

# CSS

## Cascading Style Sheets



CSS transforms plain HTML into beautiful, responsive experiences that engage users and bring your designs to life. It is used to describe the style of a document.

### Style & Layout

Control colors, fonts, spacing, and positioning to create polished interfaces

### Responsive Design

Build sites that adapt seamlessly across devices and screen sizes

### Animations & Effects

Add dynamic interactions and smooth transitions that delight users

# Basic Format of CSS

CSS (Cascading Style Sheets) uses a simple, rule-based syntax to apply styles to HTML elements. Understanding this fundamental structure is key to controlling the appearance of your web pages.

## Selector

Targets the HTML element you want to style (e.g., `p` for paragraphs, `h1` for headings, `.class-name`, or `#id`).

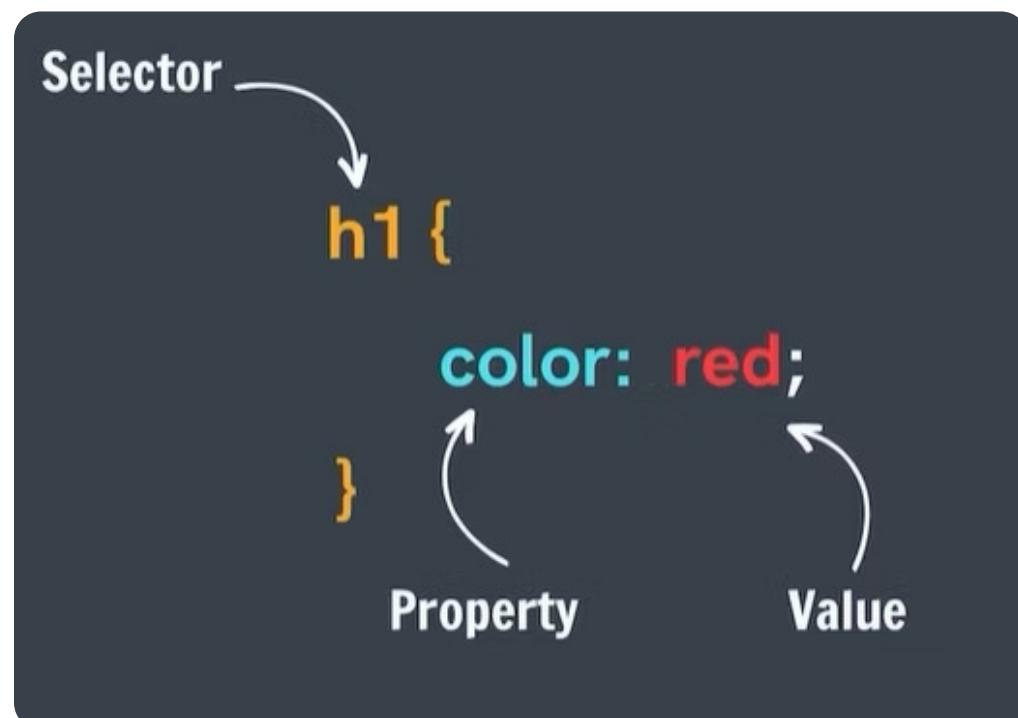
## Property

The specific aspect of the element you want to change (e.g., `color`).

## Value

Defines the setting for the chosen property (e.g., `red` for color, `16px` for font-size, `10px` for margin).

## CSS Rule Structure



# How to Include Styles

CSS styles can be applied to HTML documents in three primary ways, each suited for different scenarios and offering varying levels of flexibility and maintainability.

1

## Inline Styling

Styles applied directly to individual HTML elements using the `style` attribute.

```
<p style="color: blue; font-size: 16px;">Text</p>
```

**Use case:** Quick, specific styling for individual elements, often used for minor adjustments or testing.

2

## Internal Stylesheet

CSS rules placed within a `<style>` tag, typically located in the `<head>` section of an HTML document.

```
<head>
  <style>
    h1 {
      color: green;
    }
  </style>
</head>
<body>
  <h1>Page Title</h1>
</body>
```

**Use case:** Styling for a single HTML page when the styles are unique to that page.

3

## External Stylesheet

CSS rules stored in a separate `.css` file and linked to the HTML document using the `<link>` tag in the `<head>` section.

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

**Use case:** Best practice for styling multiple pages, promoting maintainability, reusability, and faster loading times due to caching.

# Styling Properties

CSS properties are fundamental in controlling the visual appearance and layout of HTML elements. They allow developers to specify everything from colors and fonts to spacing and positioning, creating the aesthetic design of a webpage.

## Color Property

The `color` property is used to set the foreground color of an element's text content. It defines the color of the text itself.

```
p {  
  color: red;  
}
```

**Note:** This property is specifically applied to textual information within an element.

Reference: [MDN Web Docs](#)

## Background-color Property

The `background-color` property sets the background color of an element. This color fills the content, padding, and border box of the element.

