

Lesson 2.1

By: Abdullah Abbas

CSS Introduction

CSS (Cascading Style Sheets) is a language that describes the style of an HTML document.

Introduction

Consider the following HTML example.

HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Example</title>
</head>
<body>

  <p>
    Some Text !
  </p>

</body>
</html>
```

Result

Some Text !

Introduction

Consider the following HTML example. **With CSS**

HTML

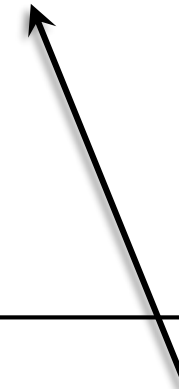
```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Example</title>
</head>
<body>

  <p style="color:blue;">
    Some Text !
  </p>

</body>
</html>
```

Result

Some Text !



Text color changed

Introduction

Consider the following HTML example. **With CSS**

HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Example</title>
</head>
<body>

<p style="color:blue; background-color:yellow; text-align:center;">
  Some Text !
</p>

</body>
</html>
```

Result

Some Text !

*This method of adding styles is called **Inline CSS**.*

CSS Different Methods

There are three methods for using CSS:

- **Inline CSS**

Like shown in previous examples, each element can have a *style* attribute.

- **Internal or Embedded CSS**

Styles are defined inside a *<style>* element, inside the *<head>* element.

- **External CSS (Recommended method)**

Styles are defined inside a separate CSS file.

- ✓ Easier to manage, by separating content (i.e. HTML) from style (i.e. CSS).
- ✓ The same CSS file can be used with several HTML pages.
- ✓ More powerful than inline CSS.

Internal CSS

How internal CSS work?

HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Example</title>

  <style>
    p {
      color: blue;
      background-color: yellow;
      text-align: center;
    }
  </style>
</head>
<body>
  <p>Some Text !</p>
</body>
</html>
```

Result

Some Text !

A CSS that describes a style for the <p> element.

Internal CSS

How internal CSS work?

HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Example</title>

  <style>
    p {
      color: blue;
      background-color: yellow;
      text-align: center;
    }
  </style>
</head>
<body>
  <p>Some Text !</p>
  <p>Another Text !</p>
</body>
</html>
```

Result

Some Text !

Another Text !

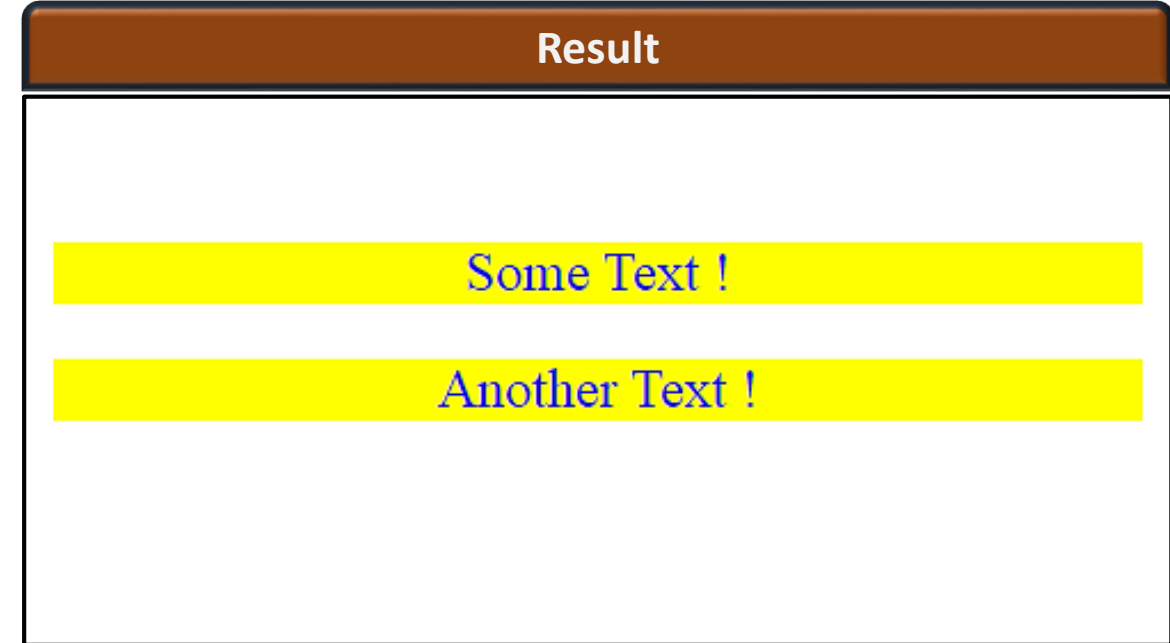
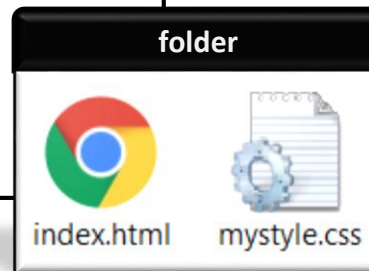
This formatting applies to all <p> elements in the HTML document !

