

# IW-II

# INTERFACE WEB II

---

Prof. Anderson Vanin

**AULA 14 – CRIAÇÃO DE COMPONENTES REACT, PROPS E CSS**

# Componentes

- Permite **dividir a aplicação** em partes;
- Os componentes **renderizam JSX**, assim como o App.js (que é um componente);
- Precisamos **criar um arquivo** de componente;
- **Importá-lo** onde precisa ser utilizado;
- Normalmente ficam em uma pasta chamada **components**;

# Criando o primeiro componente



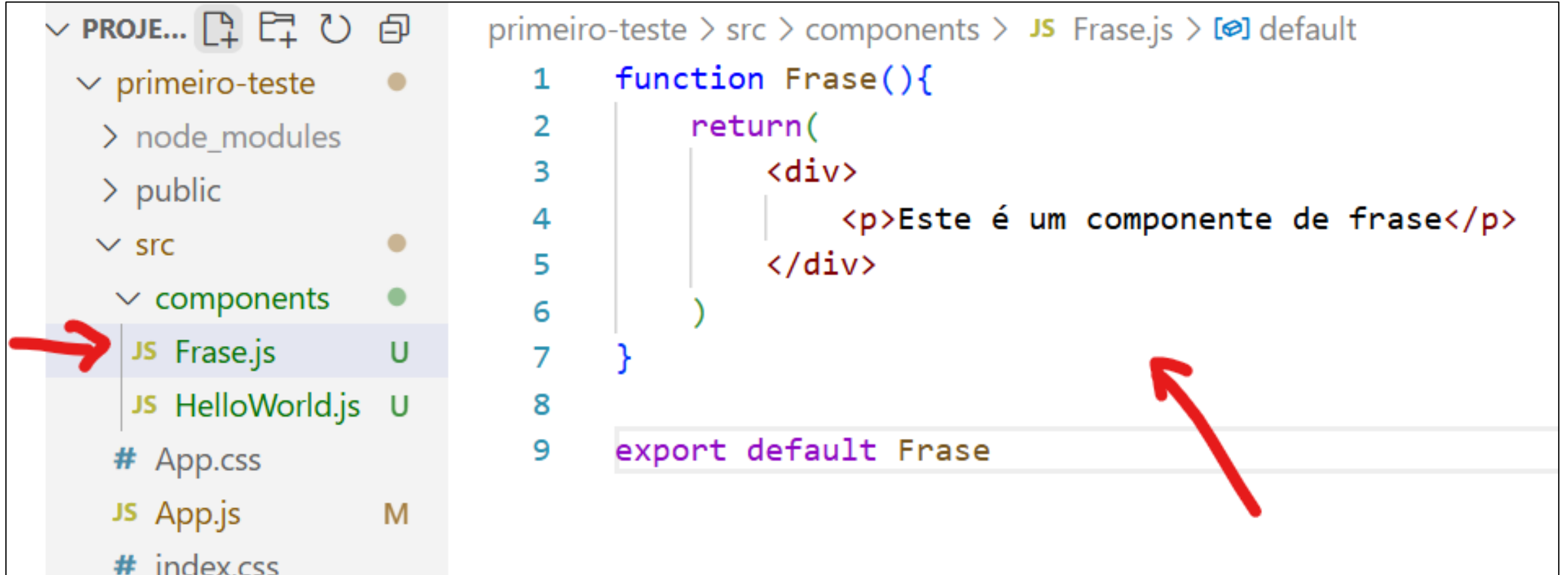
```
1  function HelloWorld(){
2      return(
3          <div>
4              <h1>MEU PRIMEIRO COMPONENTE</h1>
5          </div>
6      )
7  }
8
9  export default HelloWorld
```

# Importando o componente criado no App.js

```
1  import './App.css';
2  import HelloWorld from './components/HelloWorld';
3
4  function App() {
5    return (
6      <div className="App">
7        <h1>MEU APP REACT</h1>
8        <HelloWorld />
9      </div>
10   );
11 }
12
13 export default App;
```



