



# CSS

# Cascading Style Sheets

---

**Class # 1**

# Introduction

- **Styling:** CSS is used to define the visual appearance of web pages.
- **Separation:** CSS separates the presentation (styling) from the structure (HTML) of a web page.
- **Consistency:** CSS ensures consistent styling across multiple web pages or an entire website.
- **Responsiveness:** CSS enables responsive design, adapting the layout for different screen sizes and devices.
- **Reusability:** CSS allows for the creation of reusable styles, reducing code duplication and improving efficiency.

# 3 Ways to add CSS (Inline CSS)

---

□ **Inline CSS:** Inline CSS is added directly to the HTML elements using the "style" attribute. It applies styles to a specific element, but it can be repetitive and less maintainable for larger projects.

```
<!DOCTYPE html>
<html>
<head>
  <title>Internal CSS Example</title>
</head>
<body>
  <p style="color: blue; font-size: 16px;">This text is styled using internal CSS.</p>
</body>
</html>
```

# 3 Ways to add CSS (Internal CSS)

---

□ **Internal CSS:** Internal CSS is included within the HTML file using the `<style>` tag in the head section. It applies styles to the entire document or specific elements within it, allowing for better organization and maintainability compared to inline CSS.

```
<!DOCTYPE html>
<html>
<head>
  <title>Internal CSS Example</title>
  <style>
    p {
      color: blue;
      font-size: 16px;
    }
  </style>
</head>
<body>
  <p>This text is styled using internal CSS.</p>
</body>
</html>
```

# 3 Ways to add CSS (External CSS)

---

□ **External CSS:** External CSS is defined in a separate CSS file(style.css) and linked to the HTML file using the <link> tag. It allows for the centralization of styles, making them reusable across multiple HTML files and improving maintenance and scalability of the project.

```
.blue-text {  
  color: blue;  
  font-size: 16px;  
}
```

Style.css

# 3 Ways to add CSS (External CSS)

External CSS file ("styles.css") to the HTML file using the <link> tag.

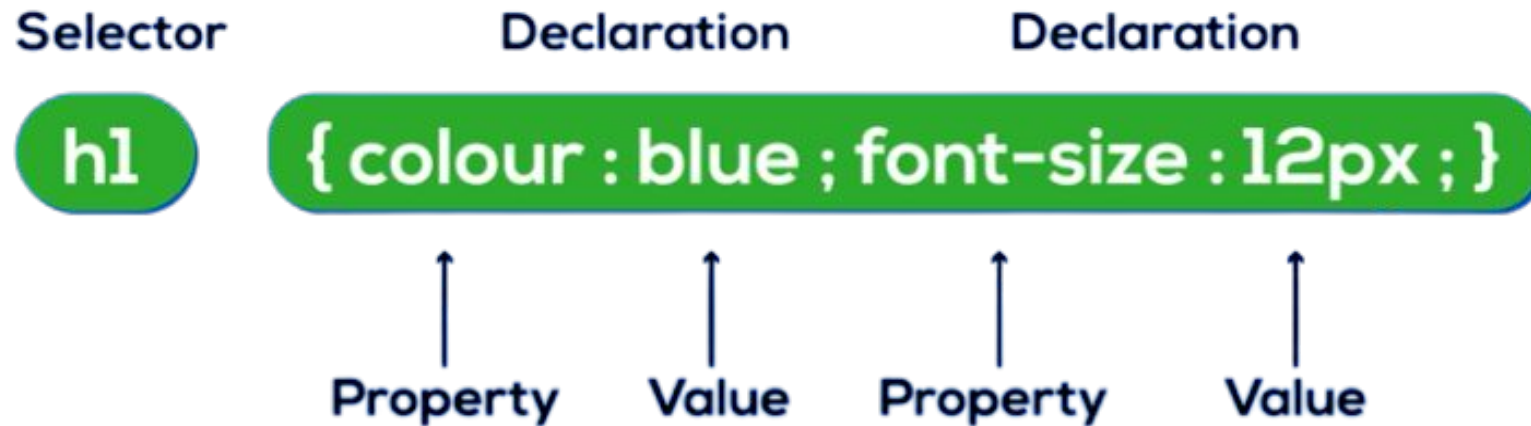
```
<!DOCTYPE html>
<html>
<head>
  <title>External CSS Example</title>
  <link rel="stylesheet" type="text/css" href="styles.css">
</head>
<body>
  <p class="blue-text">This text is styled using external CSS.</p>
</body>
</html>
```

Index.html

# Selectors In CSS

In CSS, selectors are used to target and select specific HTML elements for styling.

