CSS

- CSS was developed in year 1997
- CSS allows a web developers to define look and feel (style) of web pages
- It was intended to allow developers to separate content from design

What CSS can do?

- Change the font style
- Apply background colors / images
- Change the position of elements
- Apply animations
- Any other style rules are defined here

Types of CSS ?

1. Inline
    a. The stylerule is applied for specific tag
2. Internal stylesheet (embedded stylesheet)
    a. The stylerule is applied for all the tags, within the page
3. External stylesheet (used in real time)
    a. The stylerule is applied for the pages

What CSS page contain
selector {
property: value;
property: value;
}

How to Call CSS from HTML

<link rel="stylesheet" type="text/css" href="style.css" />

CSS Selectors

Element	P
Id	#id
Class	.class
Position	Div a

#col {color:blue;}

Any Tag containing attribute Id colored Blue

.col {color:blue;}

Any Tag containing attribute class colored Blue

P.col {color:blue;}

Any P Tag containing attribute class colored Blue

P[name] {color:blue;} **(only CSS3)**

Any P Tag containing attribute name colored Blue

P[name=”King”] {color:blue;} **(only CSS3)**

Any P Tag containing attribute name with value “King” colored Blue

P[name^=”King”] {color:blue;} **(only CSS3)**

Any P Tag containing attribute name with value starts with “King” colored Blue

P[name$=”King”] {color:blue;} **(only CSS3)**

Any P Tag containing attribute name with value ends with “King” colored Blue

P[name*=”King”] {color:blue;} **(only CSS3)**

Any P Tag containing attribute name with value contains “King” colored Blue

Pseudo Classes

P:nth-Child(3) {color:blue;} **(only CSS3)**

Third Child Coloured Blue
Eg:

```
<div>
<p>first</p>
<p>second</p>
<p>third</p>   -> colored Blue
<p>fourth</p>
<p>fifth</p>
</div>
```

P:nth-Child(odd) {color:blue;} **(only CSS3)**

Every Odd Elements colored Blue
Eg:

```
<div>
<p>first</p>    -> colored blue
<p>second</p>
<p>third</p>   -> colored Blue
<p>fourth</p>
<p>fifth</p>  -> colored Blue
</div>
```

P:nth-Child(odd) {color:blue;} **(only CSS3)**
P:nth-Child(Even) {color:Red;}

Every Odd Elements colored Blue and Even in Red
Eg:

```
<div>
<p>first</p>    -> colored blue
<p>second</p>  -> Red
<p>third</p>   -> colored Blue
```

fourth ->Red
fifth -> colored Blue

Negation Pseudo class

*{color:blue}

Every Elements colored Blue

:not(.king) {color:blue}

Any Tag where class is not equal to king colored blue
Eg:

first -> Blue
second
third ->blue
fourth ->blue
fifth

Div > P {color:blue;}

Any Paragraph Where Div as a parent will be colored blue

P.king + p {color:blue;}

Any p tag with class king and immediately P as its parent color blue

P.king ~ p {color:blue;}

Any p tag with class king and not immediately P as its parent color blue

Working with Following Types in CSS

1. Colors
2. Backgroundcolor
3. Background-Images
4. Font
5. Text
6. Borders
7. Margin
8. Padding
9. Position
10. Round Borders
11. Transitions

Color :

Color	<color>
background-color	transparent <color>
background-image	none url(<URL>)
background-repeat	repeat repeat-x repeat-y no-repeat
background-attachment	scroll fixed
Background-style	Cover

You can define colors with the use of RGB values, by simply entering the values for amounts of Red, Green and Blue.

background-color:rgb(51,51,102);