# Criando outro componente



primeiro-teste > src > components > JS Frase.js > default

```
1 function Frase(){
2   return(
3     <div>
4       <p>Este é um componente de frase</p>
5     </div>
6   )
7 }
8
9 export default Frase
```

# Importando outro componente criado no App.js

```
primeiro-teste > src > JS App.js > ...
1  import './App.css';
2  import HelloWorld from './components/HelloWorld';
3  import Frase from './components/Frase'; ←
4
5  function App() {
6    return (
7      <div className="App">
8        <h1>MEU APP REACT</h1>
9        <HelloWorld />
10       <Frase /> ←
11     </div>
12   );
13 }
14
15 export default App;
16
```



# Reaproveitamento de Componentes

```
primeiro-teste > src > JS App.js > App
1  import './App.css';
2  import HelloWorld from './components/HelloWorld';
3  import Frase from './components/Frase';
4
5  function App() {
6    return (
7      <div className="App">
8        <h1>MEU APP REACT</h1>
9        <HelloWorld />
10       <Frase />
11       <Frase />
12       <Frase />
13     </div>
14   );
15 }
16
17 export default App;
```



# Props

- As **props** são valores passados para componentes;
- Podemos deixa-los **dinâmicos**;
- Ou seja, **mudando a execução** por causa do valor da prop;
- O valor é **passado como um atributo** na chamada do componente;
- E precisa ser **resgatado dentro de uma propriedade/argumento** chamada props na função de definição do componente;
- **As props são somente de leitura!**



# Criando um novo componente para utilizar Props



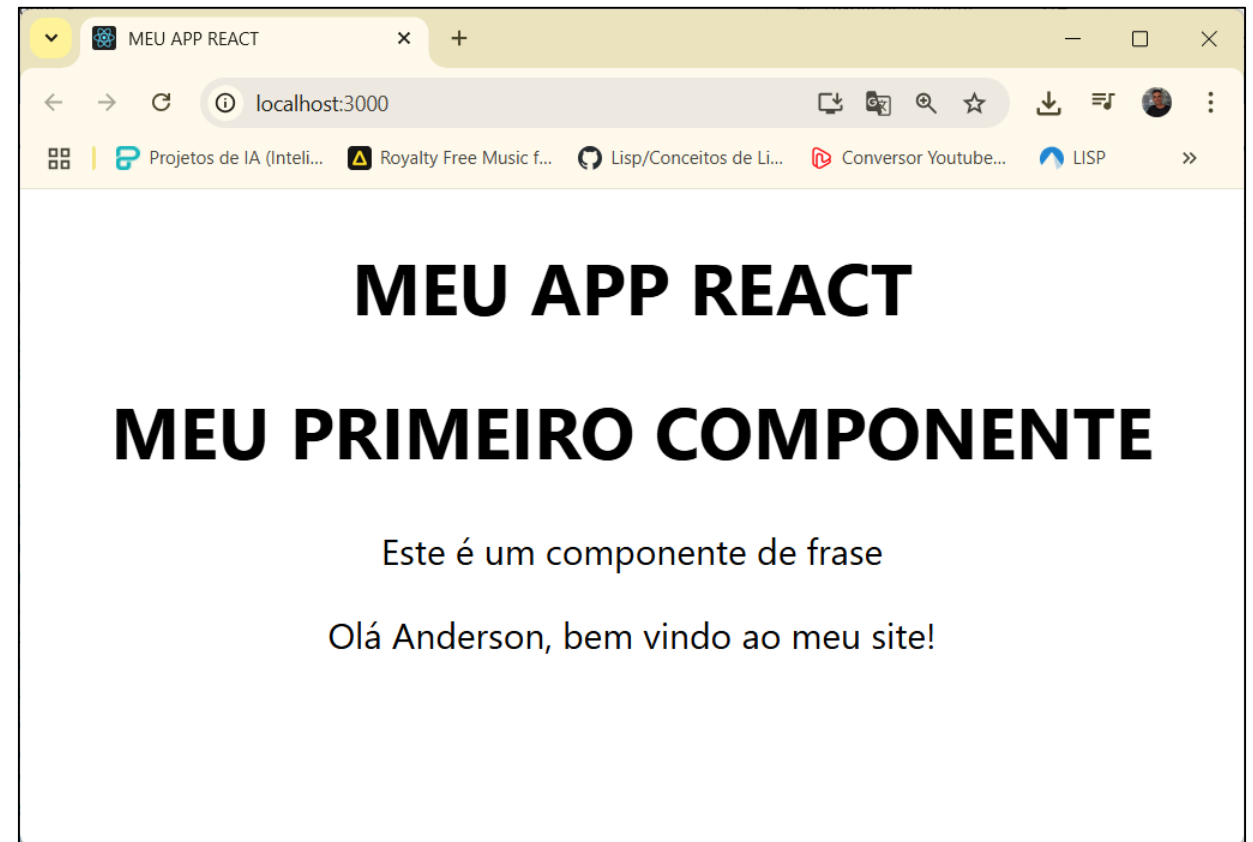
The image shows a code editor interface with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a folder named 'primeiro-teste' containing subfolders 'node\_modules', 'public', and 'src'. The 'src' folder contains a subfolder 'components' which is expanded, showing a list of files: 'JS DigaMeuNome.js' (highlighted with a red arrow), 'JS Frase.js', 'JS HelloWorld.js', '# App.css', 'JS App.js', '# index.css', and 'JS index.js'. The code editor shows the content of 'DigaMeuNome.js' with the following code:

```
primeiro-teste > src > components > JS DigaMeuNome.js > DigaMeuNome
1  function DigaMeuNome(props){
2      return(
3          <div>
4              <p>Olá {props.nome}, bem vindo ao meu site!</p>
5          </div>
6      )
7  }
8
9  export default DigaMeuNome
```

