



Introduction to CSS

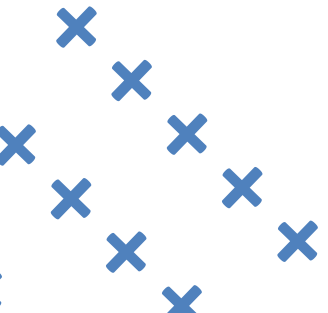


LESSON OBJECTIVES

- 
- Understand CSS and the basics of styling elements
 - Explore the various methods of styling
- 

What is CSS?

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on the website.
- This is what we will use to style the HTML



Styling methods



- Inline styling
- Internal styling
- External styling

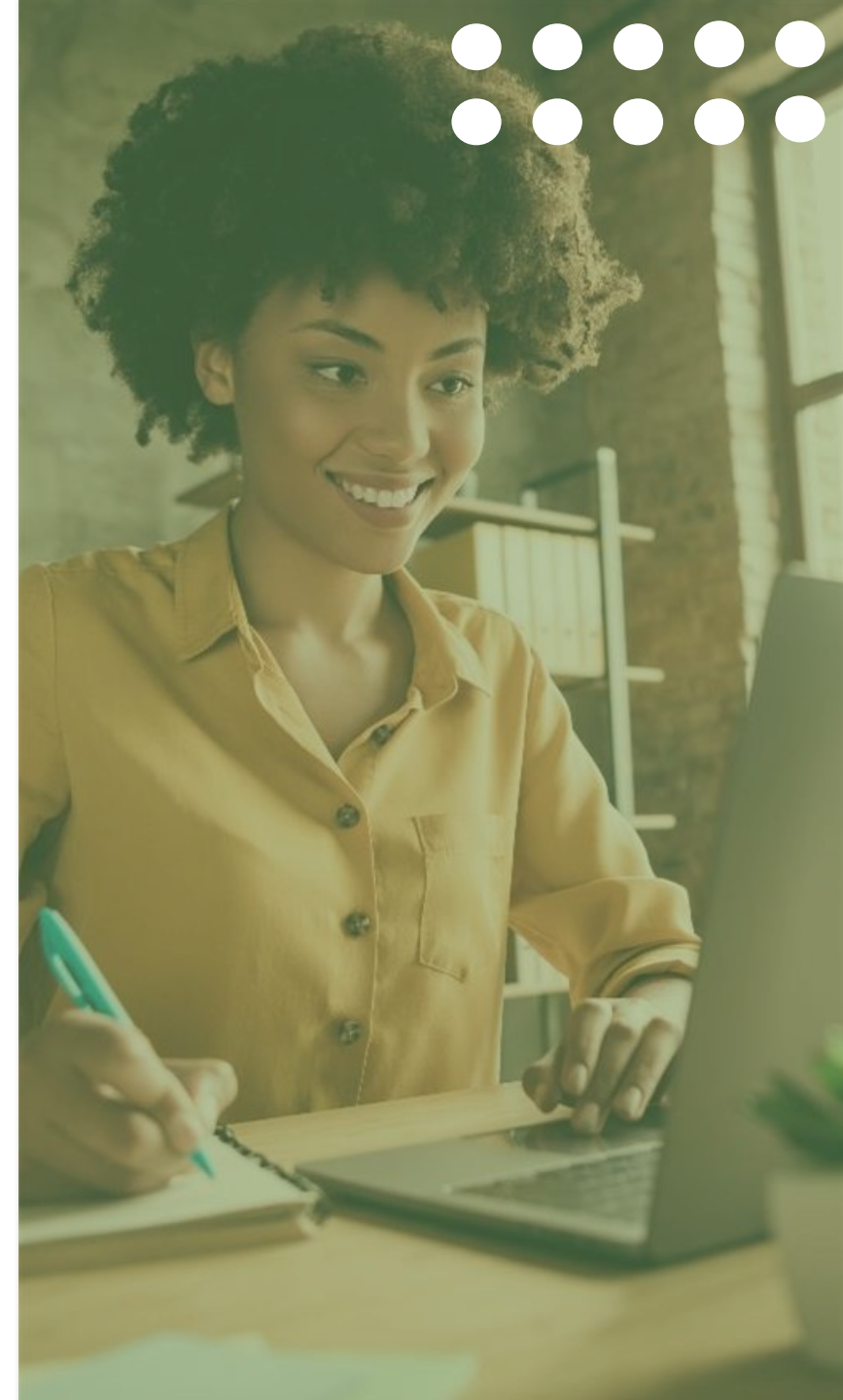
Why is External styling preferred?

