## What is CSS?

**CSS stands for cascading style sheets. CSS is used to for styling the webpages such as we can add colours, fonts etc to the web pages.**

**Cascading Style Sheets** (**CSS**) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.

## CSS Syntax

H1{

**Background-color:blue;**

**Color:white;**

**}**

H1 is Selector, Background-color is the property and blue is the value.

**Selector** − A selector is an HTML tag at which a style will be applied. This could be any tag like <h1> or <table> etc.

**Property** − A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be *color*, *border* etc.

**Value** − Values are assigned to properties. For example, *color* property can have value either *red* or *#F1F1F1* etc.

## CSS Selectors

Selectors are used to style the html elements.

## 1)Element Selector

The **element selector** in CSS is used to select **elements** inside the **elements**

**Example**

P{

Font-size:15px;

Color:#666;

Text-align:center;

}

Suppose if we write above css styles for the paragraph all the <p> elements in the web page will take the styles of p like the paragraph elements text will be displayed in center with font-size:15px and color grey.

## 2) The CSS id Selector

The id selector uses the id attribute of an HTML element to select a specific element.

The id of an element is unique within a page, so the id selector is used to select one unique element!

To select an element with a specific id, write a hash (#) character, followed by the id of the element.

**Example**

The CSS rule below will be applied to the HTML element with id="para1":

#para1 {  
  text-align: center;  
  color: red;  
}

## The CSS class Selector

The class selector selects HTML elements with a specific class attribute.

To select elements with a specific class, write a period (.) character, followed by the class name.

**Example**

In this example all HTML elements with class=para1 will be red and center aligned text

.para1 {  
  text-align: center;  
  color: red;  
}

## The CSS Universal Selector

The universal selector (\*) selects all HTML elements on the page.

**Example**

The CSS rule below will affect every HTML element on the page:

\* {  
  text-align: center;  
  color: blue;  
}

## The CSS Grouping Selector

The grouping selector selects all the HTML elements with the same style definitions.Let us take

**Example**

The CSS rule below will affect every HTML element on the page:

h1 {  
  margin: 0px;  
  padding: 0px;  
}  
  
h2 {  
  margin: 0px;  
  padding: 0px;  
}  
  
p {  
 margin: 0px;  
  padding: 0px;  
}

It will be better to group the selectors, to minimize the code.

To group selectors, separate each selector with a comma.

h1, h2, p {  
margin: 0px;  
  padding: 0px;  
}

## Three Ways to Insert CSS

**Types of CSS**

There are 3 types of CSS. We can add CSS to our web pages in 3 ways

**1)Inline** - by using the style attribute inside HTML elements

**2)Internal** - by using a <style> element in the <head> section

**3)External** - by using a <link> element to link to an external CSS file

## Inline CSS

An inline CSS uses the style attribute of an HTML element.

**An inline CSS is used to apply a unique style to a single HTML element.**

The following example sets the background color of the <h1> element to orange, and the text color of the <p> element to purple by using style attribute

**Example:**

<h1 style="background-color:orange;">A Blue Heading</h1>  
  
<p style="color:purple;">A red paragraph.</p>

## Internal CSS

**an internal CSS is used to define a style for a single HTML page.**

**An internal CSS is defined in the <head> section of an HTML page, within a <style> element.**

The following example sets the text color of ALL the <h1> elements on the page to blue, and the text color of ALL the <p> elements to red.

**Example:**

<head>  
<style>  
 h1{color: blue;}  
 p{color: red;}  
</style>  
</head>

## External CSS

An external style sheet is used to define the style for many HTML pages.

To use an external style sheet, add a link to it in the <head> section of each HTML page:

**Example:**

<head>  
  <link rel="stylesheet" href="styles.css">  
</head>

**Note:** rel attribute defines the relationship between styles.css and html page

## CSS Comments

Comments are used to explain the code, and may help when you edit the source code at a later date.

Comments are ignored by browsers.

A CSS comment is placed inside the <style> element, and starts with /\* and ends with \*/:

**Example**

/\* This is a single-line comment \*/  
p {  
  color: red;  
}

## CSS Backgrounds

The CSS background properties are used to define the background effects for elements.

In these chapters, you will learn about the following CSS background properties:

* **background-color** - Specifies the background color to be used
* **background-image** – is used to give background image to the HTML Elements
* **background-repeat** - Specifies how to repeat the background images
* **background-attachment**- Specifies whether the background images are fixed or scrolls with the rest of the page
* **background-position** - Specifies the background color to be used
* **background-size** - Specifies the size of the background images
* **background-origin** - Specifies the positioning area of the background images
* **background-clip** - Specifies the painting area of the background images

**background-origin**: padding-box|border-box|content-box|initial|inherit;

**background-clip**: padding-box|border-box|content-box|initial|inherit;

## Parallax Section

By using background-attachment:fixed property we can create parallax section

**Opacity / Transparency**

The opacity property specifies the opacity/transparency of an element. It can take a value from 0.0 - 1.0. The lower value, the more transparent:

## div {   background-color: green;   opacity: 0.3; }

## Transparency using RGBA

## If you do not want to apply opacity to child elements, like in our example above, use ****RGBA**** color values. The following example sets the opacity for the background color and not the text:

**Example**

div {  
  background: rgba(0, 128, 0, 0.3) /\* Green background with 30% opacity \*/  
}

## CSS background-position

The background-position property is used to specify the position of the background image.