A red arrow points to the closing curly brace of the function on line 6, and another red arrow points to the 'export default' statement on line 9.

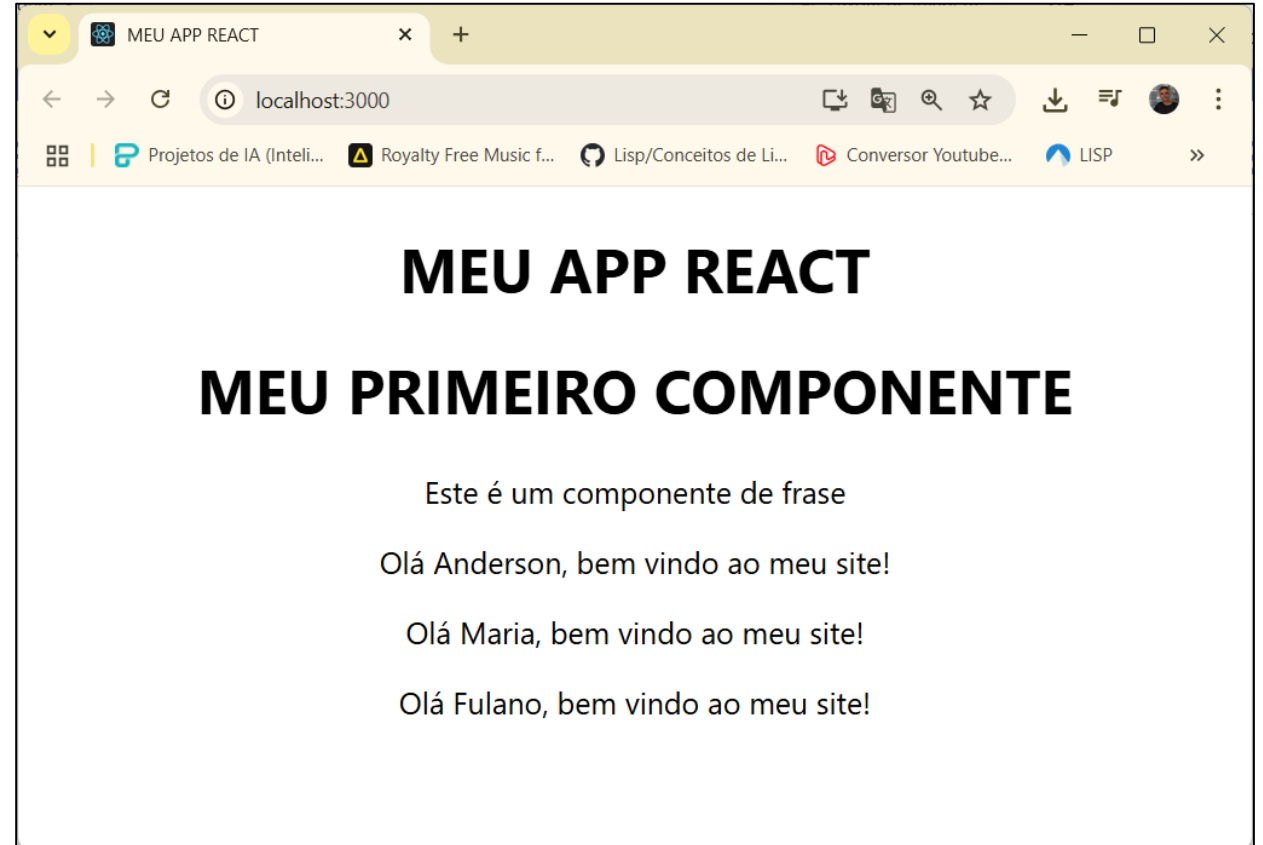
# Importando o componente criado e passando um valor

```
primeiro-teste > src > JS App.js > App
1  import './App.css';
2  import HelloWorld from './components/HelloWorld';
3  import Frase from './components/Frase';
4  import DigaMeuNome from './components/DigaMeuNome';
5
6  function App() {
7    return (
8      <div className="App">
9        <h1>MEU APP REACT</h1>
10       <HelloWorld />
11       <Frase />
12       <DigaMeuNome nome="Anderson" />
13     </div>
14   );
15 }
16
17
18 export default App;
19
```



# Reaproveitamento alterando valor da propriedade

```
primeiro-teste > src > JS App.js > App
1  import './App.css';
2  import HelloWorld from './components/HelloWorld';
3  import Frase from './components/Frase';
4  import DigaMeuNome from './components/DigaMeuNome';
5
6  function App() {
7    return (
8      <div className="App">
9        <h1>MEU APP REACT</h1>
10       <HelloWorld />
11       <Frase />
12       <DigaMeuNome nome="Anderson" />
13       <DigaMeuNome nome="Maria" />
14       <DigaMeuNome nome="Fulano" />
15     </div>
16   );
17 }
18
19
20 export default App;
```

