COURSE NAME: UI Web Development **Course Code: MR23-1CS0102**

UNIT-II
CASCADING STYLE SHEETS

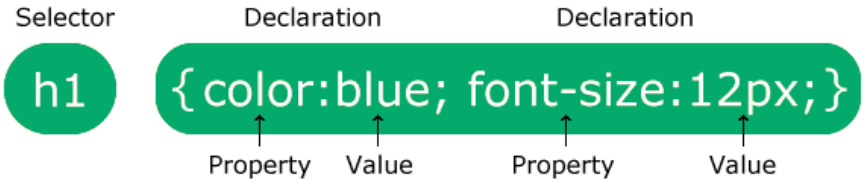
CSS (Cascading Style Sheets) is used to styles web pages. Cascading Style Sheets are fondly referred to as CSS. The reason for using this is to simplify the process of making web pages presentable. It allows you to apply styles on web pages. More importantly, it enables you to do this independently of the HTML that makes up each web page.

Why learn CSS?

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. For knowing Web Development, learning CSS is mandatory.

CSS Syntax:

A CSS rule consists of a selector and a declaration block.



- The selector points to the HTML element you want to style.
- The declaration block contains one or more declarations separated by semicolons.
- Each declaration includes a CSS property name and a value, separated by a colon.
- Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

Types of CSS:

There are three types of CSS which are given below:

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

2. **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.
3. **Inline:** Inline CSS contains the CSS property in the body section attached with the element known as inline CSS.

1. **External style sheets:** In this styles are added to the webpage by linking *.css* file to the *.html* file using the <link> tag

.css file:

```
body { background-color:pink}
h1{color:white; text-align:center}
```

.html file:

```
<html>
<head>
<link rel="stylesheet" href="external.css">
</head>
<body>
<h1> This is sample code for external CSS</h1>
</body>
</html>
```

2. **Internal Style sheets:** To apply the styles, <style> tag is inserted inside the <head> tag

```
<html>
<head>
<style>
body {background-color: blue}
h1 {color: black}
</style>
</head>
<body>
<h1> THIS IS THE SAMPLE CODE FOR INTERNAL CSS</h1>
</body>
</html>
```

3. **Inline style sheets:** In this style attributes are added inside the selector tags

```
<html>
<body style="background-color:green">
<h1 style="color: white; text-align:center"> THIS IS THE SAMPLE CODE FOR INLINE CSS</h1>
<p style="color:yellow"> Welcome to the Introduction class of CSS</p>
</body>
</html>
```

CSS Selectors: CSS selectors are used to "find" (or select) the HTML elements you want to style. The types of CSS selectors are:

1. Element Selector (h1,p, h2 etc…)
2. ID Selector (#)
3. Class Selector (.)
4. Universal Selector(*)
5. Group Selector

1. **Element Selector:** The element selector selects HTML elements based on the element name.

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  text-align: center;
  color: red;
}
</style>
</head>
<body>
<h1> ELEMENT SELECTORS </h1>
<p> This code explains the Element selectors </p>
</body>
</html>
```

2. **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.
- An id name cannot start with a number!

```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
  background-color: blue;
  color: white;
}
</style>
</head>
<body>
<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>
</body>
</html>
```

3. Class Selector (.):

- The class selector is used to specify a style for a group of elements unlike the id selector.
- This allows you to set a particular style for many Html elements with the same class.
- The class selector uses the Html class attribute, and is defined with a “.”

```
<!DOCTYPE html>
<html>
<head>
<style>
.intro {
  background-color: yellow;
}
</style>
</head>
<body>
<h1>Demo of the .class selector</h1>
<div class="intro">
  <p>This code explains class selector</p>
```

```
  <p>Class seletor must start with "."</p>
</div>

<p>This line is not styled with class</p>
<p class="intro">This line is styled with class.</p>
</body>
</html>
```

4. **Universal Selector (*):** The universal selector is used to selects all the elements on the pages. The * selector selects all elements. The * selector can also select all elements inside another element.

