Nesting

Sass lets you nest CSS selectors in the same way as HTML.

Look at an example of some Sass code for a site's navigation:

```
nav{
 ul{
margin:0;
padding:0;
list-style:none;
}
li{
display:inline-block;
}

  a {
    display: block;
    padding: 6px 12px;
    text-decoration: none;
  }
}
```

The css output

```
nav ul {

  margin: 0;

  padding: 0;

  list-style: none;

}

nav li {

  display: inline-block;

}

nav a {

  display: block;

  padding: 6px 12px;

  text-decoration: none;

}
```

you can also nest the property

```
font: {

  family: Helvetica, sans-serif;

  size: 18px;

  weight: bold;

}


text: {

  align: center;

  transform: lowercase;

  overflow: hidden;

}
```

the css is
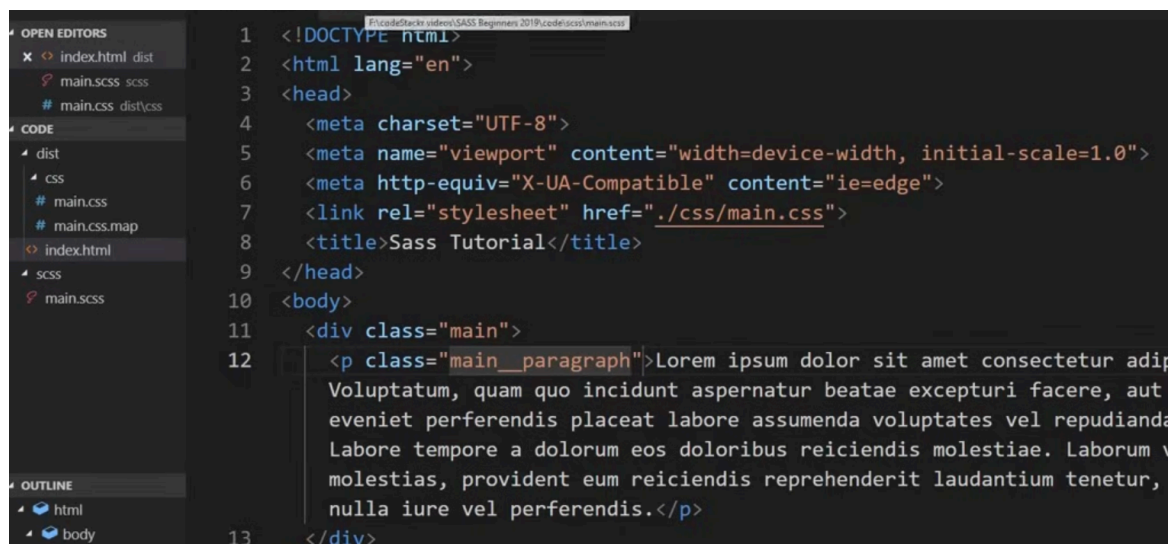
font-family: Helvetica, sans-serif;

font-size: 18px;

font-weight: bold;

text-align: center;

text-transform: lowercase;

text-overflow: hidden;

& represents the parent name

```scss
10    body {
11      background: $primary-color;
12      color: $text-color;
13    }
14
15    .main {
16      width: 80%;
17      margin: 0 auto;
18
19      #{&}__paragraph {
20        font-weight: map-get($font-weights, bold);
21
22        &:hover {
23          color: pink;
24        }
25      }
26    }
```

#{&} will take the main class and also provide the name

the converted css

```css
1    body {
2      background: #272727;
3      color: #fff;
4    }
5
6    .main {
7      width: 80%;
8      margin: 0 auto;
9    }
10
11    .main .main__paragraph {
12      font-weight: 700;
13    }
14
15    .main .main__paragraph:hover {
16      color: pink;
```

More videos