### **Example**

Position the background image in the top-right corner:

body {  
  background-image: url("img\_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
}

## CSS background-attachment

The background-attachment property specifies whether the background image should scroll or be fixed (will not scroll with the rest of the page):

### **Example**

Specify that the background image should be fixed:

body {  
  background-image: url("img\_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
  background-attachment: fixed/scroll;  
}

Specify that the background image should scroll with the rest of the page:

body {  
  background-image: url("img\_tree.png");  
  background-repeat: no-repeat;  
  background-position: right top;  
  background-attachment: scroll;  
}

## CSS background - Shorthand property

To shorten the code, it is also possible to specify all the background properties in one single property. This is called a shorthand property.

Instead of writing:

body {  
  background-color: #ffffff;  
  background-image: url("images/imagename.jpg");  
  background-repeat: no-repeat;  
  background-position: right top;  
}

You can use the shorthand property background:

### **Example**

Use the shorthand property to set the background properties in one declaration:

body {  
  background: #ffffff url("img\_tree.png") no-repeat right top;  
}

When using the shorthand property the order of the property values is:

* background-color
* background-image
* background-repeat
* background-attachment
* background-position

# **CSS Borders**

## CSS Border Properties

The CSS border properties allow you to specify the style, width, and color of an element's border.

## CSS Border Style

The border-style property specifies what kind of border to display.

The following values are allowed:

* dotted - Defines a dotted border
* dashed - Defines a dashed border
* solid - Defines a solid border
* double - Defines a double border
* 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
* none - Defines no border
* hidden - Defines a hidden border

The border-style property can have from one to four values (for the top border, right border, bottom border, and the left border).

### **Example**

p.dotted {border-style: dotted;}  
p.dashed {border-style: dashed;}  
p.solid {border-style: solid;}  
p.double {border-style: double;}  
p.groove {border-style: groove;}  
p.ridge {border-style: ridge;}  
p.inset {border-style: inset;}  
p.outset {border-style: outset;}  
p.none {border-style: none;}  
p.hidden {border-style: hidden;}  
p.mix {border-style: dotted dashed solid double;}

## CSS Border Width

The border-width property specifies the width of the four borders.

The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick:

p.one {  
  border-style: solid;  
  border-width: 5px;  
}  
  
p.two {  
  border-style: solid;  
  border-width: medium;  
}  
  
p.three {  
  border-style: dotted;  
  border-width: 2px;  
}  
  
p.four {  
  border-style: dotted;  
  border-width: thick;  
}

## Specific Side Widths

The border-width property can have from one to four values (for the top border, right border, bottom border, and the left border):

### **Example**

p {  
  border-style: solid;  
  border-width: 5px 20px; /\* 5px top and bottom, 20px on the sides \*/  
}  
  
p {  
  border-style: solid;  
  border-width: 20px 5px; /\* 20px top and bottom, 5px on the sides \*/  
}  
  
p.three {  
  border-style: solid;  
  border-width: 25px 10px 4px 35px; /\* 25px top, 10px right, 4px bottom and 35px left \*/  
}

## CSS Border Color

The border-color property is used to set the color of the four borders.

The color can be set by:

* name - specify a color name, like "red"
* HEX - specify a HEX value, like "#ff0000"
* RGB - specify a RGB value, like "rgb(255,0,0)"
* HSL - specify a HSL value, like "hsl(0, 100%, 50%)"
* transparent

**Note:** If border-color is not set, it inherits the color of the element.

### **Example**

Demonstration of the different border colors:

p {  
  border-style: solid;  
  border-color: red;

}

## Specific Side Colors

The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border).

### **Example**

p {  
  border-style: solid;  
  border-color: red green blue yellow; /\* red top, green right, blue bottom and yellow left \*/  
}

## HEX Values

The color of the border can also be specified using a hexadecimal value (HEX):

### **Example**

p.one {  
  border-style: solid;  
  border-color: #ff0000; /\* red \*/  
}

## RGB Values

Or by using RGB values:

### **Example**

p.one {  
  border-style: solid;  
  border-color: rgb(255, 0, 0); /\* red \*/  
}

## HSL Values

You can also use HSL values:

### **Example**

p.one {  
  border-style: solid;  
  border-color: hsl(0, 100%, 50%); /\* red \*/  
}

# **CSS Border Sides**

## CSS Border - Individual Sides

In CSS, there are also properties for specifying each of the borders (top, right, bottom, and left):

**Example**

p {  
  border-top-style: dotted;  
  border-right-style: solid;  
  border-bottom-style: dotted;  
  border-left-style: solid;  
}

## CSS Border - Shorthand Property

Like you saw in the previous page, there are many properties to consider when dealing with borders.

To shorten the code, it is also possible to specify all the individual border properties in one property.

The border property is a shorthand property for the following individual border properties:

* border-width
* border-style (required)
* border-color

### **Example**

p {  
  border: 5px solid red;  
}

You can also specify all the individual border properties for just one side:

### **Left Border**

p {  
  border-left: 6px solid red;  
  background-color: lightgrey;  
}

### **Bottom Border**

p {  
  border-bottom: 6px solid red;  
  background-color: lightgrey;  
}

## CSS Rounded Borders

The border-radius property is used to add rounded borders to an element:

### **Example**

p {  
  border: 2px solid red;  
  border-radius: 5px;  
}