

# CSS Introduction

## What is 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 stylesheets 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.

---

## CSS Solved a Big Problem

HTML was NEVER intended to contain tags for formatting a web page. HTML was created to **describe the content** of a web page, like:

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

```
<p>This is a paragraph.</p>
```

When tags like `<font>`, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS. CSS was created to specify the document's style, not its content.

In HTML 4.0, and later, all formatting should be removed from the HTML page, and stored in separate CSS files.

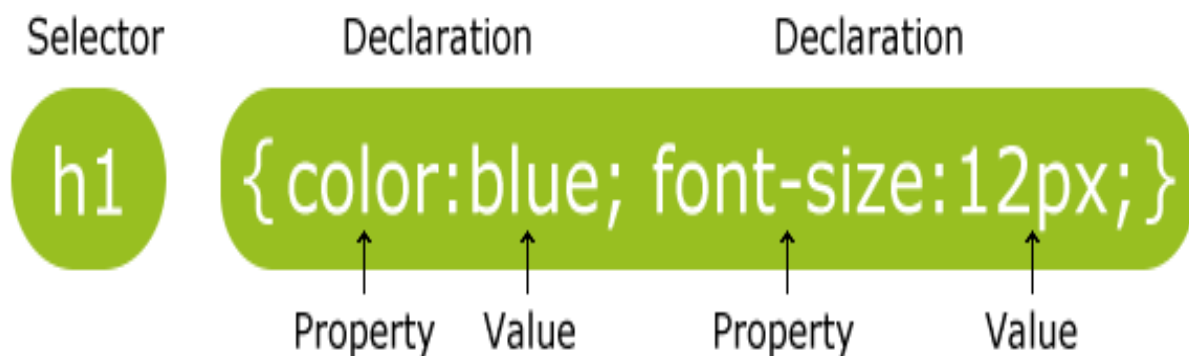
# CSS Saves a Lot of Work!

The style definitions are normally saved in external .css files.

With an external stylesheet file, you can change the look of an entire website by changing just one file!

## CSS Syntax

A CSS rule set 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 property name and a value, separated by a colon.

## CSS Example

A CSS declaration always ends with a semicolon, and declaration groups are surrounded by curly braces:

```
p
{
```

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

To make the CSS code more readable, you can put one declaration on each line.

In the following example all <p> elements will be center-aligned, with a red text color:

## Example

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

## 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 starts with /\* and ends with \*/. Comments can also span multiple lines:

## Example

```
p {
    color: red;
    /* This is a single-line comment */
    text-align: center;
}
```

```
/* This is
a multi-line
comment */
```

## CSS Selectors

CSS selectors allow you to select and manipulate HTML elements.

CSS selectors are used to "find" (or select) HTML elements based on **their id, class, type, attribute**, and more.

---

## The element Selector

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

You can select all <p> elements on a page like this (in this case, all <p> elements will be center-aligned, with a red text color):

### Example

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

## The id Selector

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

*An id should be unique within a page, so the id selector is used if you want to select a single, unique element.*

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

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

### Example

```
#lubak {  
    text-align: center;  
    color: red;  
}
```

## Class Work

<!DOCTYPE html>

```
<html>
<head>

<style>
#para1 {
    text-align: center;
    color: red;
}
</style>
</head>
<body>
```

```
<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>
```

```
</body>
</html>
```

## The class Selector

The class selector selects elements with a specific class attribute.

*To select elements with a specific class, write a period character, followed by the name of the class:*

In the example below, all HTML elements with class="center" will be red and center-aligned:

### Example

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

### Class Work

```
<!DOCTYPE html>
<html>
<head>
<style>
.center {
    text-align: center;
```

```
color: red;
}
</style>
</head>
<body>
```

```
<h1 class="center">Red and center-aligned heading</h1>
<p class="center">Red and center-aligned paragraph.</p>
```

```
</body>
</html>
```

You can also specify that only specific HTML elements should be affected by a class.