```
<!DOCTYPE html>
<html>
<head>
<style>
 * {  color: blue; }
#f1{ background-color: yellow; color:red}
</style>
</head>
<body>
<h1>Demo of the * selector</h1>

  <p id="f1">The Universal Selector selects all elements</p>
  <p>The * selector can also select all elements inside another element </p>
<p>The * selector is used to set yellow background</p>
</body>
</html>
```

5. **Grouping Selector:** The grouping selector selects all the HTML elements with the same style definitions.

Look at the following CSS code (the h1, h2, and p elements have the same style definitions)

Better to group the selectors, to minimize the code.

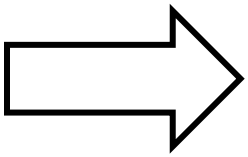
To group selectors, each selector is seperated with a comma.

```
h1 {
  text-align: center;
  color: red;
}

h2 {
  text-align: center;
  color: red;
}

p {
  text-align: center;
  color: red;
}
```

After Grouping



```
h1, h2, p {
  text-align: center;
  color: red;
}
```

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 COLORS: Colors are specified using the predefined

- color names
- RGB, HEX, HSL, RGBA, HSLA values.

CSS uses Hexadecimal or Numerical values to specify a color. These are used to set the color of the background or foreground of an Html element. Colors in CSS can be specified in various formats.

Following table tells you all possible formats:

Format	Syntax	Example
Hex Code	#RRGGBB	p {color:#FF0000;}
Short Hex Code	#RGB	p {color:#6A7;}
RGB %	rgb(rrr%,ggg%,bbb%)	p{color:rgb(50%,50%,50%);}
RGB Absolute	rgb(rrr,ggg,bbb)	p {color:rgb(0,0,255);}
keyword	aqua, black, etc.	p {color:teal;}

CSS Colors - Hex Codes

- A hexadecimal is a 6 digit representation of a color.
- First two digits(RR) represent a red value.
- Next two digits(GG) represent a green value.
- Last two digits(BB) represent a blue value.
- A hexadecimal value can be taken from any graphics software like Adobe Photoshop, Jasc Paintshop Pro or even using Advanced Paint Brush.
- Each hexadecimal code will be preceded by a hash sign #.
- Examples to use Hexadecimal notation. #66AA77, #4488DD

Syntax: background: #FFF000;

Following are the examples to use Hexadecimal notation

Color	Color HEX
	#000000
	#FF0000
	#00FF00
	#0000FF
	#FFFF00
	#00FFFF
	#FF00FF
	#C0C0C0
	#FFFFFF

CSS Colors - Short Hex Codes:

- This is a shorter form of the six-digit notation. In this format, each digit is replicated to arrive at an equivalent six-digit value; For example: #6A7 becomes #66AA77.

- A hexadecimal value can be taken from any graphics software like Adobe Photoshop, Jasc Paintshop Pro or even using Advanced Paint Brush.
- Each hexadecimal code will be preceded by a pound or hash sign #.
- Examples to use Hexadecimal notation. #6A7, #48D

Syntax: background: #FFF;

Following are the examples to use Hexadecimal notation:

Color	Color HEX
	#000
	#F00
	#0F0
	#0FF
	#FF0
	#0FF
	#F0F
	#FFF

CSS Colors - RGB Values:

This color value is specified using the rgb( ) property. This property takes three values, one each for red, green, and blue. The value can be an integer between 0 and 255 or a percentage.

Note: All the browsers does not support rgb() property of color so it is recommended not to use it.

Syntax: background:rgb(142,54,76);

Following is the example to show few colors using RGB values:

Color	Color RGB
	rgb0, 0, 0
	rgb255, 0, 0
	rgb0, 255, 0
	rgb0, 0, 255
	rgb255, 255, 0
	rgb0, 255, 255
	rgb255, 0, 255
	rgb192, 192, 192
	rgb255, 255, 255

CSS Background:

The CSS background properties are used to add background effects for elements.

CSS background properties:

1. background-color
2. background-image
3. background-repeat
4. background-attachment
5. background-position
6. background (shorthand property)

1. CSS background-color:

The background-image property specifies an image to use as the background of an element. By default, the image is repeated so it covers the entire element.

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: lightblue;
}
</style>
</head>
<body>
<h1>Hello World!</h1>
<p>This page has a light blue background color!</p>
</body>
</html>
```

