



# Styled Components

O Styled Components é uma biblioteca (lib / library) do React e do React Native que nos permite criar componentes de estilo ao escrever códigos CSS dentro de um arquivo JavaScript.

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



NEXT





## Conhecendo o Styled Components

O modo de estilizar um componente utilizando a biblioteca Styled Components também é conhecido como CSS-in-JS (CSS dentro do JavaScript).

Lembrando que o styled-components é uma biblioteca, então devemos importar para o nosso arquivo js, e fazemos isso dessa forma:

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

NEXT



## Instalando o Styled Components

Já que o Styled Components é de uma biblioteca, para utilizá-la em nosso projeto React devemos realizar a sua instalação, que pode ser feita de duas maneiras: com o uso do Yarn ou do Npm.

Com o Npm temos essas duas formas:  
`npm install styled-components --save`  
`npm install --save styled-components`

Com Yarn:  
`yarn add styled-components`

```
MINGW64:/c/Users/colby
colby@DESKTOP-GGJ0P98 MINGW64 ~
test

npx create-react-app my-app
cd my-app
npm install styled-components --save
code .
npm start
```



Abri o terminal (linux) ou o gitbash (windows)  
Estou criando um projeto react e fazendo a  
instalação do styled components  
logo em seguida abri o visual studio code (vs code)

NEXT



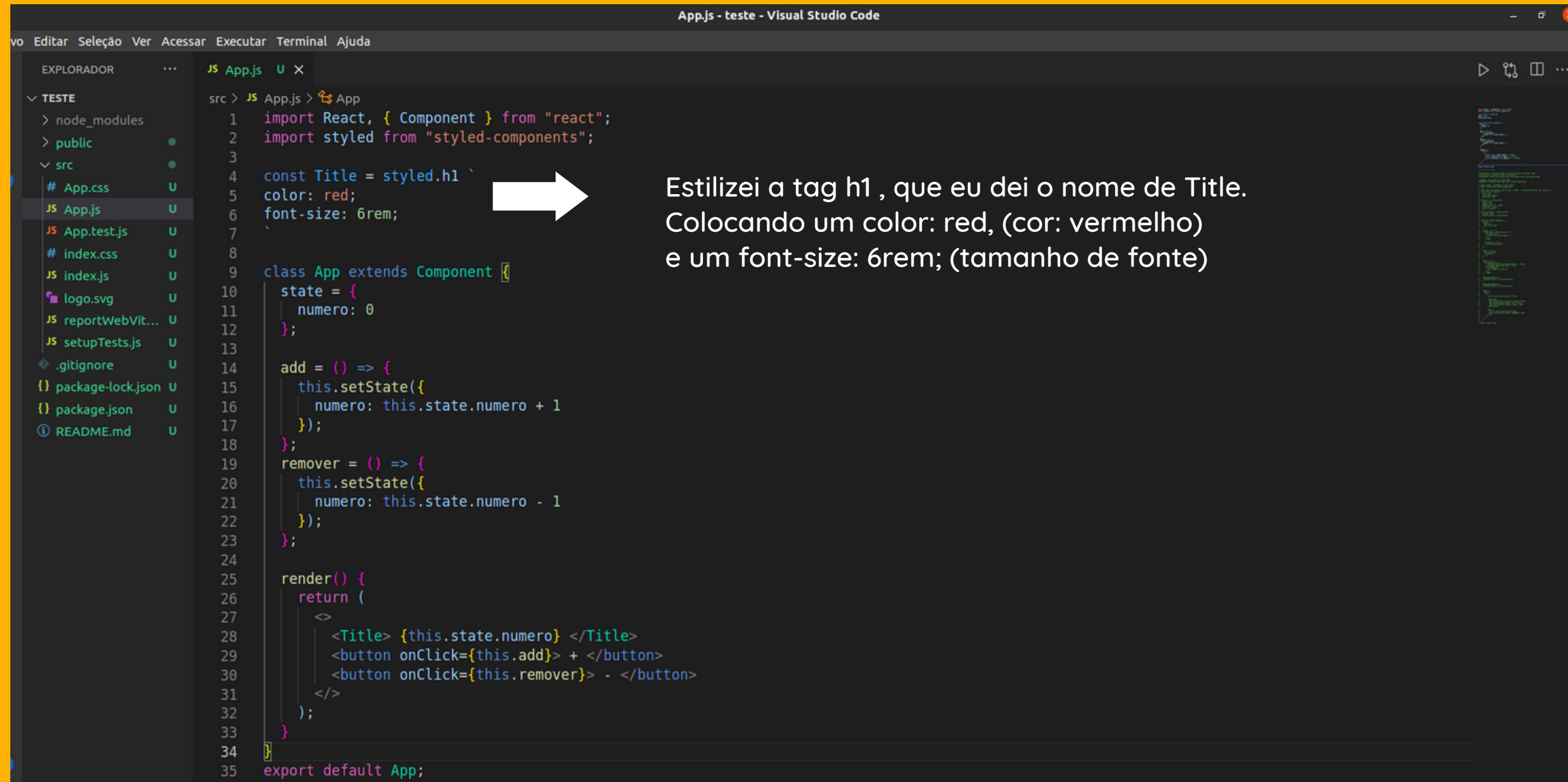
```
1 import React, { Component } from "react";
2 import styled from "styled-components"; ← importando a biblioteca styled-components
3
4 class App extends Component {
5   state = {
6     numero: 0
7   };
8
9   add = () => {
10     this.setState({
11       numero: this.state.numero + 1
12     });
13   };
14   remover = () => {
15     this.setState({
16       numero: this.state.numero - 1
17     });
18   };
19
20   render() {
21     return (
22       <>
23         <h1> {this.state.numero} </h1>
24         <button onClick={this.add}> + </button>
25         <button onClick={this.remover}> - </button>
26       </>
27     );
28   }
29 }
30 export default App;
```