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

## Color Systems

### Named Colors

Predefined color names that are recognized by browsers, making them easy to use for common colors.

- **Description:** Simple, human-readable names like `red`, `blue`, `pink`.
- **Note:** There are over 140 named colors available.

```
color: red;
```

### RGB (Red, Green, Blue)

A color model that represents millions of shades by combining varying intensities of red, green, and blue light.

- **Description:** Defined by three values for red, green, and blue components.
- **Format:** `rgb(red, green, blue)` where each value is an integer from 0-255.
- **Use case:** Ideal for precise color control, especially for colors that don't have predefined names.

```
color: rgb(255, 0, 0); /* Pure  
Red */
```

### Hex Codes (Hexadecimal)

A widely used way to specify colors in web design, offering a compact and powerful notation.

- **Description:** Uses a base-16 number system, represented by a `#` followed by 6 hexadecimal characters.
- **Format:** `#RRGGBB`, where RR, GG, and BB are two-digit hexadecimal numbers for red, green, and blue respectively.
- **Range:** From `#000000` (black) to `#FFFFFF` (white).

```
color: #FF0000; /* Pure Red */  
background-color: #FFFFFF; /*  
White */
```

# Text Properties

CSS text properties are essential for controlling the visual appearance and formatting of textual content on web pages. They allow developers to manipulate aspects such as alignment, weight, decoration, spacing, and height of text, ensuring readability and aesthetic appeal.

## Text Properties

**text-align**

**font-weight**

**text-decoration**

**line-height**

**letter-spacing**

**font-size**

**text-align**

Controls the horizontal alignment of text within an element.

- **Common values:** left, right, center, justify

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

**font-weight**

Controls the thickness or boldness of text.

- **Common values:** normal, bold, 100-900 (numeric values), bolder, lighter

```
h1 {  
    font-weight: bold;  
}
```

**font-weight**

**font-weight: normal**

**font-weight: bold**

**font-weight: 100**

**font-weight: bolder**

**font-weight: lighter**

**text-decoration**

Adds decorative lines to text.

- **Values:** underline, overline, line-through, none
- **Sub-properties:**
  - **text-decoration-color:** Sets the color of the decoration line (e.g., red)
  - **text-decoration-style:** Sets the style of the line (e.g., wavy, solid, dashed, dotted)

```
a {  
    text-decoration: underline;  
    text-decoration-color: red;  
    text-decoration-style: wavy;  
}
```

**line-height**

Controls the vertical spacing between lines of text (the height of each line).

- **Values:** Can be unitless numbers, pixels, percentages, or em units

```
p {  
    line-height: 1.5; /* 1.5 times  
    the font size */  
}
```

**letter-spacing**

Controls the horizontal space between characters in text.

- **Values:** Can be positive (increase spacing) or negative (decrease spacing)

```
h2 {  
    letter-spacing: 2px;  
}
```

# Font Properties

CSS font properties are crucial for controlling the typography and overall appearance of text on web pages. They allow developers to customize aspects like size, typeface, style, and weight, ensuring readability and visual consistency across different devices and browsers.

## font-size

Controls the size of text within an element.

### Font-size units in CSS

#### Absolute

- px
- pt
- pc
- cm
- mm
- in

#### Relative

- %
- em
- rem
- ch
- vh
- vw + many more

Font sizes can be specified using various units, broadly categorized into absolute and relative units:

#### Absolute Units:

- px (pixels) - most commonly used, fixed size
- pt (points)
- pc (picas)

#### Relative Units:

- % (percentage) - relative to the parent element's font size
- em (relative to the parent element's font size)
- rem (relative to the root element's font size)
- vh (viewport height)
- vw (viewport width)

### Pixels (px)

**most commonly used absolute unit**

**96px = 1 inch**

**not suitable for responsive websites**

**font-size: 50px ;**

#### Key points about Pixels:

- Most commonly used absolute unit.
- 96px is generally considered equal to 1 inch on screens.
- Not suitable for responsive websites where text size needs to adapt to different screen sizes.

```
p {  
    font-size: 50px;  
}
```

## font-family

Specifies the typeface or font to be used for text within an element.

- Common values:** serif, sans-serif, monospace, cursive, fantasy. These are generic font families that represent a collection of fonts with similar characteristics.
- You can specify multiple fonts as fallbacks, separated by commas. The browser will use the first font in the list that is available on the user's system.

```
p {  
    font-family: Arial, Helvetica, sans-serif;  
}
```

- Note:** Font names with spaces (e.g., "Times New Roman") should be enclosed in quotes.
- Consider using **web-safe fonts** for maximum compatibility, or carefully implement **custom fonts** using @font-face rules to ensure they load correctly for all users.