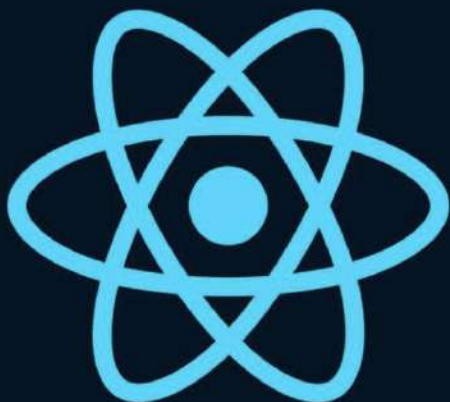# Styled components

What are styled components

Styled components are a CSS-in-JS tool that enable writing actual CSS code to style your components. With it - whenever you define your styles, you actually create a regular React component, with your styles attached to it.
It is compatible with both - React js and React Native

styled
components

Getting started

Installation:
npm install --save styled-components

No more need to create separate css files or using inline styling. For example this:

Could be achieved writing the following code:

```
const App = () => {
  return (
    <>
      <h1 style={{fontSize: '2em', textAlign: 'center', color: '#00cc00'}}>app title</h1>
    </>
  );
}

export default App;
```

We can now take compare this solution with styled components - let's create a **reusable** Title component

```jsx
import styled from "styled-components";

const Title = styled.h1`
  font-size: 2em;
  text-align: center;
  color: #00cc00;
`;

const App = () => {
  return (
    <>
      <Title>app title</Title>
    </>
  );
}
```

Result

app title

```html
▼<div id="root">
    <h1 class="sc-gsDKAQ bMqNRs">app title</h1>
  </div>
```

Another example - this time using buttons

```javascript
import styled from "styled-components";

const Button = styled.button`
  border: none;
  display: block;
  font-size: inherit;
  margin: 10px;
  color: #fff;
  background: #00cc00;
  cursor: pointer;
  padding: 25px 80px;
  text-transform: uppercase;
  letter-spacing: 1px;
`

const App = () => {
  return (
    <>
      <Button>button 1</Button>
      <Button>button 2</Button>
      <Button>button 3</Button>
    </>
  );
}
```

BUTTON 1

BUTTON 2

BUTTON 3

Final example including props

```jsx
import { useState } from "react";
import styled from "styled-components";

const Button = styled.button`
  margin: 10px;
  background: ${props=> props.isClicked ? '#00cc00': '#FF0000'};
  cursor: pointer;
  padding: 25px 80px;
`

const App = () => {
  const [btnClicked, setBtnClicked] = useState(false)

  const handleBtnClick = () => setBtnClicked(true)

  return (
    <>
      <Button onClick={handleBtnClick} isClicked={btnClicked}>
        {btnClicked ? 'You clicked me' : 'click me'}
      </Button>
    </>
  );
}
export default App;
```

click me ➡ You clicked me