**Syntax:**



# Types of Selectors(Universal Selector)

The universal selector (\*) matches any element in the document and is used to apply styles globally.

```
* {  
  /* Styles applied to the whole html file*/  
}
```



# Types of Selectors(Element Selector)

An element selector in CSS is used to target and apply styles to a specific type of HTML element. It selects and styles all occurrences of that element type within the document. For example, using the selector "p" will select and style all paragraph elements in the HTML document.

```
p {  
  /* Styles applied to all <p> elements */  
}
```

# Types of Selectors(Class Selector)

Targets elements with a specific class attribute, allowing you to apply styles to multiple elements with the same class. Example: ".my-class" selects elements with the class "my-class".

```
.my-class {  
    /* Styles applied to elements with class "my-class" */  
}
```

# Types of Selectors(ID Selector)

Targets a unique element based on its ID attribute, allowing you to apply styles to a single element. Example: "#my-id" selects the element with the ID "my-id".

```
#my-id {  
  /* Styles applied to the element with id "my-id" */  
}
```

# Types of Selectors(Attribute Selector)

Targets elements based on their attribute values, such as type, href, or data attributes. Example: "input[type='text']" selects input elements with the attribute type set to "text".

```
input[type="text"] {  
  /* Styles applied to <input> elements with attribute type="text" */  
}
```

# Types of Selectors(Descendant Selector)

Selects elements that are descendants of another element, regardless of the level of nesting. Example: ".container p" selects all paragraph elements that are descendants of elements with the class "container".

```
.container p {  
  /* Styles applied to <p> elements that are descendants of elements with class "container" */  
}
```

# Types of Selectors(Child Selector)

Selects elements that are direct children of another element. Example: ".container > p" selects all paragraph elements that are immediate children of elements with the class "container".

```
.container > p {  
  /* Styles applied to <p> elements that are direct children of elements with class "container" */  
}
```

# Types of Selectors(Pseudo-Class Selector)

Targets elements based on a specific state or condition, such as :hover, :active, or :first-child. Example: "a:hover" selects anchor elements when hovered over.

```
a:hover {  
  /* Styles applied to <a> elements when hovered over by the cursor */  
}
```

# Types of Selectors(Pseudo-Element Selector)

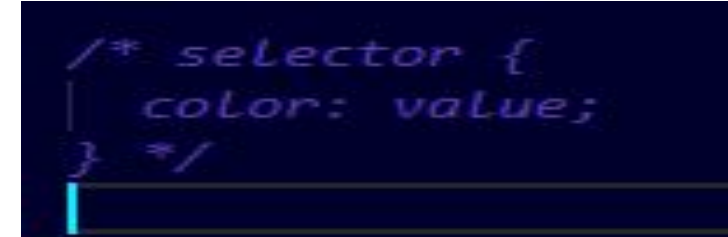
Targets specific parts of an element's content or structure, such as :before or :after, to apply styles. Example: "p::first-line" selects the first line of paragraph elements.

```
p::first-line {  
    /* Styles applied to the first line of <p> elements */  
}
```



# CSS Properties(Color)

The color property in CSS is used to specify the text color of an element. It determines the foreground color for the content within that element.

A screenshot of a code editor with a dark blue background. It shows a CSS rule snippet: 

```
/* selector {  
  color: value;  
} */
```

 A vertical blue cursor is positioned at the start of the line containing 'color: value;'.

## Syntax

### Usage:

- ❑ **Hexadecimal color:** color: #RRGGBB; (e.g., color: #FF0000; for red)
- ❑ **RGB color:** color: rgb(R, G, B); (e.g., color: rgb(255, 0, 0); for red)
- ❑ **Color names:** color: colorname; (e.g., color: red;)
- ❑ **RGBA color (with transparency):** color: rgba(R, G, B, A); (e.g., color: rgba(255, 0, 0, 0.5); for semi-transparent red)

# CSS Properties(Color)

## Examples:

```
h1 {  
  color: #FF0000; /* Sets the text color of <h1> elements to red */  
}
```

```
p {  
  color: rgb(0, 128, 0); /* Sets the text color of <p> elements to green */  
}
```

```
span {  
  color: rgba(255, 0, 0, 0.5); /* Sets the text color of <span> elements to  
  semi-transparent red */  
}
```

# CSS Properties(Background Color)

The background-color property in CSS is used to specify the background color of an element. It sets the color behind the content of the element.

```
/* selector {  
  background-color: value;  
} */
```

