

# **APPLICATION DEVELOPMENT PRACTICES**

**Amypo Technologies Pvt Ltd**

# CSS:



- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External style sheets are stored in CSS files

## Why Use CSS:

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

## Syntax:

```
selector {  
    property: value;  
    /* more properties and values */  
}
```

**Selector:** It selects the HTML element you want to style. For example, if you want to style all paragraphs, the selector would be p. If you want to style an element with a specific class, the selector would be .classname. If you want to style an element with a specific ID, the selector would be #id.

**Property:** It defines the aspect of the selected element that you want to style. For example, color, font-size, margin, etc.

**Value:** It specifies the value for the chosen property. For example, red, 14px, 10%, etc.

## Type of css:

1. Inline
2. Internal or Embedded
3. External

### Inline CSS:

Applied directly to HTML elements using the style attribute.

Example:

```
<p style="color: blue; font-size: 16px;">This is a paragraph with  
inline CSS.</p>
```

### Internal or Embedded CSS:

Placed within the HTML document using the <style> tag in the <head> section.

## Example:

```
<head>
  <style>
    p {
      color: green;
      font-size: 18px;
    }
  </style>
</head>
<body>
  <p>This is a paragraph with embedded CSS.</p>
</body>
```

## External CSS:

Defined in a separate CSS file (.css) and linked to HTML documents using the <link> tag in the <head> section.

## Example:

Indx.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>External CSS Example</title>
  <!-- Linking to the external CSS file -->
  <link rel="stylesheet" type="text/css" href="styles.css">
</head>
<body>
```

```
<h1>Welcome to My Website</h1>
```

```
<p>This is a paragraph on the page.</p>
```

```
<p class="highlight">This paragraph has a special highlight.</p>
```

```
</body>
```

```
</html>
```

Style.css

```
p {  
  color: navy;  
  font-size: 18px;  
}  
body{  
background-color : red;  
}
```

## Selector:

1. Simple selectors (select elements based on name, id, class)
2. Combinator selectors (select elements based on a specific relationship between them)
3. Pseudo-class selectors (select elements based on a certain state)
4. Pseudo-elements selectors (select and style a part of an element)
5. Attribute selectors (select elements based on an attribute or attribute value)



## The CSS element Selector:

The element selector selects HTML elements based on the element name.

### Example:

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

## 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:

```
#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:

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

## The CSS Universal Selector:

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

### Example:

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

## The CSS Grouping Selector:

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

Example:

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

Selector	Example	Example description
<a href="#"><u>#id</u></a>	#firstname	Selects the element with id="firstname"
<a href="#"><u>.class</u></a>	.intro	Selects all elements with class="intro"
<a href="#"><u>element.class</u></a>	p.intro	Selects only <p> elements with class="intro"
<a href="#"><u>*</u></a>	*	Selects all elements
<a href="#"><u>element</u></a>	p	Selects all <p> elements
<a href="#"><u>element,element,..</u></a>	div, p	Selects all <div> elements and all <p> elements

## Pseudo-classes:

A pseudo-class is used to define a special state of an element.

### Syntax

```
selector:pseudo-class {  
    property: value;  
}
```

### Anchor Pseudo-classes:

- link,
- visited,
- hover,
- Active

# Example:

```
<!DOCTYPE html>
<html>

<head>
  <style>
    /* unvisited link */
    a:link {
      color: red;
    }

    /* visited link */
    a:visited {
      color: green;
    }

    /* mouse over link */
    a:hover {
      color: hotpink;
    }
  </style>
</head>

<body>

  <h2>Styling a link depending on state</h2>

  <p><b><a href="https://skcet.amypo.com/login"
  target="_blank">This is a link</a></b></p>

</body>

</html>
```

```
/* selected link */
a:active {
  color: blue;
}
</style>
</head>

<body>

  <h2>Styling a link depending on state</h2>

  <p><b><a href="https://skcet.amypo.com/login"
  target="_blank">This is a link</a></b></p>

</body>

</html>
```

## Pseudo-elements:

A CSS pseudo-element is used to style specified parts of an element.

For example, it can be used to:

Style the first letter, or line, of an element

Insert content before, or after, the content of an element

## Syntax:

```
selector::pseudo-element {  
  property: value;  
}
```

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
h1::before {
  content: url(smiley.gif);
}
p::after{
content:url(smiley.gif);}
p:hover{
color:red;
font-size:25px;}
</style>
</head>
```

```
<body>
```

```
<h1>This is a heading</h1>
```

```
<p>The ::before pseudo-element inserts
content before the content of an
element.</p>
```

```
<h1>This is a heading</h1>
```

```
</body>
```

```
</html>
```

## Text Color and Background Color:

```
h1 {  
  background-color: black;  
  color: white;  
}
```

**This is a Heading**

```
div {  
  background-color: blue;  
  color: white;  
}
```

This is a div.



## Text Alignment:

- text-align
- text-align-last
- direction
- unicode-bidi
- vertical-align

### text-align:

center, right, left

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
  text-align: center;
}
h2 {
  text-align: left;
}
h3 {
  text-align: right;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Heading 1 (center)</h1>
```

```
<h2>Heading 2 (left)</h2>
```

```
<h3>Heading 3 (right)</h3>
```

```
<p>The three headings above are aligned center, left and right.</p>
```

```
</body>
```

```
</html>
```

Output:

**Heading 1 (center)**

**Heading 2 (left)**

**Heading 3 (right)**

The three headings above are aligned center, left and right.

