# **ELEMENT ,CLASS AND ID SELECTORS**

[MUSIC]

Play video starting at ::7 and follow transcript0:07

CSS selectors are used to determine which HTML element, or set of elements,

to apply the CSS declarations to.

The browser uses its selector API to traverse the DOM or

Document Object Model, and pick out the elements matching the selector.

Now crafting a selector is a great skill to have and not only for

styling using CSS.

That's because a lot of JavaScript libraries out there

use the browser selector API to attach behavior and data to HTML elements.

Much in the same way that CSS applies style to those elements.

In this lecture, we're going to talk about three different types of selectors,

element, class, and id selector.

Play video starting at ::48 and follow transcript0:48

First element selector,

the element selector is just you specifying the element name.

For example, in this case, p for

paragraph says that every paragraph in our HTML document should be colored blue.

Play video starting at :1:2 and follow transcript1:02

As an example, if we specified p paragraph and color blue, that means

every paragraph in the HTML document will have its content's text be blue.

Obviously, this really doesn't affect any other elements that contain text in them.

For example, we can have a div element that contains text but

that text will not be affected by our CSS rule

that specifies that the paragraph tag should have content text colored blue.

Play video starting at :1:31 and follow transcript1:31

Next is the class selector, and

the class selector is specified with a dot and the name of the class.

In this case again, we're creating a blue class,

a blue CSS class, that will color whatever it targets with the color blue.

Play video starting at :1:47 and follow transcript1:47

The class selector does require a little bit of a change to your HTML document

in that each element that you want that class applied to

has to have a attribute class equal to the class name.

So in this case we have a p, the paragraph tag and

div tag both have the attribute class="blue" and therefore,

their content or their text content will be colored blue.

Notice that a different paragraph that is not marked with attribute

class="blue" is completely unaffected by this.

Note the differences between the way you define a class and

then the way you use the class.

You define the class always with a dot in front of a name.

There cannot be any space between the dot and

the name of the class in case no space between the dot and the word blue.

However, when you use the class, you don't use the dot in its name,

you just use its name.

So here we have the paragraph tag with class="blue" without

the dot in front of it.

The last type of selector we're going to take a look at in this lecture is the id

selector.

The way you specify an id selector is by specifying the value of an id

of an element within your HTML document, preceded by a pound sign.

So the name is a value of some id within our HTML document.

So for example, if you have a div when id="name" and

then you specify your CSS rule with a selector #name, the CSS declarations for

that #name will apply to the content of the div element with its id value name.

Note that obviously everything else in the HTML document

is completely unaffected by this.

Similarly, in the way you define a class selector versus use a class selector,

the id selector is defined with a pound sign followed

by the value of an id within your HTML document.

And again, you can not have any space between the two.

But when you use it the pound goes away,

it's just the actual value of the id itself.

Now in order to write more efficient rules,

CSS allows us to group several selectors into one CSS rule.

Here we have two selectors grouped together, div and the selected div for

the div element, and a selector class blue grouped into one,

since they both share the same CSS declaration.

And you can see, in our HTML, the blue colored text will appear in the paragraph

tag that has the attribute class="blue" and in any div tag in the HTML document.

So let's take a look at an example of using these selectors.

Play video starting at :4:24 and follow transcript4:24

Okay, so here I am in Sublime Text and

I'm looking at the file simple-selectors-before.html and

it's located in the examples, Lecture 13 folder.

Let me go ahead and review the browser and

we'll even close the sidebar so we can see the browser and the code at the same time.

So let's scroll down and take a look at what our HTML structure is like.

So if you take a look at the left side of the screen, which is the Sublime Text.

On the right side of the screen you could see that we have the h1 tag right here,

which is this heading.

We have a couple of h2 headings which are called Subheading 1, Subheading 2.

We have a few paragraphs.

A couple of those paragraphs have a class attribute set it to highlight.

And the last thing here on page, we have a div that has a span

inside of it with an id attribute ="mainPoint".

So let's go ahead and scroll up and take a look as to what we'd like to do here.

So first of all we want all h2 elements to be color red and aligned in the center.

So in order to do that,

we have to substitute these XXX with some sort of selector.

Well, this is very easy.

That's just the h2 element selector, that's all it is.

So once we do that and we save it and refresh our browser,

we can see that both h2 headings became red and centered.

Play video starting at :5:44 and follow transcript5:44

Next, we want to affect every single element that has an attribute

whose class is ="highlight".

Well, in order to do that we have to define the highlight class.

And so what we're going to do here is apply a bigger font size,

we'll make it bold, italic.

The background of the entire element should turn green.

And also we're going to turn the opacity down a little bit from 1,

1 would be 100%, and .6 is 60%.

So the way we define a class

selector is by saying .highlight, and here's our class selector.

Notice, there is not space between the dot and the highlight.

So if we go ahead and refresh.

So we have two paragraphs here that had their styles affected by that.

And that is because if you scroll down, we have a couple of paragraphs,

both with class ="highlight".

Next, let's see, we need to have the element with id="mainPoint".

So the way we specify an id selector is by saying pound sign and

the name of the id which is mainPoint.

And the upper and lowercase letters here do matter.

And again, there is no space between the pound sign and the word mainPoint.

So if we go ahead, come over here, and refresh, we could see that

the span that surrounded the words id="mainPoint" became red.

And it became a little bit of a bigger font size, and

opacity's turned down, and the font is bold.

Last, we wanted to take all elements that are p and elements h1, and

we want to group their styles such that we don't have to specify them separately.

So the way we do that is very simply by saying, p, h1.

So we grouped the styles for the p element and the h1 element into one.

So every p element now and

every h1 element should now turn its color to blue and align center.

So we'll go ahead and save this document, refresh the browser.

And here we go, every p element and every h1,

which there's only one, here turned blue and aligned in the center.

So one thing to notice here is that out of all the selectors, the element selector,

the class selector and the id selector, the id is the least reusable one.

That makes sense, right?

Since by HTML rules,

a particular id attribute value can only appear once in a document.

So for example, the span with id attribute ="mainPoint", I can't really

reuse the value mainPoint as an id for any other element in this HTML document.

So as you plan your styles, it's good to keep this in mind and not basically rely

on assigning ids to every single element in the entire HTML document.

Because that way your CSS will not be very efficient.

You'll have to write specific rules for every element, and

you won't really be able to generalize and reuse your CSS styles very easily.

Play video starting at :8:42 and follow transcript8:42

In summary, we spoke about the syntax of simple CSS selectors.

We spoke about the element selector, the class selector, that you remember to

define with a dot right in front of the name, without the space.

And the id selector, that you can find with a pound sign, right before

the value of the id and the HTML page, again, without the space in between.

Next, we're going to talk about combining selectors

to create a more targeted selection within the document object model.