## Syntax

### Usage:

- ❑ **Hexadecimal color:** color: #RRGGBB; (e.g., color: #FF0000; for red)
- ❑ **RGB color:** color: rgb(R, G, B); (e.g., color: rgb(255, 0, 0); for red)
- ❑ **Color names:** color: colorname; (e.g., color: red;)
- ❑ **RGBA color (with transparency):** color: rgba(R, G, B, A); (e.g., color: rgba(255, 0, 0, 0.5); for semi-transparent red)

# CSS Properties(Background Color)

## Examples:

```
body {  
  background-color: #F0F0F0; /* Sets the background color of the entire page to a light gray */  
}
```

```
div {  
  background-color: rgb(255, 0, 0); /* Sets the background color of <div> elements to red */  
}
```

```
button {  
  background-color: rgba(0, 0, 255, 0.3); /* Sets the background color of <button> elements to  
semi-transparent blue */  
}
```

# CSS Properties(Background Image)

The background-image property in CSS is used to set an image as the background of an element. It allows you to add visual interest and enhance the design of the element.

```
selector {  
  background-image: url("image-url");  
}
```

## Syntax

### Usage:

- ❑ **External image:** background-image: url("path/to/image.jpg");
- ❑ **Inline data URL:** background-image: url("data:image/png;base64,base64-encoded-image-data");

# CSS Properties(Background Image)

## Examples:

```
body {  
    background-image: url("path/to/image.jpg"); /* Sets the background image of the entire page  
*/  
}
```

```
div {  
    background-image: url("data:image/png;base64,iVBORw0KG..."); /* Sets the background image  
of <div> elements using a data URL */  
}
```

```
section {  
    background-image: none; /* Removes the background image from <section> elements */  
}
```

# CSS Properties(Background Repeat)

The background-repeat property in CSS is used to control how a background image is repeated within an element. It determines if and how the image is repeated horizontally and vertically.

```
selector {  
  background-repeat: value;  
}
```

## Syntax

### Usage:

- ❑ **background-repeat: repeat;** - The background image is repeated both horizontally and vertically (default).
- ❑ **background-repeat: repeat-x;** - The background image is repeated only horizontally.
- ❑ **background-repeat: repeat-y;** - The background image is repeated only vertically.
- ❑ **background-repeat: no-repeat;** - The background image is not repeated and displayed only once.

# CSS Properties(Background Repeat)

## Examples:

```
body {  
    background-repeat: repeat; /* The background image is repeated both horizontally and  
vertically */  
}
```

```
div {  
    background-repeat: no-repeat; /* The background image is displayed only once within <div>  
elements */  
}
```

```
section {  
    background-repeat: repeat-x; /* The background image is repeated only horizontally within  
<section> elements */  
}
```



# CSS Properties(Background Position)

The background-position property in CSS is used to specify the starting position of a background image within an element. It determines where the background image is positioned horizontally and vertically.

```
selector {  
  background-position: value;  
}
```

## Syntax

### Usage:

- **background-position: x-value y-value;** - Sets the horizontal and vertical position of the background image using keywords, length values, percentages, or a combination of them.

# CSS Properties(Background Position)

## Examples:

```
body {  
    background-position: center top; /* The background image is positioned horizontally at the  
    center and vertically at the top */  
}
```

```
div {  
    background-position: 50% 25%; /* The background image is positioned horizontally at 50% and  
    vertically at 25% within <div> elements */  
}
```

```
section {  
    background-position: right bottom; /* The background image is positioned horizontally at the  
    right and vertically at the bottom within <section> elements */  
}
```

# CSS Properties(Background Size)

The background-size property in CSS is used to control the size of a background image within an element. It determines how the background image is scaled or displayed in relation to the element's dimensions.

```
selector {  
  background-size: value;  
}
```

## Syntax

### Usage:

- ❑ **background-size: auto;** - The background image is displayed at its original size (default).
- ❑ **background-size: cover;** - The background image is scaled proportionally to cover the entire background area, potentially cropping parts of the image.
- ❑ **background-size: contain;** - The background image is scaled proportionally to fit within the background area entirely, potentially leaving empty space.
- ❑ **background-size: lengthValue;** - The background image is set to a specific length value, such as pixels (px) or percentages (%).
- ❑ **background-size: lengthValueX lengthValueY;** - Sets the width and height of the background image using different length values.

# CSS Properties(Background Size)

## Examples:

```
body {  
    background-size: cover; /* The background image is scaled to cover the entire background area  
of the page */  
}
```

```
div {  
    background-size: contain; /* The background image is scaled to fit entirely within <div>  
elements */  
}
```

```
section {  
    background-size: 200px; /* The background image is set to a width of 200 pixels within  
<section> elements */  
}
```

# CSS Properties(Background Attachment)

The background-attachment property in CSS is used to control whether a background image scrolls with the content or remains fixed in place as the content is scrolled. It determines how the background image is attached to the viewport.

```
selector {  
  background-attachment: value;  
}
```

## Syntax

### Usage:

- ❑ **background-attachment: scroll;** - The background image scrolls along with the content as the element is scrolled (default).
- ❑ **background-attachment: fixed;** - The background image remains fixed in place, regardless of scrolling. The image appears to be attached to the viewport.
- ❑ **background-attachment: local;** - The background image scrolls with the content of the element, separate from the main document scrolling.