2. CSS background-image:

The background-image property specifies an image to use as the background of an element. By default, the image is repeated so it covers the entire element. Following are the attributes of background-image:

a. background-image: url("C:\\user\\desktop\\image.jpg")

b. background-repeat: no-repeat /repeat(default)

c. background-attachment: fixed /scroll(default)

d. background-position: left/ right/ center/ left top/ left bottom/ right top/ right bottom

```
<!DOCTYPE html>
<html>
<head>
<style>
body {   background-image: url("img1.jpg"); color:black }
</style>
</head>
<body>
<h1>Hello!!!! Good Morning</h1>
<p>This page has an image as the background!</p>
</body>
</html>
```

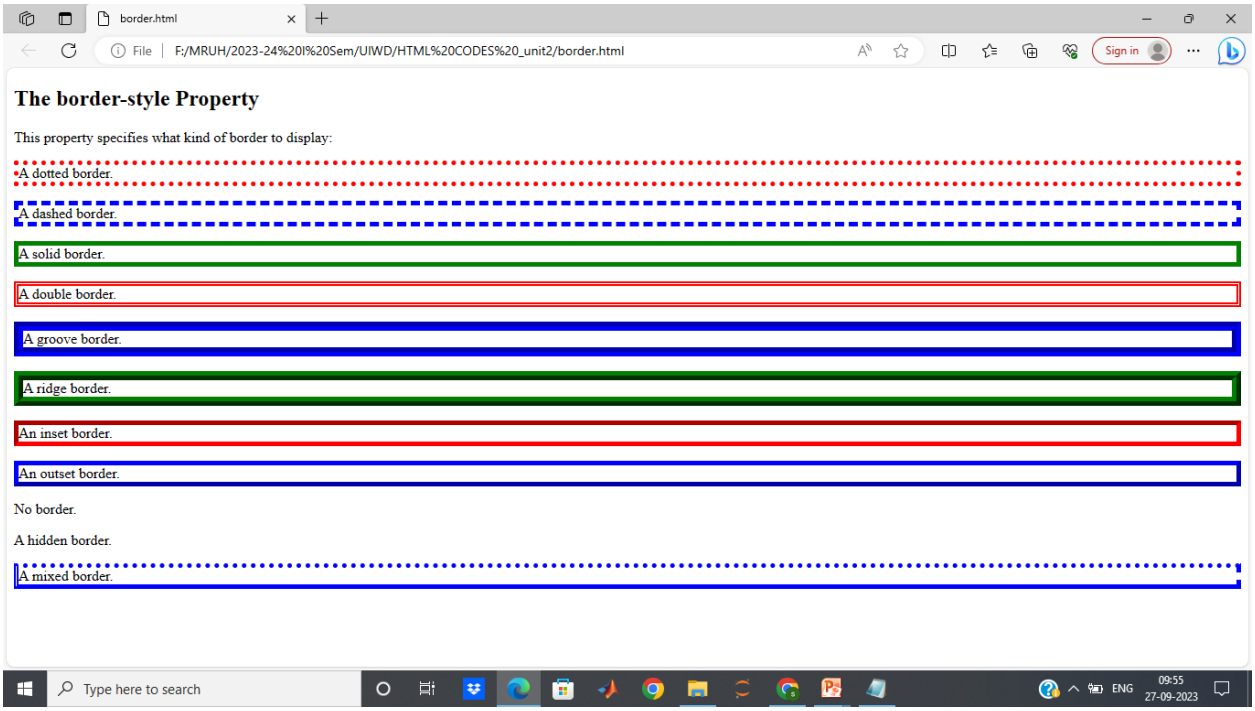
CSS Border: The border-style property specifies what kind of border to display.

- 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).

```
<html>
<head>
<style>
p.dotted {border-style: dotted; border-width: 5px; border-color: red}
p.dashed {border-style: dashed; border-width: 5px; border-color:blue }
p.solid {border-style: solid; border-width: 5px; border-color: green}
p.double {border-style: double; border-width: 5px; border-color: red}
p.groove {border-style: groove; border-width: 10px; border-color:blue}
p.ridge {border-style: ridge; border-width: 10px; border-color:green}
p.inset {border-style: inset; border-width: 5px; border-color:red}
p.outset {border-style: outset; border-width: 5px; border-color:blue}
p.none {border-style: none; border-width: 5px;border-color:green}
p.hidden {border-style: hidden; border-width: 5px; border-color:red}
p.mix {border-style: dotted dashed solid double; border-width: 5px; border-color:blue}
</style>
</head>
<body>
<h2>The border-style Property</h2>
<p>This property specifies what kind of border to display:</p>
<p class="dotted">A dotted border.</p>
<p class="dashed">A dashed border.</p>
<p class="solid">A solid border.</p>
<p class="double">A double border.</p>
<p class="groove">A groove border.</p>
<p class="ridge">A ridge border.</p>
<p class="inset">An inset border.</p>
<p class="outset">An outset border.</p>
<p class="none">No border.</p>
<p class="hidden">A hidden border.</p>
<p class="mix">A mixed border.</p>
</body>
</html>
```