- Code Duplication
- Project Maintenance - having to change styles on many elements

Inline Styling

```
<h1 style="">Heading 1</h1>
```

The style attribute appears inside of the opening tag

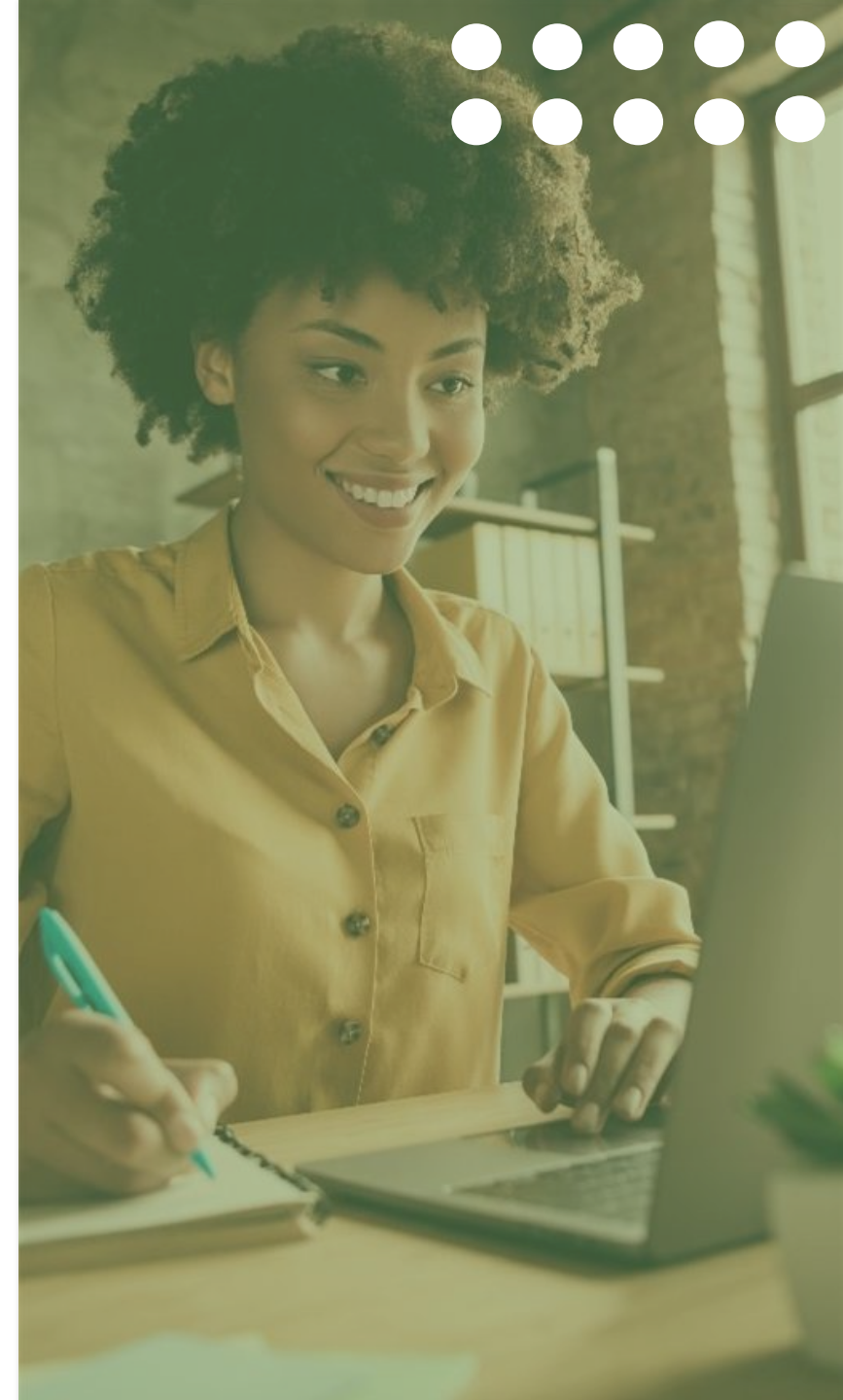


Internal Styling

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width
  <title>Document</title>
  <style>
  </style>
</head>
<body>