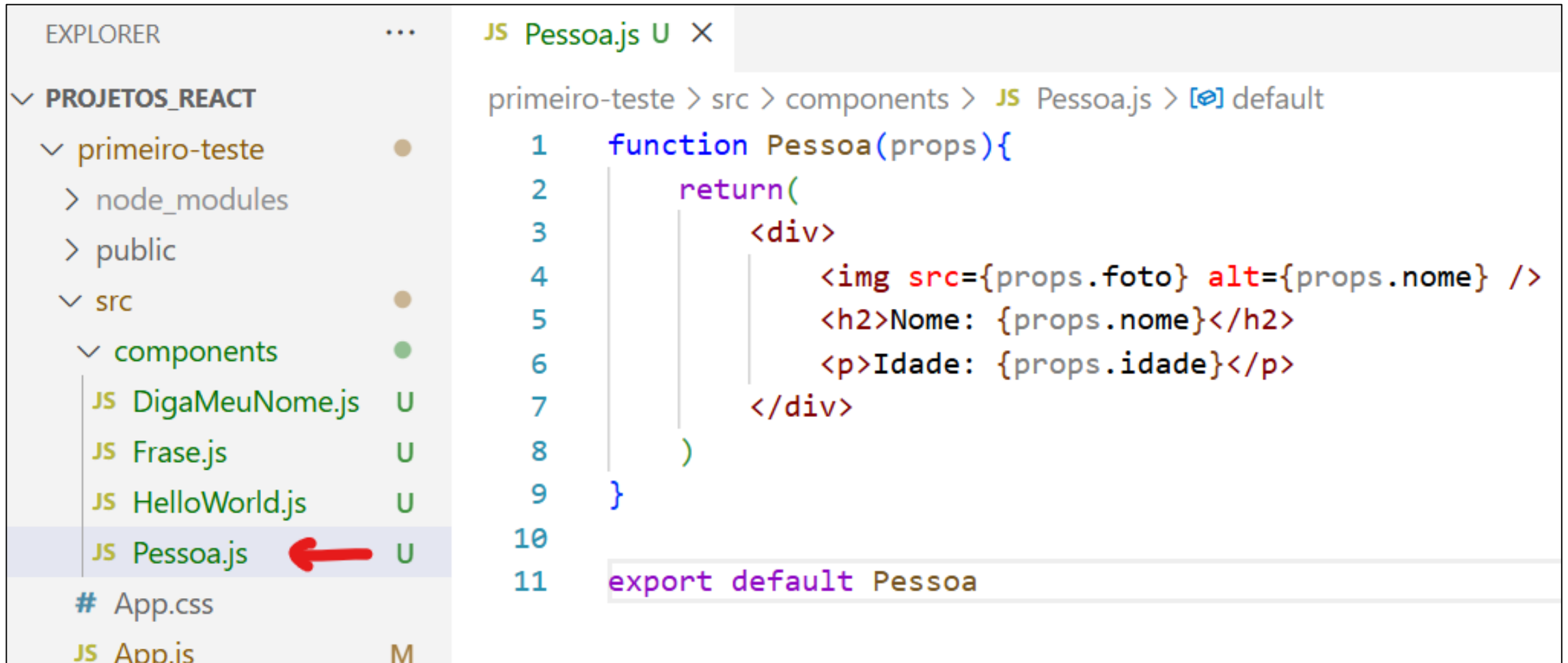


# Recuperando valores de uma variável para uma props

```
primeiro-teste > src > JS App.js > ...
1  import './App.css';
2  import HelloWorld from './components/HelloWorld';
3  import Frase from './components/Frase';
4  import DigaMeuNome from './components/DigaMeuNome';
5
6  function App() {
7    const nome = "Beltrano da Silva" ←
8    return (
9      <div className="App">
10        <h1>MEU APP REACT</h1>
11        <HelloWorld />
12        <Frase />
13        <DigaMeuNome nome="Anderson" />
14        <DigaMeuNome nome="Maria" />
15        <DigaMeuNome nome={nome} /> ←
16      </div>
17    );
18  }
19
20
21  export default App;
```



# Enviando mais valores



EXPLORER

PROJETOS\_REACT

- primeiro-teste
  - node\_modules
  - public
  - src
    - components
      - JS Pessoa.js** U
      - JS DigaMeuNome.js U
      - JS Frase.js U
      - JS HelloWorld.js U
- # App.css
- JS App.js M