# CSS Properties(Background Attachment)

## Examples:

```
body {  
    background-attachment: fixed; /* The background image remains fixed as the page is scrolled  
    */  
}
```

```
div {  
    background-attachment: scroll; /* The background image scrolls along with the content of  
<div> elements */  
}
```

```
section {  
    background-attachment: local; /* The background image scrolls with the content within  
<section> elements */  
}
```

# CSS Properties(Font Family)

The font-family property in CSS is used to specify the font or list of fonts to be used for text content within an element.

```
selector {  
  font-family: value;  
}
```

## Syntax

### Usage:

- ❑ **font-family: fontname;** - Specifies a specific font name to be used. For example, font-family: Arial;.
- ❑ **font-family: generic-name;** - Specifies a generic font family, such as serif, sans-serif, monospace, cursive, or fantasy. The browser will use the default font of that category on the user's system.

# CSS Properties(Font Family)

## Examples:

```
body {  
    font-family: Arial, sans-serif; /* Sets the font to Arial, and if not available, uses a sans-serif  
    fallback font */  
}
```

```
h1 {  
    font-family: "Helvetica Neue", Arial, sans-serif; /* Sets the font to "Helvetica Neue", and if not  
    available, uses Arial or a sans-serif fallback font */  
}
```

```
p {  
    font-family: serif; /* Sets the font to a default serif font on the user's system */  
}
```



# CSS Properties(Font Size)

The font-size property in CSS is used to specify the size of the text content within an element. It determines the height of the font glyphs, affecting the overall appearance and readability of the text.

```
selector {  
  font-size: value;  
}
```

## Syntax

### Usage:

- ❑ **font-size: lengthValue;** - Sets the font size to a specific length value, such as pixels (px), points (pt), em, rem, or percentages (%).
- ❑ **font-size: xx-small | x-small | small | medium | large | x-large | xx-large;** - Specifies a predefined size relative to the user's default font size.

# CSS Properties(Font Size)

## Examples:

```
body {  
  font-size: 16px; /* Sets the font size to 16 pixels */  
}
```

```
h1 {  
  font-size: 2em; /* Sets the font size to twice the size of the parent element's font size */  
}
```

```
p {  
  font-size: large; /* Sets the font size to a predefined large size */  
}
```

# CSS Properties(Font Style)

The font-style property in CSS is used to specify the style of the text within an element. It determines whether the text is displayed in a normal, italic, or oblique style.

```
selector {  
  font-style: value;  
}
```

## Syntax

### Usage:

- ❑ **font-style: normal;** - Displays the text in a normal style (default).
- ❑ **font-style: italic;** - Displays the text in an italic style. If the italic font variant is not available, the browser may simulate the style by applying oblique.
- ❑ **font-style: oblique;** - Displays the text in an oblique style. It slants the text without using the italic font variant.

# CSS Properties(Font Style)

## Examples:

```
body {  
  font-style: normal; /* Displays the text in a normal style */  
}
```

```
h1 {  
  font-style: italic; /* Displays the text in an italic style */  
}
```

```
p {  
  font-style: oblique; /* Displays the text in an oblique style */  
}
```

# CSS Properties(Font Weight)

The font-weight property in CSS is used to specify the weight or thickness of the text within an element. It determines whether the text appears normal, bold, or in a different weight variant if available.

```
selector {  
  font-weight: value;  
}
```

## Syntax

### Usage:

- ❑ **font-weight: normal;** - Displays the text in a normal weight (default).
- ❑ **font-weight: bold;** - Displays the text in a bold weight. If a bold font variant is not available, the browser may simulate the boldness by using a different variant or algorithm.
- ❑ **font-weight: number;** - Specifies a numeric value from 100 to 900, indicating different weight levels. For example, font-weight: 400; is equivalent to font-weight: normal, and font-weight: 700; is equivalent to font-weight: bold.

# CSS Properties(Font Weight)

## Examples:

```
body {  
  font-weight: normal; /* Displays the text in a normal weight */  
}
```

```
h1 {  
  font-weight: bold; /* Displays the text in a bold weight */  
}
```

```
p {  
  font-weight: 600; /* Displays the text in a weight level of 600 */  
}
```