Nesting

In this lesson, we'll be introducing the SASS nesting syntax.

WE'LL COVER THE FOLLOWING

When you observe the structure of an HTML file, you'll notice it has a very clear hierarchy:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

As you can see, HTML has a structure that makes it quite easy to read.

CSS, on the other hand, lacks this visual structure. Which is why it has a tendency to become disorganized quite quickly. Enter Sass **nesting**!

Definition

Using nesting, we can nest child selectors inside of the parent selector.

This results in much cleaner and less repetitive code.

Example

Example

Take the following HTML:

```
<nav class="navbar">
  <ul>
    <li>Home</li>
    <li>Store</li>
    <li>Contact Us</li>
  </ul>
</nav>
```

Using regular CSS, we would write this like so:

Output

HTML

CSS (SCSS)

```
.navbar {
  background-color: orangered;
  padding: 1rem;
}
.navbar ul {
  list-style: none;
}
.navbar li {
  text-align: center;
  margin: 1rem;
}
```

There's a lot of repetition here. Each time we want to style a child of `navbar`, we have to repeat the class name.

With Sass, we can write much cleaner code.

Like so:

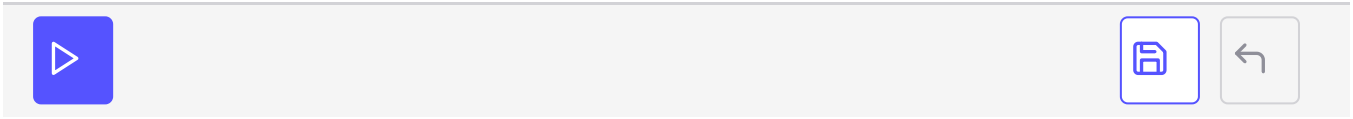
Output

HTML

CSS (SCSS)

```
.navbar {
  background-color: orangered;
```

```
  padding: 1rem;
  ul {
    list-style: none;
  }
  li {
    text-align: center;
    margin: 1rem;
  }
}
```



Using indentation, you can now see the `ul` and `li` selectors are neatly nested inside the `navbar` selector.

We have a much less repetitive syntax, which is far easier to read! Working with stylesheets is about to get fun!

Next up, we'll be looking at mixins.