Font

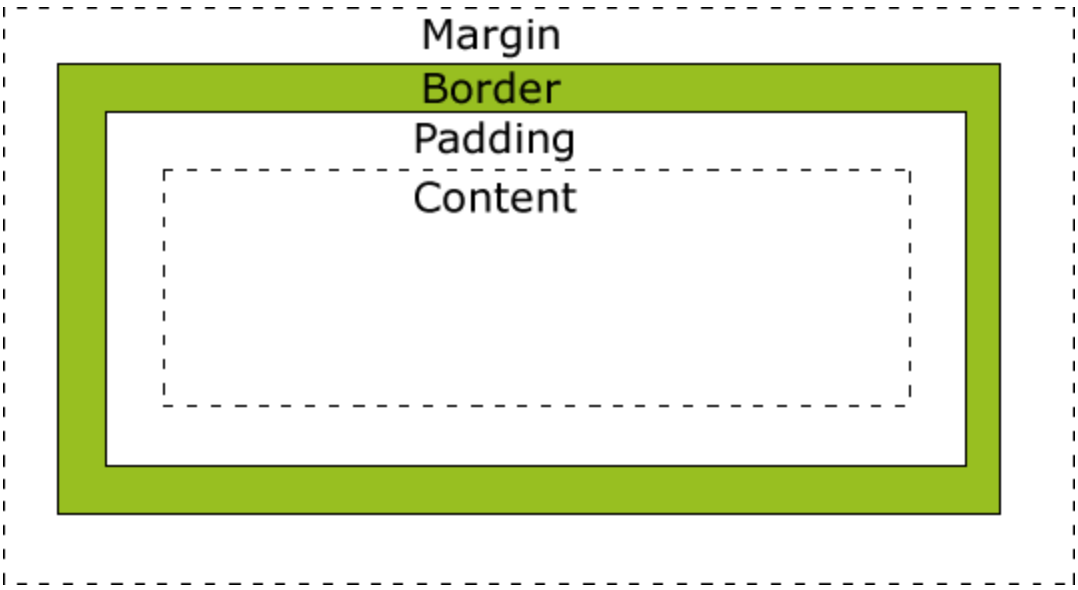
font-size: larger; (small,smaller,medium,large,x-large,xx-large.)

font-weight: bold; font-style: italic; font-size: 10px; font-family: sans-serif;	font: italic bold 10px sans-serif times;

Text

1. text-align: center;
2. text-decoration: underline;
3. text-transform :uppercase
4. text-transform :lowercase

CSS Box Model



Link Properties

A:link A:visited A:active A:hover	<style> <style> <style> <style>

A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline; color: red;}

Border

```
color:red;
border-width:2px;
border-color:solid green;
border-style:solid
```

or

```
border-top-style: dotted;
  border-right-style: solid;
  border-bottom-style: dotted;
  border-left-style: solid;
```

border types

dotted: Defines a dotted border

dashed: Defines a dashed border

solid: Defines a solid border

double: Defines two borders. The width of the two borders are the same as the border-width value

groove: Defines a 3D grooved border. The effect depends on the border-color value

ridge: Defines a 3D ridged border. The effect depends on the border-color value

inset: Defines a 3D inset border. The effect depends on the border-color value

outset: Defines a 3D outset border. The effect depends on the border-color value

margin

margin-left:50px; margin-top:50px; margin-right:150px;	margin:10 50 110 10; (top right bottom left)

Padding:

```
padding-left:100px;
padding-top:100px;
padding-bottom:50px;
```

Radius

-webkit-border-radius:25px;

BoxShadow

```
webkit-box-shadow:rgb(110,110,110)8px 8px;
```

Transform

```
-webkit-transform:scale(1.5);
-webkit-transform:rotate(45deg);
-webkit-transform:skew(45deg);
webkit-transition:-webkit-transform 2s;
```

Animations

```
animation:myfirst 5s;
@-webkit-keyframes myfirst
{
from {background:red;}
to {background:yellow;}
}
@-webkit-keyframes myfirst
{
0%   {background:red; left:0px; top:0px;}
25%  {background:yellow; left:200px; top:0px;}
50%  {background:blue; left:200px; top:200px;}
75%  {background:green; left:0px; top:200px;}
100% {background:red; left:0px; top:0px;}
}
```

```
-webkit-animation-iteration-count:infinite;

background:-webkit-linear-gradient(top,black,white);
```

Print

```
<link rel='stylesheet' media='all' href='a.css' />
<link rel='stylesheet' media='print' href='b.css' />
<link rel='stylesheet' media='screen and (min-width: 701px)' href='c.css' />



<html>

<body>
<div>Hello media</div>

</body>

</html>
```