

CSS Styled Components in React

Wednesday, September 2, 2020 12:08 PM

```
40 // backgroundColor: theme.buttonBgHover,
41 // color: theme.buttonTextHover
42 }
43 };
44
45 return (
46   <Fragment>
47     /* <Global styles={globalStyles} /> */
48     <div
49       // css={containerStyles}
50     >
51       <h1>
52         {isDark ? "Welcome to the Dark side" : "This is the Light side"}
53       </h1>
54       <h2>Let's watch our styles update.</h2>
55       <button
56         css={buttonStyles}
57         onClick={() => {
58           setIsDark(!isDark);
59         }}
60       >
61         Change to {isDark ? "light" : "dark"} mode
62     </button>
63   </div>
64 </Fragment>
65 );
66 }
67
68 export default App;
```

```
const containerStyles = {
  padding: "50px 0",
  // backgroundColor: theme.background,
  // color: theme.text,
  textAlign: "center",
  height: "100vh",
  transitionDuration: "0.2s",
  transitionProperty: "background-color, color"
};
```

- CSS "Container styles"
as a JS Object!

↳ can use regular CSS
syntax

$css = \{button\ styles\} \Rightarrow$ another example of a CSS variable

```
const buttonStyles = {
  marginTop: "30px",
  background: '#fff',
  // border: `2px solid ${theme.buttonBorder}`,
  // backgroundColor: theme.buttonBg,
  // color: theme.buttonText,
  padding: "14px 28px",
  fontSize: "16px",
  transitionDuration: "0.2s",
  transitionProperty: "background-color, color",
  cursor: "pointer",

  "& :hover": {
    // backgroundColor: theme.buttonBgHover,
    // color: theme.buttonTextHover
  }
}
```

Code sandbox \Rightarrow a web-app based IDE

```
const themeLight = {
  text: "#181819",
  background: "#f0ece6",
  buttonText: "#181819",
  buttonTextHover: "#f0ece6",
  buttonBorder: "#181819",
  buttonBg: "#f0ece6",
  buttonBgHover: "#181819"
};
```

```
const themeDark = {
  text: "#f0ece6",
  background: "#181819",
  buttonText: "#f0ece6",
  buttonTextHover: "#181819",
  buttonBorder: "#f0ece6",
  buttonBg: "#181819",
  buttonBgHover: "#f0ece6"
};
```

} set different
themes and
switch as
easily as a
variable declaration

```
const buttonStyles = {
  marginTop: "30px",
  background: "#fff",
  border: `2px solid ${theme.buttonBorder}`,
  backgroundColor: theme.buttonBg,
  color: theme.buttonText,
  padding: "14px 28px",
  fontSize: "16px",
  transitionDuration: "0.2s",
  transitionProperty: "background-color, color",
  cursor: "pointer",

  "& :hover": {
    backgroundColor: theme.buttonBgHover,
    color: theme.buttonTextHover
  }
};
```

} example of setting
a style for
buttons