

Styling components

tl;dr

- The bad news: There are lots of methods to style and all of them feel kludgy in different ways
- The good news:
 - Import a CSS file to create global styles
 - JavaScript objects can live locally in a component
 - ... or we can import them from a .js file
 - Open source style libraries can be npm installed
 - All the above techniques can be used together

5 methods to style React components

1. Importing traditional CSS files
2. Inline JavaScript styles
3. Importing JavaScript modules
4. npm libraries
5. Hybrids of the above

Importing CSS files

```
import 'react-mdl/extra/material.css';
```

- Note: no "from"
- When you do this it flows DOWN only to inner components, but not up.

Then you're using it just like you do
with regular CSS

```
<div className="big fancy">...</div>  
<ul className="no-bullets">...</ul>  
<input className="form-control" />  

```



**Remember, you can't
use 'class'. You must
use 'className'**

Inline styling

Box.js

```
import React from 'react';

const divStyle = {
  margin: '40px',
  border: '5px solid pink'
};

export function Box() {
  const pStyle = {
    fontSize: '15px',
    textAlign: 'center'
  };
  return (
    <div style={divStyle}>
      <p style={pStyle}>
        Text goes here
      </p>
    </div>)
  )
}
```

- Seems so low-level
- So non-SOC/non-SRP!

... But this is Facebook's
recommended method.

Importing JavaScript CSS files

styles.js

```
export const pStyle = {  
  fontSize: '15px',  
  textAlign: 'center'  
};  
// As many more styles as you like here
```

First, create a JavaScript file with
as many styles as you like

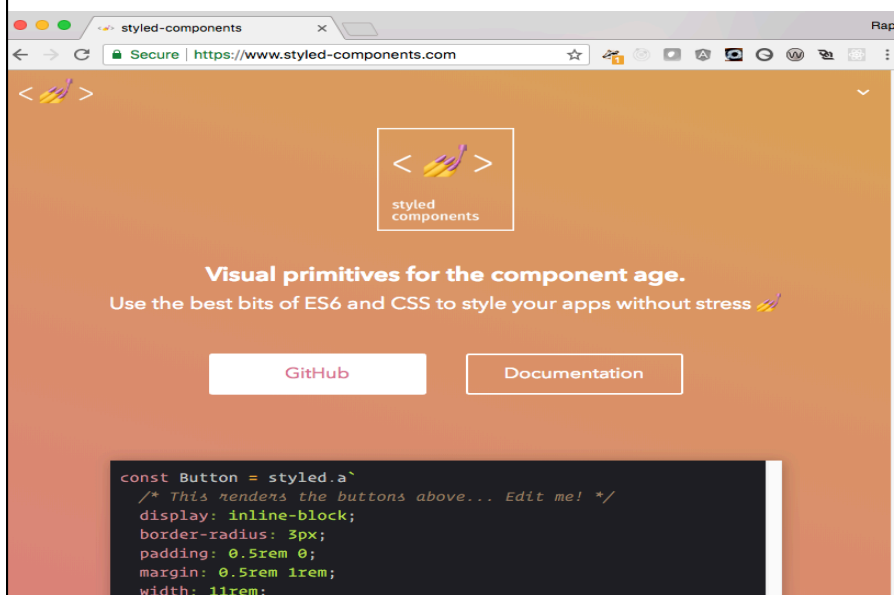
... then import the file and use the
styles.

Box.js

```
import pStyle from './styles.js';  
export function Box() {  
  return (  
    <div className="alert">  
      <p style={pStyle}>  
        Text goes here  
      </p>  
    </div>  
  )  
}
```

npm Libraries

<https://github.com/styled-components>



```
import React from 'react';
import styled from 'styled-components';

// Create a <Title> react component that renders an <h1>
const Title = styled.h1`
  font-size: 1.5em;
  text-align: center;
  color: palevioletred;
`;

// Create a <Wrapper> react component that renders a <section>
const Wrapper = styled.section`
  padding: 4em;
  background: papayawhip;
`;

// Use them like any other React component – except they're styled!
<Wrapper>
  <Title>Hello World, this is my first styled component!</Title>
</Wrapper>
```

Images

- node_modules
- ▾ public
 - ▾ images
 - 🖼️ meme.jpg
 - ★ favicon.ico
 - 🔗 index.html
 - { } manifest.json
- ▾ src
 - # App.css
 - JS App.js
 - JS App.test.js
 - # index.css
 - JS index.js
 - 🖼️ logo.svg
 - JS Person.js

- Static resources should be put under public. Including images!
- Say we have a folder under public called images. We put an image file in there:
- Then reference them relative to that public folder.

```

```

tl;dr

- The bad news: There are lots of methods to style and all of them feel kludgy in different ways
- The good news:
 - Import a CSS file to create global styles
 - JavaScript objects can live locally in a component
 - ... or we can import them from a .js file
 - Open source style libraries can be npm installed
 - All the above techniques can be used together