**CSS Basics #1: First Steps -** [Eric Hu](https://huericnan.medium.com/?source=post_page-----43b285c9580--------------------------------) Feb 6, 2022

In our previous course on HTML, we briefly introduced the Cascading Style Sheet (CSS). To put it simply, it is something that can be used to define the appearance (colour, size, position…) of any element in an HTML document.

*You can download the source code for this tutorial here:*

[*Download the Source Code*](https://www.ericsdevblog.com/index.php/beginners-roadmap-to-web-development/)

**CSS Syntax**

In CSS, you can define a set of rules that can be applied to a particular element or a group of elements on your web page.

For instance, the following example defines that all the <h2> elements in the HTML document should be red and the size of the texts should be 5em, where em is a relative unit for font, 5em mean five times the size of the original font.

h2 {  
 color: red;  
 font-size: 5em;  
}

The h2 here is called a selector, it selects the HTML elements that we are going to style. And we have a set of different declarations wrapped inside the curly braces { }, which are in the form of "property: value" pairs.

**Applying CSS to HTML**

There are three different ways to apply CSS rules to an HTML document. Let’s consider this example:

<!DOCTYPE html>  
<html><head>  
 <meta charset="utf-8" />  
 <meta name="author" content="Eric Hu">  
 <title>My HTML Document</title>  
</head><body>  
 <h1>My HTML Document</h1> <h2>First Title</h2>  
 <p>...</p> <h2>Second Title</h2>  
 <p>...</p>  
</body></html>

Now, our objective is to make both second-level headings (<h2>) red and underlined using CSS.

**Inline CSS**

First, we can add the CSS declarations as attributes for the HTML elements like this:

<body>  
 <h1>My HTML Document</h1> <h2 style="color: red; text-decoration: underline;">First Title</h2>  
 <p>...</p> <h2>Second Title</h2>  
 <p>...</p>  
</body>

Graphical user interface, text, application, timeline

Description automatically generated

And as you can see, only the first heading is changed. So, in order to change both of them, we have to add the style attribute to both headings, which is a very time-consuming task for a large HTML document, so it is not the recommended method to insert CSS.

**Internal CSS**

We can also put the CSS code in the <head> section of the HTML document like this:

<head>  
 <meta charset="utf-8" />  
 <meta name="author" content="Eric Hu">  
 <title>My HTML Document</title> <style>  
 h2 {  
 color: red;  
 text-decoration: underline;  
 }  
 </style></head>

Graphical user interface, text, application

Description automatically generated

And as you can see, this time, both headings are changed. However, the downside of putting HTML and CSS together is that the document could easily get too large as your website grows, which is not good for future maintenance.

**External CSS**

The best way to deal with CSS and HTML is to put them in separate files, and then in the HTML document, we will reference the external CSS in the <head> section. For example, here we have two documents, myHTML.html and myCSS.css:

<head>  
 <meta charset="utf-8" />  
 <meta name="author" content="Eric Hu">  
 <title>My HTML Document</title> <link rel="stylesheet" href="myCSS.css">  
</head>h2 {  
 color: red;  
 text-decoration: underline;  
}

**About the DOM**

We already introduced the concept of the Document Object Model (DOM) in our “[HTML Basics](https://www.ericsdevblog.com/index.php/html-css-basics/)” course. The DOM has a tree-like structure. Every HTML element is a node in this tree structure.

When the browser displays a document, it first reads the document’s content (HTML) and converts it into a DOM tree, and then it will parse the style information (CSS) and assign it to the corresponding node of the DOM tree. The following graph from MDN Web Docs (<https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps/How_CSS_works>) illustrates this process.

Diagram

Description automatically generated

Let’s take a look at another example,

<body>  
 <h1>My HTML Document</h1> <div>  
 <h2>First Heading</h2>  
 <p>...</p>  
 </div> <div>  
 <h2>Second Heading</h2>  
 <p>...</p>  
 </div> <section>  
 <h2>Third Heading</h2>  
 <p>...</p>  
 </section>  
</body>

Diagram

Description automatically generated

DOM Tree

Here is the corresponding DOM tree. We can see that we have three block-level elements (two <div> elements and one <section> element) who are siblings to each other, and each of them has a heading and a paragraph.

Our objective is to locate the headings in the <div> blocks and make it red, and then find the heading under the <section> block, and make it blue.

div h2 {  
 color: red;  
}  
section h2 {  
 color: blue;  
}

That looks easy, right? We first locate both <div> elements and find the <h2> headings under the <div> blocks, and then in the declaration section, we set the colour to red. Next, we do the same for the <section> blocks.

Graphical user interface, text, application, email

Description automatically generated

However, as your HTML document gets bigger, this method may cause problems, because you are likely to use the same block-level element for different purposes, and you don’t want all of them to have the same style. So, the common practice is to select the class and id attributes instead.

**Selectors**

**Class, and ID Selectors**

This is the most common type of selector that we use in CSS. You select either a group of elements or a specific element based on its class or id.

/\* Class Selector \*/  
.box { }/\* ID Selector \*/  
#unique { }

Here is an example:

<body>  
 <h1>My HTML Document</h1> <div class="first-class">  
 <h2>First Heading</h2>  
 <p>...</p>  
 </div> <div id="uniqueID">  
 <h2>Second Heading</h2>  
 <p>...</p>  
 </div> <section class="first-class">  
 <h2>Third Heading</h2>  
 <p>...</p>  
 </section>  
</body>.first-class h2 {  
 color: red;  
}#uniqueID h2 {  
 color: blue;  
}

Graphical user interface, text, application, email

Description automatically generated

As you can see, it is possible for us to put different types of elements in the same class.

**Attribute Selectors**

You can also select elements based on the presence of a certain attribute.

<body>  
 <h2 id="firstHeading">First Heading</h2>  
 <h2 id="secondHeading">Second Heading</h2>  
 <h2>Third Heading</h2>  
</body>h2[id] {  
 color: red;  
}

This selector will find all the second-level headings that have an id attribute and set its colour to red.

Text

Description automatically generated with medium confidence

**Pseudo Classes**

Finally, there is something called a pseudo-class in CSS, which indicates a certain state of an element. The :hover pseudo-class, for example, selects an element only when the cursor hovers over it. Try this example in your own browser:

<body>  
 <h2>First Heading</h2>  
 <h2>Second Heading</h2>  
 <h2>Third Heading</h2>  
</body>h2:hover {  
 color: red;  
}