CSS Rounded Borders:

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

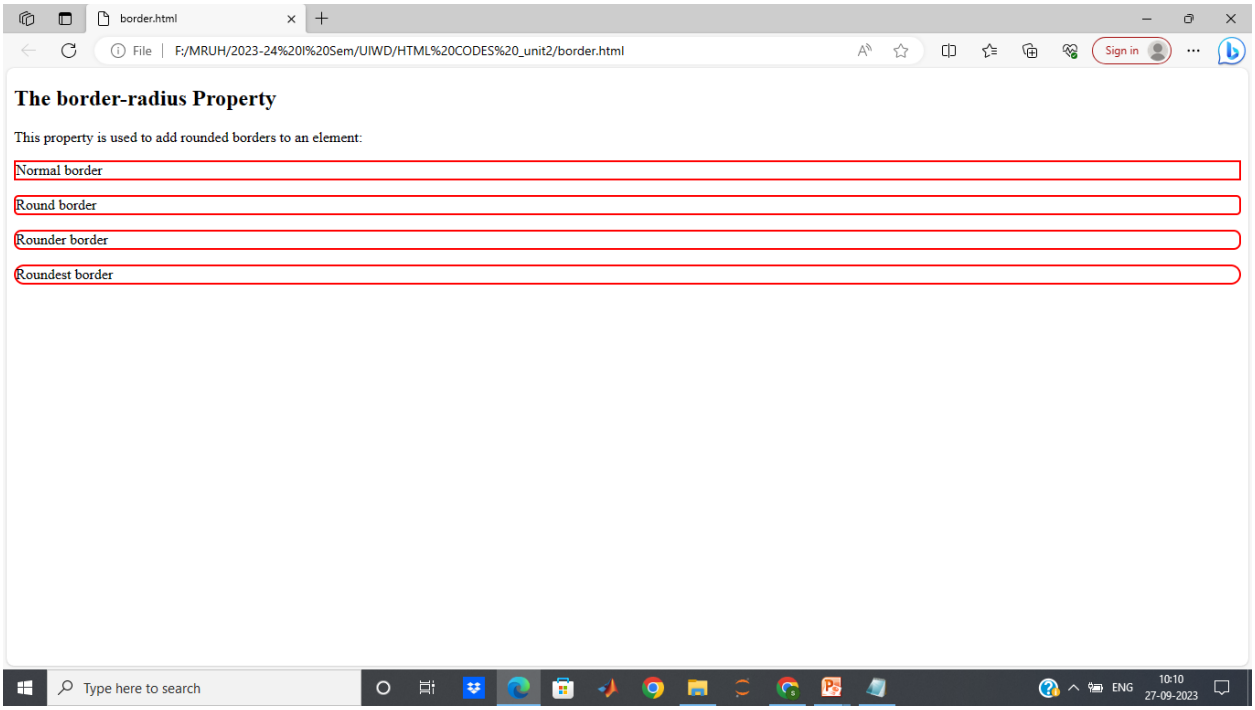
```
<!DOCTYPE html>
<html>
<head>
<style>
p.normal { border: 2px solid red; }

p.round1 { border: 2px solid red;
  border-radius: 5px; }

p.round2 { border: 2px solid red;
  border-radius: 8px; }

p.round3 { border: 2px solid red;
  border-radius: 12px; }
</style>
</head>
<body>
<h2>The border-radius Property</h2>
<p>This property is used to add rounded borders to an element:</p>
```

```
<p class="normal"> Normal border</p>
<p class="round1"> Round border</p>
<p class="round2"> Rounder border</p>
<p class="round3"> Roundest border</p>
</body>
</html>
```



CSS Fonts:

CSS font is used for design text or font for display on web page.