Com a biblioteca importada no nosso arquivo, vamos começar a estilizar o nosso projeto do contador. Observe que tem uma tag `<h1>` na linha 23 e duas tags `<button>` na linha 24 e 25

NEXT





```
src > JS App.js > App
1  import React, { Component } from "react";
2  import styled from "styled-components";
3
4  const Title = styled.h1 `
5    color: red;
6    font-size: 6rem;
7  `
8
9  class App extends Component {
10    state = {
11      numero: 0
12    };
13
14    add = () => {
15      this.setState({
16        numero: this.state.numero + 1
17      });
18    };
19    remover = () => {
20      this.setState({
21        numero: this.state.numero - 1
22      });
23    };
24
25    render() {
26      return (
27        <>
28          <Title> {this.state.numero} </Title>
29          <button onClick={this.add}> + </button>
30          <button onClick={this.remover}> - </button>
31        </>
32      );
33    }
34  }
35  export default App;
```

Estilizei a tag h1 , que eu dei o nome de Title.  
Colocando um color: red, (cor: vermelho)  
e um font-size: 6rem; (tamanho de fonte)



Diferente do arquivo css puro, com o styled components chamamos a tag dessa forma:

```
const Title = styled.h1 ` `
```

- const para guardar a informação
- Title é o nome que eu dei pra minha tag <h1>
- styled: que ela vai receber = um estilo
- e que vai estilizar uma tag h1
- ` dentro das crases vai meu css nomal, color, background, font, position ... `

NEXT



App.js - teste - Visual Studio Code

Arquivo Editar Seleção Ver Acessar Executar Terminal Ajuda

EXPLORADOR

TESTE

node\_modules

public

src

App.css

App.js

App.test.js

index.css

index.js

logo.svg

reportWebVit...

setupTests.js

.gitignore

package-lock.json

package.json

README.md

src > JS App.js > [x] ButtonOne

1 import React, { Component } from "react";

2 import styled from "styled-components";

3

4 const Title = styled.h1`

5 color: red;

6 font-size: 6rem;

7 `

8 const ButtonOne = styled.button`

9 width: 10vw;

10 height: 7vh;

11 font-size: 3rem;

12 background-color: yellow;

13 `

14

15 const ButtonTwo = styled.button`

16 width: 10vw;

