
  [class*="t"] {  
    background-color: yellow;  
  }  
  [class^="t"] {  
    background-color: yellow;  
  }  
  [class$="t"] {  
    background-color: yellow;  
  }  
</style>  
  
<body>  
  <div class="tree">This is tree class</div>  
  <div class="stroke">This is stroke class</div>  
  <div class="best">This is best class</div>  
</body>
```

# Float

```
div{  
    float:right /* left none */  
}
```

# Styling Tables

```
table,th,td {  
    border: 1px solid; /* try border-bottom */  
    border-collapse: collapse;  
    text-align: center;  
    vertical-align: middle; /*try bottom and top */  
    padding:20px;  
}  
  
tr:hover {background-color: orange;}  
thtr:nth-child(even) {background-color: cyan;}  
{width:100px; /* try percent also */}
```

# Gradients

```
div {  
    /*background-image: linear-gradient(maroon, yellow);*/  
    /*background-image: linear-gradient(to right,red,pink);*/  
    /*background-image: linear-gradient(to bottom right,maroon, yellow);*/  
    /*background-image: linear-gradient(90deg,maroon, yellow);*/  
    /*background-image: linear-gradient(maroon, yellow,pink);*/  
    /*background-image: radial-gradient(red, yellow, green);*/  
    /*background-image: radial-gradient(maroon 10%, yellow 20%, green 30%);*/  
    /*background-image: radial-gradient(circle,red, pink, green);*/  
    background-image: repeating-radial-gradient(maroon, pink 10%, green 15%);  
}
```