CSS font properties define the font family, boldness, size, and the style of a text.

CSS Font Families:

Font Style: Font style are used for set font style. Font style property has three values they are:

normal: The text is shown normally

italic: The text is shown in italics

```
<!DOCTYPE html>
<html>
<head>
<style>
p.normal { font-style: normal;}
p.italic { font-style: italic;}
</style>
</head>
<body>
```

<h1>The font-style property</h1>

This is a paragraph in normal style.

This is a paragraph in italic style.

</body>

</html>

We can define size of font in following way:

1. With Em
2. Percent
3. With Pixels

1. CSS Fonts: font-size with pixels:

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {   font-size: 40px; }
h2 {   font-size: 30px; }
p {   font-size: 14px; }
</style>
</head>

<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>

</body>
</html>
```

2. CSS Fonts: font-size with 'em': To allow users to resize the text (in the browser menu), many

developers use em instead of pixels.

1em = current font size =16px, Where, 16px is the default text size in browsers**.**

```
<!DOCTYPE html>
<html>
<head>
<style>
<body>

h1 {   font-size: 100%; }

h2 {   font-size: 300%; }

p {   font-size: 500%; }
</style> </head>
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<p>This is a paragraph.</p>
</body> </html>
```

3. **CSS Fonts: font-size with percentage:**

```
<!DOCTYPE html>
<html>
<head>
<style>
<body>
h1 {   font-size: 100%; }
h2 {   font-size: 300%; }
p {   font-size: 500%; }
</style> </head>
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<p>This is a paragraph.</p>
</body> </html>
```

Text Color and Background Color: In this example, we define both the background-color property and the color property:

Example:

```
body {
  background-color: lightgrey;
  color: blue;
}
h1 {
  background-color: black;
  color: white;
}
```

Text Alignment

The text-align property is used to set the horizontal alignment of a text. The text can be left or right aligned, centered, or justified.

Example

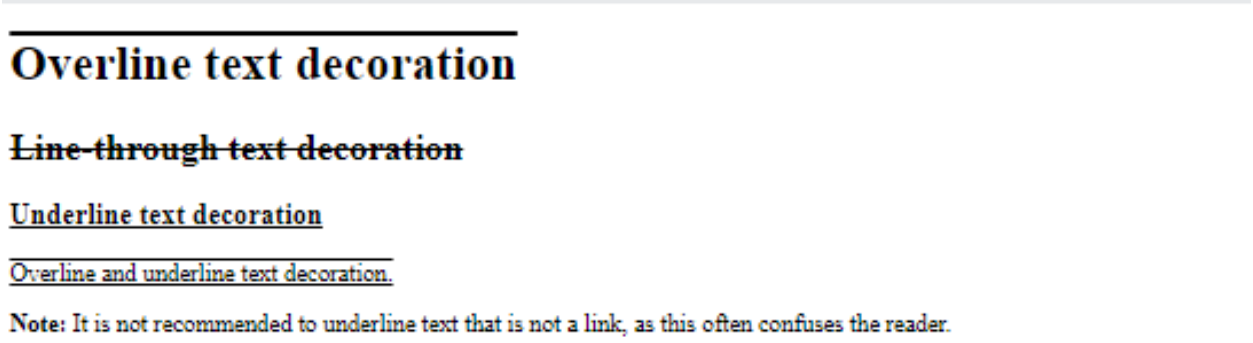
```
h1 {
  text-align: center;
}
h2 {
  text-align: left;
}
h3 {
  text-align: right;
}
```

Text Decoration: The text-decoration-line property is used to add a decoration line to text.

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
  text-decoration: overline;
}
h2 {
  text-decoration: line-through;
}
h3 {
  text-decoration: underline;
```

```
}
p.ex {
  text-decoration: overline underline;
}
</style>
</head>
<body>
<h1>Overline text decoration</h1>
<h2>Line-through text decoration</h2>
<h3>Underline text decoration</h3>
<p class="ex">Overline and underline text decoration.</p>
</body>
</html>
```



Text Transformation: The text-transform property is used to specify uppercase and lowercase letters in a text.