In the example below, all <p> elements with class="center" will be center-aligned:

## Example

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

## Class Work

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>
```

```
<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and
center-aligned.</p>
```

```
</body>
```

</html>

# Grouping Selectors

If you have elements with the same style definitions, like this:

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

you can group the selectors, to minimize the code.

To group selectors, separate each selector with a comma.

In the example below we have grouped the selectors from the code above:

## Example

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

## Class Work

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
h1, h2, p {  
    text-align: center;  
    color: red;  
}  
</style>
```

```
</head>
<body>

<h1>Hello World!</h1>
<h2>Smaller heading!</h2>
<p>This is a paragraph.</p>

</body>
</html>
```

## Three Ways to Insert CSS

There are three ways of inserting a style sheet:

- **External style sheet**
- **Internal style sheet**
- **Inline style**

## External Style Sheet

**With an external style sheet, you can change the look of an entire website by changing just one file!**

Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:

```
<head>
<link rel="stylesheet" type="text/css" href
="ourstyle.css">
</head>
```

An external style sheet can be written in any text editor. The file should not contain any html tags. The style sheet file must be saved with a .css extension.

An example of a style sheet file called "ourStyle.css", is shown below:

```
body {
    background-color: lightblue;
}

h1 {
```

```
    color: navy;  
    margin-left: 20px;  
}
```

## Internal Style Sheet

**An internal style sheet may be used if one single page has a unique style.**

Internal styles are defined within the <style> element, inside the head section of an HTML page:

### Example

```
<head>  
<style>  
body {  
    background-color: linen;  
}  
  
h1 {  
    color: maroon;  
    margin-left: 40px;  
}  
</style>  
</head>
```

### Class Work

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
body {  
    background-color: linen;  
}  
h1 {  
    color: maroon;  
    margin-left: 40px;  
}  
</style>  
</head>  
<body>
```

```
<h1>This is a heading</h1>  
<p>This is a paragraph.</p>
```

```
</body>  
</html>
```

## Inline Styles

**An inline style may be used to apply a unique style for a single element.**

An inline style loses many of the advantages of a style sheet (by mixing content with presentation). Use this method sparingly!

To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

The example below shows how to change the color and the left margin of a <h1> element:

### Example

```
<h1 style="color:blue;margin-left:30px;">This is  
a heading.</h1>
```

## Multiple Style Sheets

If some properties have been defined for the same selector (element) in different style sheets, the value from the last read style sheet will be used.

### Example

Assume that an external style sheet has the following style for the <h1> element:

```
h1 {  
    color: navy;  
}
```

then, assume that an internal style sheet also has the following style for the <h1> element:

```
h1 {  
    color: orange;  
}
```

If the internal style is defined after the link to the external style sheet, the <h1> elements will be "orange":

## Example

```
<head>  
<link rel="stylesheet" type="text/css" href="mystyle.css">  
<style>  
h1 {  
color: orange;  
}  
</style>  
</head>
```

## Class Work

```
<!DOCTYPE html>  
<html>  
<head>  
<link rel="stylesheet" type="text/css" href="mystyle.css">  
<style>  
h1 {  
    color: orange;  
}  
</style>  
</head>  
<body>
```

```
<h1>This is a heading</h1>  
<p>The style of this document is a combination of an external stylesheet,  
and internal style</p>
```

```
</body>  
</html>
```

However, if the internal style is defined before the link to the external style sheet, the <h1> elements will be "navy":

## Example

```
<head>
<style>
h1 {
color: orange;
}
</style>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

## Class Work

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
    color: orange;
}
</style>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>The style of this document is a combination of an external stylesheet,
and internal style</p>

</body>
</html>
```

## Cascading Order

What style will be used when there is more than one style specified for an HTML element?

Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number one has the highest priority:

1. Inline style (inside an HTML element)
2. External and internal style sheets (in the head section)
3. Browser default

So, an inline style (inside a specific HTML element) has the highest priority, which means that it will override a style defined inside the <head> tag, or in an external style sheet, or a browser default value.

## Class Work

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
<style>
body {background-color: linen;}
</style>
</head>
<body style="background-color: lightcyan">
```

```
<h1>Multiple Styles Will Cascade into One</h1>
<p>In this example, the background color is set inline, in an internal
stylesheet, and in an external stylesheet.</p>
<p>Try experimenting by removing styles to see how the cascading
stylesheets work. (try removing the inline first, then the internal, then the
external)</p>
```

```
</body>
</html>
```