## Text Decoration Line:

- text-decoration-line
- text-decoration-color
- text-decoration-style
- text-decoration-thickness
- text-decoration

### text-decoration-line:

- overline,
- line-through,
- underline

Property	Description
<a href="#"><u>text-decoration</u></a>	Sets all the text-decoration properties in one declaration
<a href="#"><u>text-decoration-color</u></a>	Specifies the color of the text-decoration
<a href="#"><u>text-decoration-line</u></a>	Specifies the kind of text decoration to be used (underline, overline, etc.)
<a href="#"><u>text-decoration-style</u></a>	Specifies the style of the text decoration (solid, dotted, etc.)
<a href="#"><u>text-decoration-thickness</u></a>	Specifies the thickness of the text decoration line

## 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>

<p><strong>Note:</strong> It is not recommended to
underline text that is not a link, as this often confuses
the reader.</p>

</body>
</html>
```

Text-decoration-style :

- Solid
- Dashed
- Dotted
- Double
- Wavy

text-decoration:

it's a short hand property

Example:

```
p {  
    text-decoration: underline red double  
    5px;  
}
```

## Text-Transform :

- uppercase
- lowercase
- capitalize

## Text-indent:

- letter-spacing
- line-height
- word-spacing
- white-space

Property	Description
<a href="#">letter-spacing</a>	Specifies the space between characters in a text
<a href="#">line-height</a>	Specifies the line height
<a href="#">text-indent</a>	Specifies the indentation of the first line in a text-block
<a href="#">white-space</a>	Specifies how to handle white-space inside an element
<a href="#">word-spacing</a>	Specifies the space between words in a text

## Text Shadow:

The text-shadow property adds shadow to text.

In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px):

## Text shadow effect!

Example:

```
h1 {  
  text-shadow: 2px 2px red;  
}
```

## Font styling:

- Font-family
- Font-style
- Font-variant
- Font-weight
- Font-size/Line-height

## Font family:

we use the font-family property to specify the font of a text.

Ex:

```
.p1 {  
font-family: "Times New Roman", Times, serif;  
}
```



## Font-style:

The font-style property is mostly used to specify italic text.

This property has three values:

1. normal - The text is shown normally
2. italic - The text is shown in italics
3. oblique - The text is "leaning" (oblique is very similar to italic, but less supported)

Ex:

```
p.normal {  
  font-style: normal;  
}
```

```
p.italic {  
  font-style: italic;  
}
```

```
p.oblique {  
  font-style: oblique;  
}
```

# The font-style property

This is a paragraph in normal style.

*This is a paragraph in italic style.*

*This is a paragraph in oblique style.*



## Font Weight:

to set a font weight

Example:

```
p.normal {  
  font-weight: normal;  
}
```

```
p.thick {  
  font-weight: bold;  
}
```

Value	Common weight name
100	Thin (Hairline)
200	Extra Light (Ultra Light)
300	Light
400	Normal
500	Medium
600	Semi Bold (Demi Bold)
700	Bold
800	Extra Bold (Ultra Bold)
900	Black (Heavy)

## Font-size:

the font-size property sets the size of the text.  
Being able to manage the text size is important in web design.

Size :

- px (1pixel)
- em(16pixels)
- vw(view port width)
- %(percentage)

Ex:

```
h1 {  
  font-size: 2.5em;  
}  
  
h2 {  
  font-size: 17px;  
}
```

## 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

Ex:

```
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;}
```

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border. The effect depends on the border-color value.

A ridge border. The effect depends on the border-color value.

An inset border. The effect depends on the border-color value.

An outset border. The effect depends on the border-color value.

No border.

A hidden border.

A mixed border.

```
p.mix {border-style: dotted dashed solid double;}
```

## 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

Ex:

```
p.one {  
  border-style: solid;  
  border-color: rgb(255, 0, 0); /* red */  
}  
  
p.one {  
  border-style: solid;  
  border-color: #ff0000; /* red */  
}  
  
p.one {  
  border-style: solid;  
  border-color: red green blue yellow; /* red top,  
green right, blue bottom and yellow left */  
}
```

## Rounded Borders:

```
p {
  border: 2px solid red;
  border-radius: 5px;
}
p {
  border: 2px solid red;
  border-radius: 5px 10px;
//top ,bottom -5px left,right-10px
}
```

Normal border

Round border

Rounder border

Roudest border

## Border shadow:

using box shadow property

Ex:

```
#example1 {
  box-shadow: 5px 10px;
  border: 2px solid red;
  border-radius: 5px 10px;
}
```

A div element with a shadow. The first value is the horizontal offset and the second value is the vertical offset. The shadow color will be inherited from the text color.

`box-shadow: 5px 10px #000000;`