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
p.uppercase {
  text-transform: uppercase;
}
p.lowercase {
  text-transform: lowercase;
```

```
}
p.capitalize {
  text-transform: capitalize;
}
</style>
</head>
<body>
<h1>Using the text-transform property</h1>
<p class="uppercase">This text is transformed to uppercase.</p>
<p class="lowercase">This text is transformed to lowercase.</p>
<p class="capitalize">This text is capitalized.</p>
</body>
</html>
```

CSS Shadow Effects: With CSS shadow can be added to the text and to the elements.

Types of shadow effects:

1. Text-shadow
2. Box-shadow

- **Text-shadow:** The CSS text-shadow property applies shadow to text. In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px)

```
<!DOCTYPE html>
<html> <head> <style>
h1 {  text-shadow: 2px 2px;
/* ( horizontal shadow (2px) and the  vertical shadow (2px))*/ }
h2 {   text-shadow: 2px 2px red; }
h3 {  text-shadow: 2px 2px 5px red;  }
```

</style> </head>

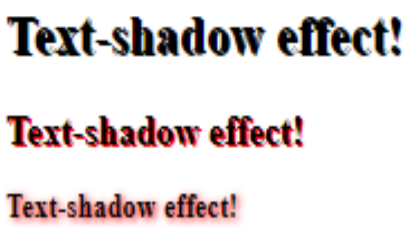
<body>

<h1>Text-shadow effect!</h1>

<h2>Text-shadow effect!</h2>

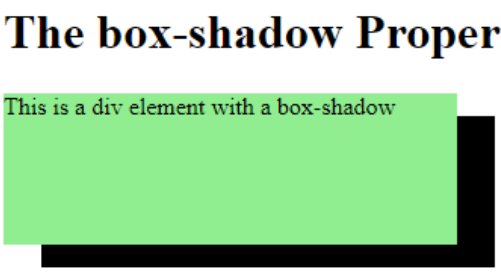
<h3>Text-shadow effect!</h3>

</body> </html>



- **Box Shadow:**

```
<!DOCTYPE html>
<html>
<head>
<style>
div { width: 300px;   height: 100px;
  background-color: lightgreen;
  box-shadow: 25px 15px; }
</style>
</head>
<body>
<h1>The box-shadow Property</h1>
<div>This is a div element with a box-shadow</div>
</body>
</html>
```



CSS Attribute Selector

- The CSS attribute selector is used when we want to style multiple HTML elements that have the same attribute or attribute values.
- It is a very convenient way to style multiple-element by grouping them on a basis of similar attributes.
- The attribute selector selects all the elements that have a particular attribute and sets the styling for all of them.
- The attribute selectors are by default case sensitive and can be written in the square brackets [].

Types of attribute selector

There are several types of attribute selector, which are given below:

1. CSS [attribute] selector
2. CSS [attribute="value"] selector
3. CSS [attribute$="value"]

CSS [attribute] selector:

The [attribute] selector selects all the elements that contain the same attribute and applies the CSS properties to all of them at once. For example, the .selector **[class]** will selects and style all the elements that have the same class name.

Syntax of [attribute] selector:

```
HTML element[attribute]/[attribute]{
/* CSS properties*/
}
```

Example

```
<style>
[class] {
background-color: red;
color: black
}
<p class="para">This is the first paragraph.</p>
```

CSS [attribute="value"] selector:

This [attribute="value"] selector allows us to select and set styling properties to all the elements whose attribute value is the same as the assigned value.

Syntax of [attribute="value"] selector:

```
HTML element [attribute="value"] {
/* CSS property*/
}
```

Example:

```
<style>
p[class="para"] {
background-color: yellow;
}
</style>
<p class="para">This is the first paragraph.</p>
```

CSS [attribute$="value"] selector

The CSS [attribute$="value"] selector selects the element whose attribute value ends with the particular value.

Syntax CSS [attribute$="value"] selector

