MODULE 01 - 012: SCSS :Pseudo-Selector Nesting

This guide explains how to use nested pseudo selectors in SCSS, with an example of customizing a link's hover state.

What are Pseudo Selectors?

Pseudo selectors allow you to apply styles to elements based on specific states or behaviors.

In adittion, **nesting pseudo-selectors** ensures that styles for pseudo-selectors are clearly linked to their parent selectors.

Step-by-Step Example

Here's a scenario where we style a subheading that contains a link.

1 Initial HTML Structure:

```html
<div class="container">
  <div class="page-wrapper">
    <div class="featured">
      <h1>About Us</h1>

      <div class="subheading">
        <h4>
          <a href="#">My subheading</a>
        </h4>
      </div>
    </div>

    <div class="page-content">
      <div class="container">
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
      </div>
    </div>
  </div>
</div>
```

2 SCSS Code:

```scss
$off-white: #f6f6f6;
$featured-color: DarkRed;

body {
  background-color: $off-white;
  height: 100vh;
  width: 100vw;
}

.container {
  font-family: Verdana;
  font-size: 0.8rem;
}

.page-wrapper {
  padding: 21px;
  $featured-color: LightCoral;

  .featured {
    color: $featured-color;
```

```scss
    .subheading a {
      color: CornflowerBlue;
      text-decoration: none;

      &:hover {
        color: DarkOliveGreen;
        text-decoration: underline;
      }
    }
  }

  .page-content {
    background-color: $featured-color;
    padding: 42px;
    color: $off-white;

    .container {
      font-family: Courier;
    }
  }
}
```

Observations

1. Nesting pseudo-selectors:
    - Use the `&` symbol followed by the pseudo-selector (e.g., &:hover).
    - The `&` references the parent selector, ensuring the hover styles apply specifically to the link.
    -
    - The `a tag` inside *.subheading* is styled with a specific color (CornflowerBlue).
    - On the hover (`&:hover`), the color changes to DarkOliveGreen, and an underline is added.

Video lesson Speech

SCSS Pseudo Selector Nesting

This guide walks through how to work with nested pseudo selectors in SCSS files, specifically examining how to customize a link's hover state.

So far we've discussed how we can implement nesting in SCSS and how we can use that to be able to organize our class definitions and make sure that the styles that we're defining are the ones we actually want to implement inside of the HTML doc. So what we're going to talk about in this guide is how we can implement nesting with pseudo selectors.

Pseudo selectors give you the ability to select the style that you give an element given a different state than just the standard one. For example, if we decide to make a subheading into a link, By using an tag that will create the link.

And so this text is going to be nested inside of this link. And as you can see right here we have my subheading.

Now, this is pretty ugly. This is the standard HTML blue with the underline and it's exactly what you would get just out of the box with HTML. So let's work on that. The first thing we're going to do is change the subheading color. And the reason why this color is not working anymore because you can see subheading is still nested inside of feature. We didn't change that part. But what's different now is we created an tag. So now there's another element that's actually inside of our subheading.

So whenever we style links we actually have to define the way we're styling them and we have to tell SCSS that we're going to define the style for a link. So now if I add in an a right here right after subheading you can see that the color has now changed to red.

I'm going to update this again and we're going to change it to cornflower blue. I have no idea how they came up with some of the names that they have. The HTML full set of color guides has some very interesting names and I like to play with

these. So here we have cornflower blue. Now if I hover over you can see nothing changes. So first and foremost I want to take away the underline. So the way I can do that is by saying text-decoration and say none here and now you can see that the underlining disappears.

Right now there's nothing happening but when I hover, I want to change the color and then I also want to add the underlining back in. So the way that you can do that is to perform another set of nesting. The way that you use a pseudo selector is by starting with an ampersand followed by a colon and then say hover and then inside of that, you give your different style definition.

If you hover over the link now, we can see that the changes have taken place.

Now, this is one that may look kind of weird right now. One of the main reasons I wanted to put it in the course one is because it's important and helpful to use but also as you're going through other people's SCSS files if you come across something that looks like this that may look a little bit weird. If you've never seen it before so I'd definitely recommend trying this out on a few different projects that you're working on.
HTML Code from Guide

```html
<div class="container">
  <div class="page-wrapper">
    <div class="featured">
      <h1>About us</h1>

      <div class="subheading">
        <h4>
          <a href="#">My subheading</a>
        </h4>
      </div>
    </div>

    <div class="page-content">
      <div class="container">
        <p>Cras mattis consectetur purus sit amet fermentum. Aenean eu leo quam. Pellentesque ornare sem laci
      </div>
    </div>
  </div>
</div>
```

SCSS Code Example of Pseudo Selector Nesting

```scss
$off-white: #f6f6f6;
$featured-color: DarkRed;

body {
  background-color: $off-white;
  height: 100vh;
  height: 100vw;
}

.container {
  font-family: Verdana;
  font-size: 0.8rem;
}

.page-wrapper {
  padding: 21px;
  $featured-color: LightCoral;

  .featured {
    color: $featured-color;
```

```scss
  .subheading a {
    color: CornflowerBlue;
    text-decoration: none;
    &:hover {
      color: DarkOliveGreen;
      text-decoration: underline;
    }
  }
}

.page-content {

  background-color: $featured-color;
  padding: 42px;
  color: $off-white;
  .container {
    font-family: courier;
  }
}
}
```

[SPA]
Anidando *:pseudo-selectors.*

Los pseudo selectores te permiten aplicar estilos a elementos en función de estados o comportamientos específicos.

Además, la anidación de pseudo selectores asegura que los estilos de estos estén claramente relacionados con sus selectores principales. "'