17 height: 7vh;

18 font-size: 3rem;

19 background-color: pink;

20 `

21

22

23 class App extends Component {

24 state = {

25 numero: 0

26 };

27

28 add = () => {

29 this.setState({

30 numero: this.state.numero + 1

31 });

32 };

33 remover = () => {

34 this.setState({

35 numero: this.state.numero - 1

36 });

37 };

38

39 render() {

40 return (

41 <>

42 <Title> {this.state.numero} </Title>

43 <ButtonOne onClick={this.add}> + </ButtonOne>

44 <ButtonTwo onClick={this.remover}> - </ButtonTwo>

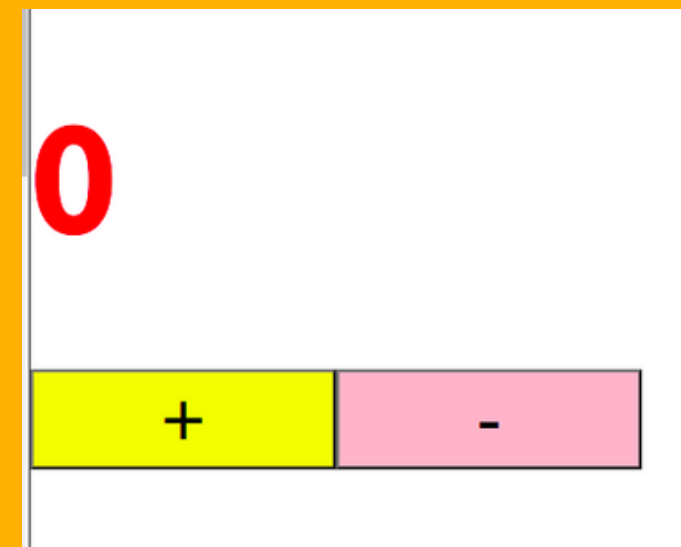
45 </>

46 );

ESTRUTURA DO CÓDIGO

LINHA DO TEMPO

Agora estilizei as duas tags <button>  
Dei um nome pra cada uma delas:  
<ButtonOne> e <ButtonTwo>  
E fiz o mesmo processo da tag <h1> ,  
chamei a const o nome do button = styled.button ``



NEXT





Arquivo

Editar

Seleção

Ver

Acessar

Executar

Terminal

Ajuda

EXPLORADOR

...

TESTE

node\_modules

public

src

App.css

App.js

App.test.js

index.css

index.js

logo.svg

reportWebVit...

setupTests.js

.gitignore

package-lock.json

package.json

README.md

JS App.js

U X

src > JS App.js > [0] Container

17 height: 7vh;

18 font-size: 3rem;

19 background-color: pink;

20 `

21

22 const Container = styled.section `

23 width: 100%;

24 height: 100vh;

25 display: flex;

26 flex-direction: column;

27 justify-content: space-evenly;

28 align-items: center;

29 `

30

31 const BoxBtn = styled.div `

32 width: 100%;

33 height: 50vh;

34 display: flex;

35 justify-content: space-evenly;

36 align-items: center;

37 `

38

39

40 class App extends Component {

41 > state = { ...

43 };

44

45 > add = () => { ...

49 };

50 > remover = () => { ...

54 };

55

56 render() {

57 return (

58 <>

59 <Container>

60 <Title> {this.state.numero} </Title>

61 <BoxBtn>

62 <ButtonOne onClick={this.add}> + </ButtonOne>

63 <ButtonTwo onClick={this.remover}> - </ButtonTwo>

64 </BoxBtn>

65 </Container>

66 </>

67 );

68 }

O estilo segue o mesmo processo, porém estou usando flex-box (caixa flexível) para facilitar a estilização.

0

+

-

Agora vamos brincar um pouco com as caixas flexíveis (Flex box) . Criei mais duas tags, uma chamada <Container> E uma outra tag chamada <BoxBtn> , que está envolvendo somente os botões, ou seja, ela é a mãe dos botões. E a tag <Container>, é a mãe do <Title> e da tag <BoxBtn>

NEXT





styled  
components