```
[attribute$="value"] {
/* CSS properties*/
}
```

Example

```
<style>
[class$="1"]  {
border: 2px solid yellow;
}
<p class="test-1">This is the first paragraph.</p>
```

CSS gradients:

CSS gradients let you display smooth transitions between two or more specified colors.

CSS defines three types of gradients:

1. Linear Gradients (goes down/up/left/right/diagonally)
2. Radial Gradients (defined by their center)
3. Conic Gradients (rotated around a center point)

1. **Linear Gradients (goes down/up/left/right/diagonally):** To create a linear gradient at least two color stops must be defined. Color stops are the colors that render smooth transitions. we can also set a starting point and a direction (or an angle) along with the gradient effect.

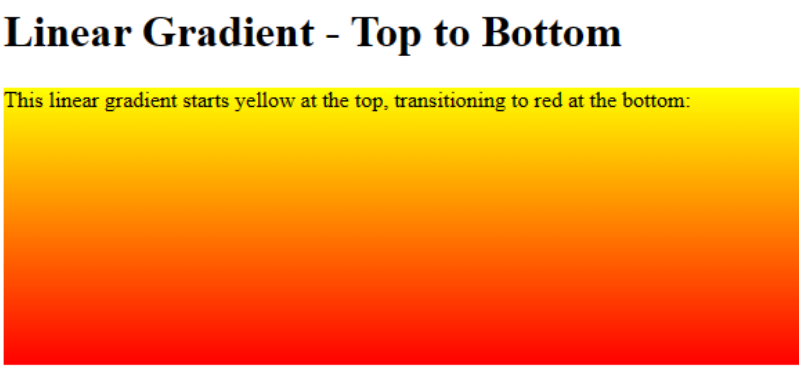
   Syntax: background-image: linear-gradient(direction, color-stop1, color-stop2, ...);

   Direction can either:

   - Top to Bottom (default)
   - Left to Right
   - Diagonal

```
<!DOCTYPE html>
```

```
<html>
<head>
<style>
#grad1 {
  height: 200px;
  background-image: linear-gradient(yellow, red);
}
</style>
</head>
<body>
<h1>Linear Gradient - Top to Bottom</h1>
<p id="grad1">This linear gradient starts yellow at the top, transitioning to red at the bottom:</p>
</body>
</html>
```



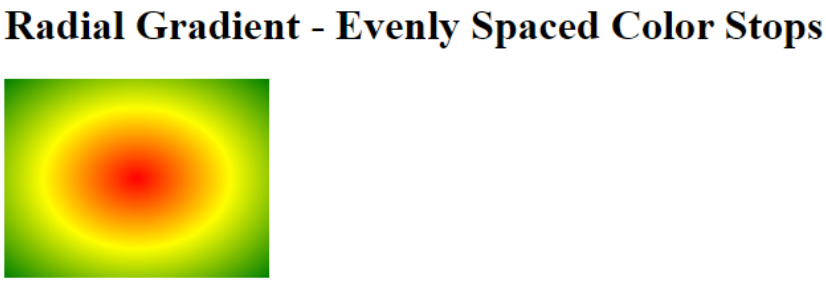
2. **CSS Radial Gradients:** A radial gradient is defined by its center. To create a radial gradient you must also define at least two color stops.

Syntax: background-image: radial-gradient(shape size at position, start-color, ..., last-color);

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {   height: 150px;
  width: 200px;
  background-image: radial-gradient(red, yellow, green); }
</style>
</head>
<body>
```

```
<h1>Radial Gradient - Evenly Spaced Color Stops</h1>
<div id="grad1"></div>
</body>
</html>
```



3. **CSS Conic Gradients:** A conic gradient is a gradient with color transitions rotated around a center point. To create a conic gradient you must define at least two colors.

   Syntax: background-image: conic-gradient([from angle] [at position,] color [degree], color [degree], ...);

Note:

- By default, *angle* is 0 degree and *position* is center.
- If no *degree* is specified, the colors will be spread equally around the center point

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
#grad1 {   height: 200px;
  width: 200px;
background-image: conic-gradient(red, yellow, green); }
</style>
</head>
<body>
<h1>Conic Gradient - Three Colors</h1>
<div id="grad1"></div>
</body>
</html>
```

Example:

```
<!DOCTYPE html>
<html>
```

```
<head>
<style>
#grad1 {   height: 200px;   width: 200px;
background-image: conic-gradient(red, yellow, green, blue, black); }
</style>
</head>
<body>
<h1>Conic Gradient - Five Colors</h1>
<div id="grad1"></div>
</body>
    </html>
```

Conic Gradient - Five Colors