<!-- block and inline elements -->
<h1 style="">Heading 1</h1>
```

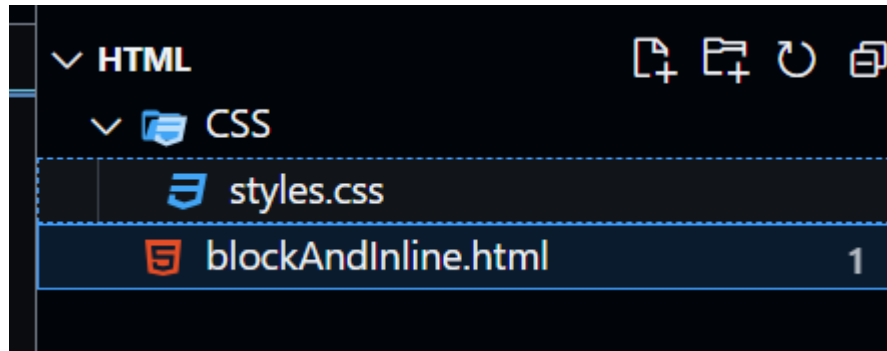
The style tag goes in the head or beginning/end of the body tag



External Styling

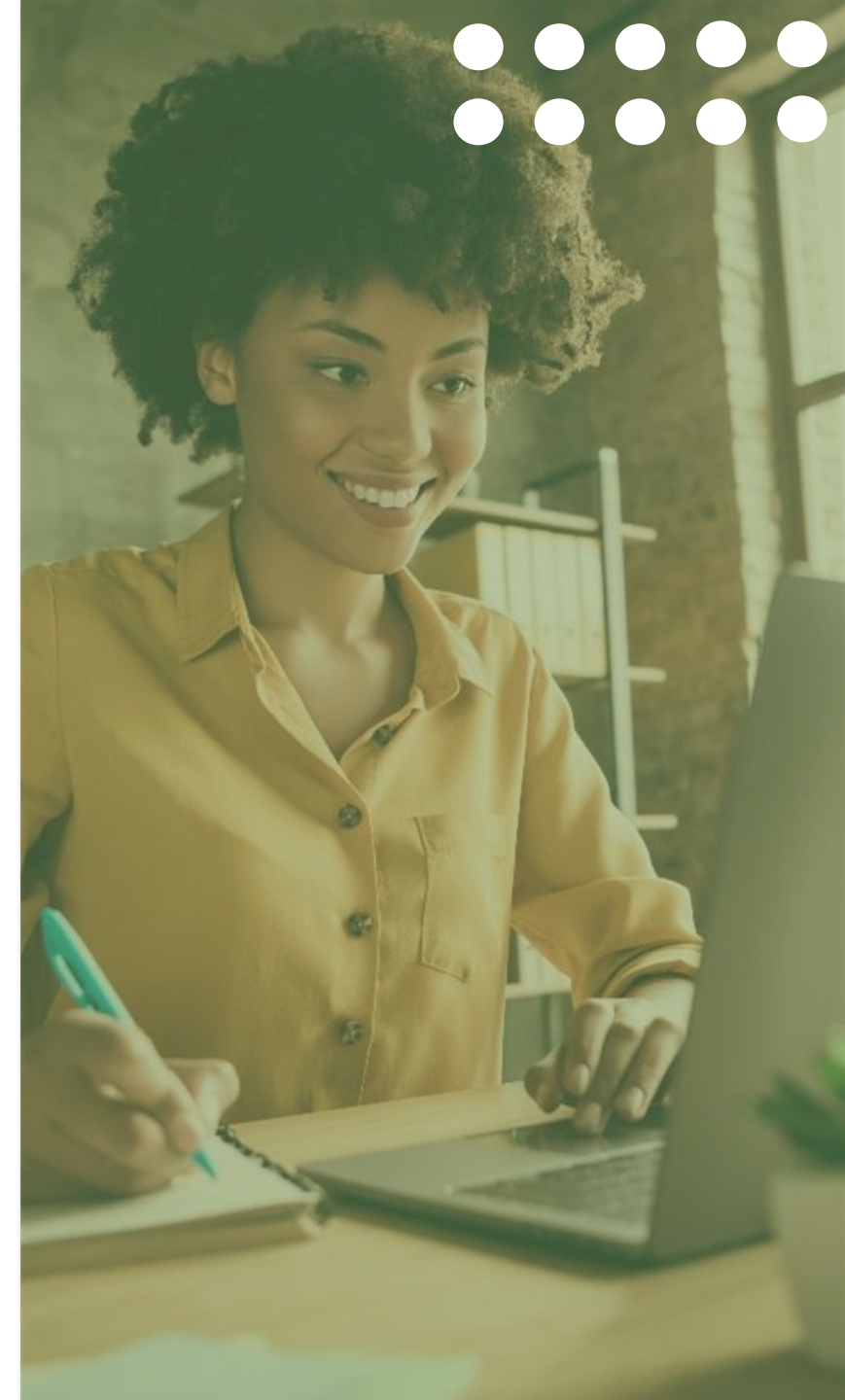
There is no style tag, however there is a link tag that goes in the head tag. This link tag links the external file sheet to the HTML.

External file in a CSS folder



Link tag in head tag

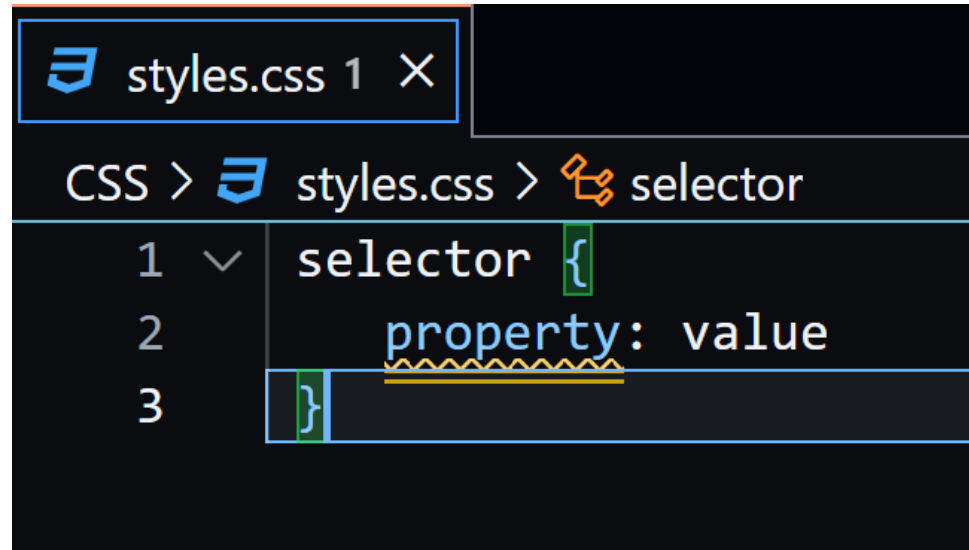
```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width
  <title>Document</title>
  <link rel="stylesheet" href="./CSS/styles.css">
</head>
<body>
```



Styling Syntax

- Selector
- Property
- Value

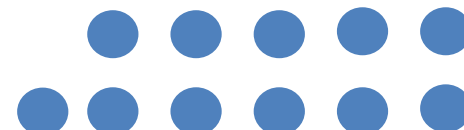
~Code example to follow ~



The screenshot shows a code editor window titled 'styles.css 1'. The breadcrumb navigation at the top reads 'CSS > styles.css > selector'. The code is as follows:

```
1 selector {  
2   property: value  
3 }
```

The code is syntax-highlighted: 'selector' is in white, '{' is in green, 'property' is in blue with a yellow wavy underline, 'value' is in white, and '}' is in green. A green cursor is positioned at the end of the opening curly brace on line 1.



Pseudo-classes and pseudo elements



- A pseudo-class is used to style an element only when a specific action occurs
selector: pseudo-class{

}
- A pseudo-element is used to style specific parts of an element
selector::pseudo-elements{

}

~Code example to follow~