

Note: this feature is available with `react-scripts@2.0.0` and higher.

This project supports [CSS Modules](#) alongside regular stylesheets using the `[name].module.css` file naming convention. CSS Modules allows the scoping of CSS by automatically creating a unique classname of the format `[filename]__[classname]__[hash]`.

Tip: Should you want to preprocess a stylesheet with Sass then make sure to [follow the installation instructions](#) and then change the stylesheet file extension as follows: `[name].module.scss` or `[name].module.sass`.

CSS Modules let you use the same CSS class name in different files without worrying about naming clashes. Learn more about CSS Modules [here](#).

Button.module.css

```
.error {  
  background-color: red;  
}
```

another-stylesheet.css

```
.error {  
  color: red;  
}
```

Button.js

```
import React, { Component } from 'react';  
import styles from './Button.module.css'; // Import css modules stylesheet as styles  
import './another-stylesheet.css'; // Import regular stylesheet  
  
class Button extends Component {  
  render() {  
    // reference as a js object  
    return <button className={styles.error}>Error Button</button>;  
  }  
}
```

Result

No clashes from other `.error` class names

```
<!-- This button has red background but not red text -->  
<button class="Button_error_ax7yz">Error Button</button>
```

This is an optional feature. Regular `<link>` stylesheets and CSS files are fully supported. CSS Modules are turned on for files ending with the `.module.css` extension.