# **COMBINING SELECTORS**

[MUSIC]

Play video starting at ::5 and follow transcript0:05

Combining Selectors is a very powerful technique that allows you to more

precisely target dumb elements.

In this lecture we're going to take a look at some of the most common ways of

combining CSS selectors The first one is with element with class selector.

As you can see here we have a P element followed immediately without any spaces by

a dot big which is a A class selector.

And what this expresses is that I want to target every p element

that has an attribute class equal to big.

Play video starting at ::34 and follow transcript0:34

Note the lack of space between the element that the class definition.

If you do put a space it will mean a completely different combination so

make sure that if you want an element that has a class assigned to a particular value

that there's no spaces between the first selector and the second selector

in this case between the P element selector and the .big class selector.

Play video starting at ::59 and follow transcript0:59

And, as an example, here you can see on the right that p class="big"

will get to text size 20px, as opposed to all the other elements that have

a class="big" will not get that same 20 pixel font size styling.

This technique is pretty commonly used when you have a CSS class definition

that applies to different elements but when it's applied to a particular element

you would like the styling to slightly change.

Play video starting at :1:27 and follow transcript1:27

Next type of combination of selectors is the child selector.

And its syntax is pretty straightforward.

It's a selector, followed by a right-angle bracket, followed by another selector.

And the way you read these combinations is from right to left.

And in this case, what we're trying to do is target every

P element that is a direct child of an article element.

So in this example, you can see that the P element that is a direct child of

an article element will get the styling of color blue applied to it,

as opposed to a regular P element that will obviously be unaffected.

Affected, but also a P element that is inside an article element but

not a direct child of it, so there's a div element that's contained

within the article element and only then is the P element contained within the div.

So since the P element is not a direct child of the article in the second case,

its font styling is unaffected.

Play video starting at :2:22 and follow transcript2:22

Another type of selector id is a descendant selector.

And the syntax of it is, selector space another selector.

And just like before you read it from right to left.

So in this case what we're trying to do is target every P element

that is inside at any level.

Of an article element, which means that even if it's not a direct child,

even if it's deep down, as long as one of its parents in its ancestry

is an article, this rule will apply.

Play video starting at :2:53 and follow transcript2:53

So for example.

In this case, it is a direct child of an article, and the text will turn blue.

In this case, it's going to be completely unaffected,

because it's not inside of article element altogether.

And in this case, it will still turn blue, that even though it's not

a direct child or element, since we're using a descendant, selector combination.

The P element is still technically inside the article even though it's not a direct

child and will have its text turn blue.

Now, I want to point out that these selector combinations aren't limited

to element selectors even though that's what I've used in the examples.

So for example we could have a .colored P and that will mean again reading it

from right to left that every p element that is inside at any level of an element

with class equals colored will have its text color turn to blue.

In the case on the right, we're using a direct child selector to say that

every element with the attribute class equal to colored that is a direct

child of an article element will have its text turn to blue.

Let's take a look at a few coding examples to see these combinations in action.

Okay so I'm in sub-line text and

I'm looking at a file called element-with-class-befrore.html and

it's located in the examples, lecture 14 folder.

Let me quickly rearrange the screen so you can see the browser and

the code editor better.

Play video starting at :4:15 and follow transcript4:15

Okay, so you can see we have a pretty simple html document.

Let's take a look at it's structure we have in the h1 tag whose

class is highlight.

We have a couple of paragraph tags, one just the regular paragraph tag and

another one has a class highlight applied to it.

And we have a div tag that has actually two classes applied to it

called main point and highlight.

And notice the way you apply multiple classes to an element

is just by listing them in the class attribute with a space between them.

Okay so let's go ahead and write some selector rules.

So the first one we want to do is want to target everything with class highlight.

Well that's very easy, we've done it before.

That's just .highlight.

We'll save and we'll preview the difference in the browser and we see that

everything with a class highlight does in fact have the background turned to green.

Next, we'll look at all p elements with class highlight.

Well we've done this before.

So we need to do all P element but we can't put a space we'll just have to

put .highlight right after that, which will mean all P elements

that have a class highlight assigned to them and we want to turn them into italic.

So let's save that and refresh the browser.

And we could see that the P element,

that has a class equal to highlight, did in fact, turn italic.

Next, we'll combine two class elements.

In this case, we want all the elements with class highlight,

as well as with class main point.

If both of those classes get applied to the same element,

we want that element to turn red, and its background color to turn black.

So the way we could specify that, it's just to say, .highlight.mainpoint,

that is to say that an element that has highlight class and also has

a mainpoint class assigned to it should be colored red and background black.

Let's go and refresh this, and sure enough, the last element on the page,

the div that has both classes assigned to it

has its styling affected in the way we want it.

Next let's take a look at the file called child selector before that HTML.

Let's go ahead and quickly examine its HTML structure.

It has a couple of section tags.

The first section tag has an H2 tag called sub heading one, a div tag and

inside the div tag we have the article tag.

In the second section,

we have an article tag that is a direct child of this section tag.

And the sub heading two which is an H2 element.

Is sitting outside of the section tag altogether.

So let's take a look as to what we need to do here.

In the first rule we want to target any article element

which is a direct child of a section element.

In other words any article whose immediate parent is a section element.

Well we've done this before in examples, so

that section with the right angle bracket to say immediate child and

it's immediate child should be, article, so we save that and refresh the browser.

We see that the styling is only applied to the article that is direct

child of the section but the article that is not direct child of a section who's

sitting inside the DIV is unaffected.

The next target is, any h2 element which is a direct child of a section element.

In other words, any h2 element whose immediate parent is a section element.

So again, the idea is similar It's section and it's direct child should be h2.

So any h2 that is a direct child of a section element.

Let's save that and refresh.

And since subheading one is a direct child of the section element.

That is the only element that gets our styling rule applied to it.

Next, let's take a look at the file code descendant selector before that HTML.

Let's quickly examine it's HTML structure.

We have a couple of unordered lists here.

The first unordered list is sitting inside the section element and

the second unordered list is sitting inside the article element.

So let's take a look at our CSS rule.

What we want to do is we want to target all Li elements

that are inside of a section element.

That is, no matter how deep they are in the element tree

underneath the section element, we want the style of color green applied to it.

So that sounds like we need to use a decendant selector.

And the way we do that is to say section space and li, and as you can see,

this selector combination should target this first list.

And even though ul separates the li elements from the section element,

the style should still apply since we used a descendant selector combination.

So let's go ahead and refresh the browser.

And you can see that all the li elements of the first list

turn green just as we expected.

So to summarize we looked at different ways of combining selectors.

We looked at element with class selector which is a selector

without a space followed by class definition.

We looked at child direct selector which is a selector with an angle bracket

followed by another selector.

And we looked at a descendent selector which is a selector with a space followed

by another selector.

And notice that I'm using the word selector not necessarily element

just to remind you that it doesn't have to be an element.

It can be any selector that is used in combination with another selector.

There are a couple of combination selectors that we did not cover.

For example, one is a JSON selector and another one is general sibling selector.

Now these selector combinations are a little bit less common then the other ones

that I showed you so I will leave it up to you to look it up in the specification or

a line as to how they work.

Next we're going to talk about Pseudo Class Selectors.