External CSS

How **external CSS** work? We have two files (HTML file and CSS file).

```
index.html
<!DOCTYPE html>
<html>
<head>
  <title>HTML Example</title>
  <link rel="stylesheet" href="mystyle.css">
</head>
<body>
  <p>Some Text !</p>
  <p>Another Text !</p>
</body>
</html>
```

```
mystyle.css
p {
  color: blue;
  background-color: yellow;
  text-align: center;
}
```



The HTML file and the CSS file must be in the same folder for this example to work.

External CSS

(Example with Classes and IDs)

HTML

```
...  
<body>  
  <h1 class="colored" id="title1">Our Title</h1>  
  <p class="colored" id="paragraph1">Some Text !</p>  
  <p id="paragraph2">Another Text !</p>  
</body>  
...
```

CSS

```
p {  
  background-color: yellow;  
}  
  
h1 {  
  text-align: center;  
}  
  
.colored {  
  color: blue;  
}  
  
#paragraph2 {  
  font-family: helvetica;  
  font-weight: bold;  
}
```

Result

Our Title

Some Text !

Another Text !

- (p) All paragraphs has a yellow background.
- (h1) The title is centered.
- The title and paragraph 1 has a blue text color because they have a common class **colored**.
- Paragraph 2 has Helvetica font and it is bold because it is identified by the id **paragraph2**.

CSS Syntax (Summary)

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

Property1

Value

Property2

Value

Selector

- Can be: *h1*, *h2*, *p*, *table*, *th*, *td*, ...
- Can be: *** (style applies to all elements in the HTML document)
- Can be: a .*[class]* or an #*[id]* of an element.

What are class and id?

- Every element in the HTML document can optionally have a **class** attribute, and an **id** attribute.
- The **class** defines a category to which the element belong.
- The **id** uniquely identifies an element, which means two elements may not have the same **id**.
- In CSS we reference a class by using a **(.)** before its name, and an id by using a **(#)** before its name.

CSS Inheritance

HTML

```
...  
<body>  
  <h1>Our Title</h1>  
  <p>Some Text !</p>  
  <p id="paragraph2">Another Text !</p>  
</body>  
...
```

CSS

```
body {  
  color: red;  
}  
  
#paragraph2 {  
  background-color: yellow;  
}
```

Result

Our Title

Some Text !

Another Text !

- **(body)** All elements inside body inherits red text color because they are contained in it.
- Paragraph 2 has specific additional formatting.

NOTE: Not all properties are inherited.

Multiple Classes

An element can have multiple classes like the second <p> in the following example.

HTML

```
...  
<body>  
  <h1 class="colored">Our Title</h1>  
  <p class="colored">Some Text !</p>  
  <p class="colored highlighted">Another Text !</p>  
</body>  
...
```

CSS

```
.colored {  
  color: red;  
}  
  
.highlighted {  
  background-color: yellow;  
}
```

Result

Our Title

Some Text !

Another Text !

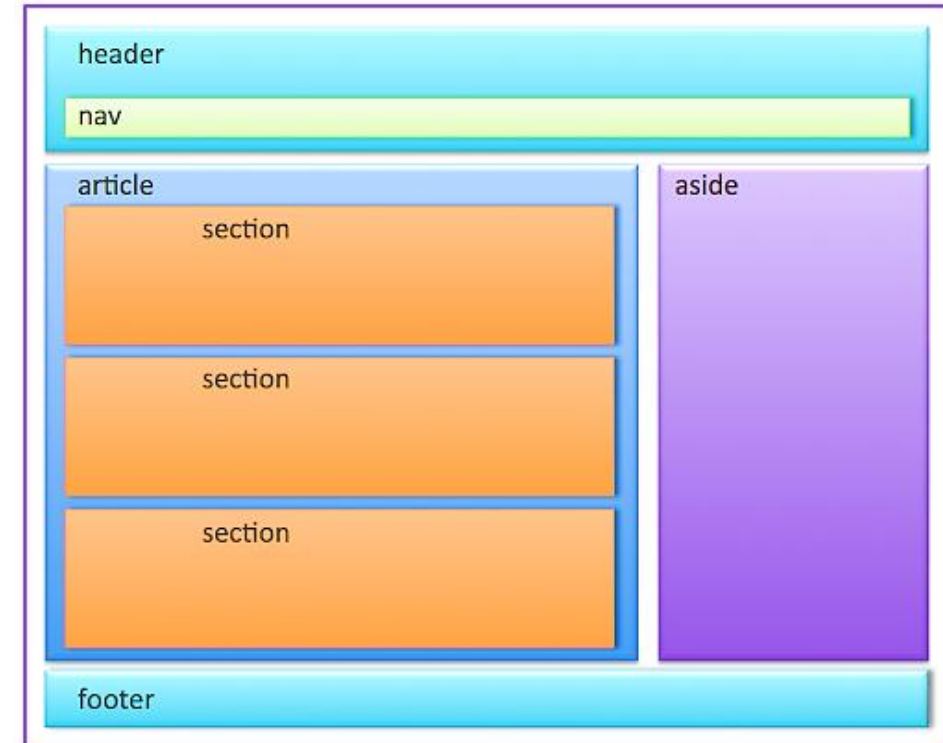
HTML5 Semantic Elements

What does your element represent?

HTML5 Semantic Elements

In HTML5 some elements were added, for example:

- **<header></header>:**
Everything inside it should be placed on the top of the document.
- **<nav></nav>:**
Contains links for navigation through the website.
- **<article></article>:**
Defines an article inside the document.
- **<section></section>:**
Defines a section inside an article or the document.
- **<footer></footer>:**
Everything inside it should be placed on the bottom of the document.
- **<aside></aside>:**
Represents a part that should be displayed on the side (left or right).



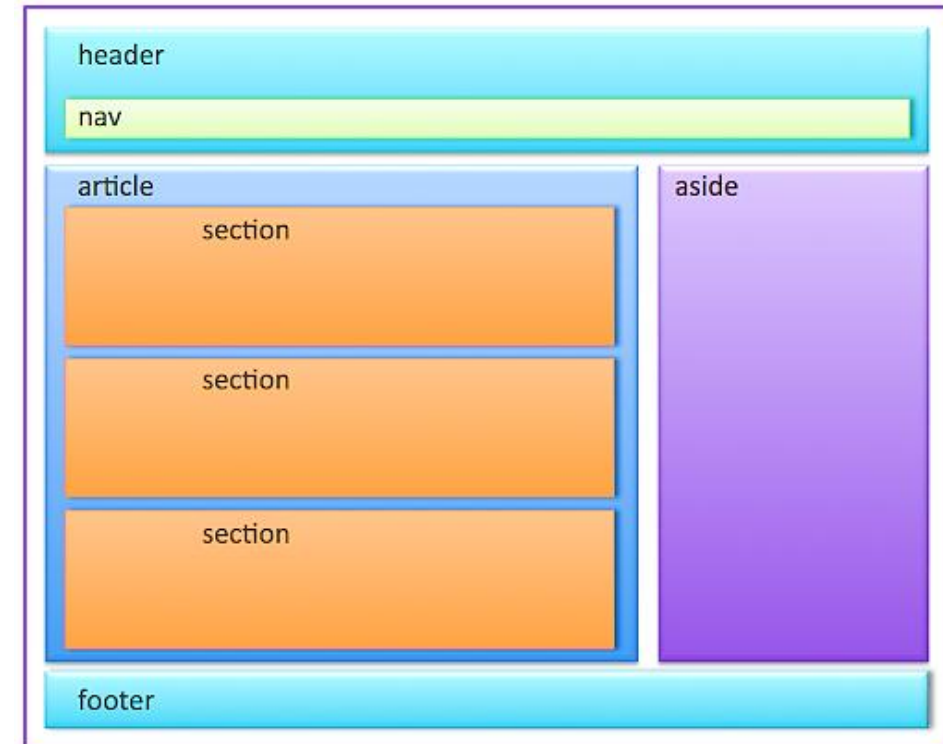
HTML5 Semantic Elements

These elements are semantic!

- They have a meaning, but they don't have any visual effect.

Why do we need them?

- They allow to structure the code and apply styles (CSS) on different parts.
- They help search engines and other machines to understand the page correctly for better accessibility.



General Purpose Elements

`<div>` and ``

Block Element

The `<div></div>` element is a general purpose element that allows you to add a part that is not one of the previous elements in HTML5. You give it an id/class attribute and you apply CSS style to it.

- It is a **block element**, which means it always starts on a new line.

Inline Element

The `` element is also a general purpose element that you can give an id/class attribute and apply CSS style to it.

- It is an **inline element**, which means it does not start on a new line.

Simple Example

HTML

```
...  
<body>  
  <header>  
    <h1> Our Title </h1>  
  </header>  
  
  <section>  
    <p> This text by <span id="vip">alice</span>.</p>  
  </section>  
  
  <div class="comments">  
    <p> Comment 1 </p>  
    <p> Comment 2 </p>  
  </div>  
</body>  
...
```

CSS

```
header { background-color: green; text-align: center; }  
  
section { font-size: 30px; }  
  
#vip { font-weight: bold; }  
  
.comments { font-style: italic; }
```

Result

Our Title

This text by **alice.**

Comment 1

Comment 2