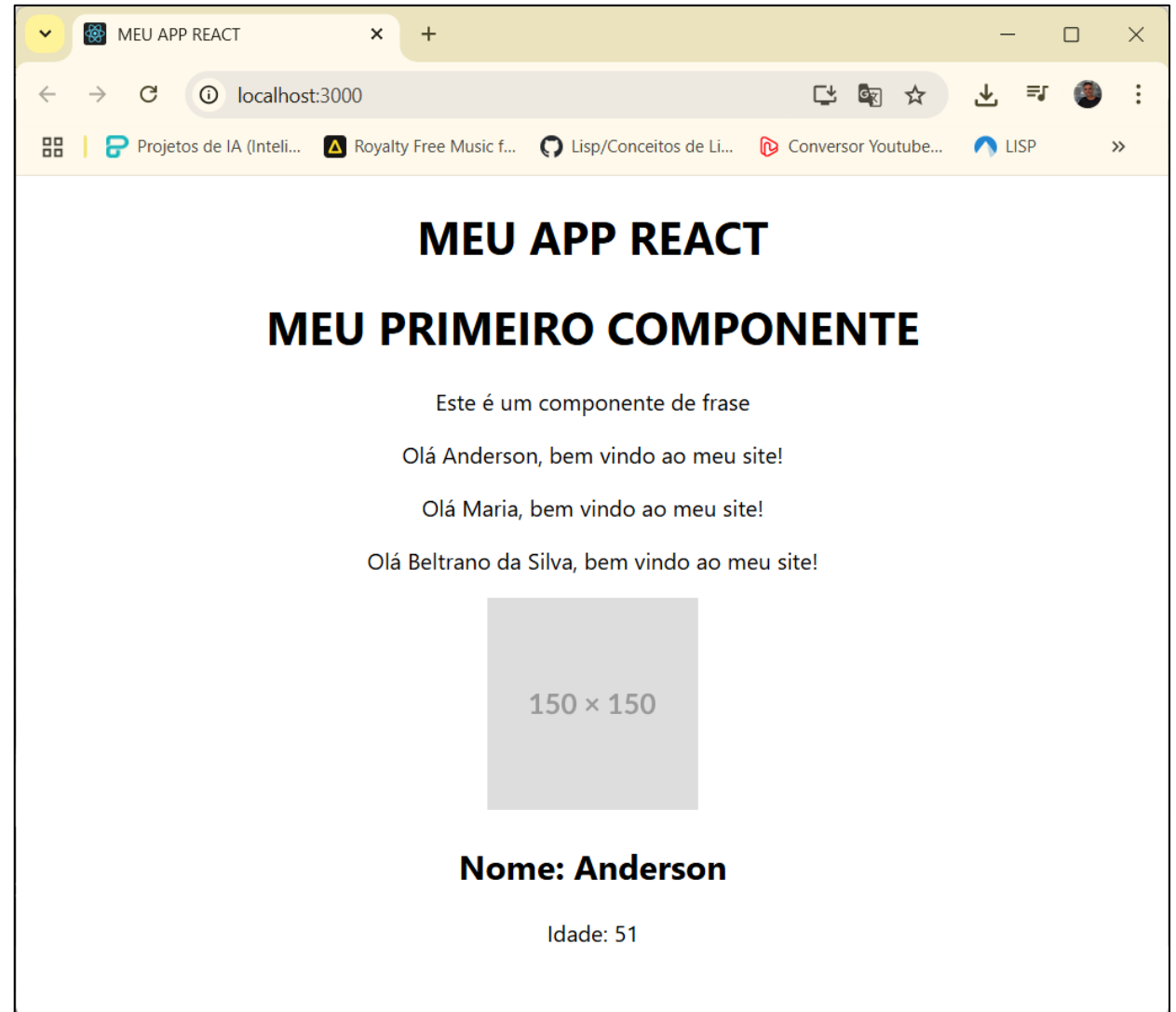
JS Pessoa.js U X

primeiro-teste > src > components > JS Pessoa.js > [🔗] default

```
1 function Pessoa(props){
2   return(
3     <div>
4       <img src={props.foto} alt={props.nome} />
5       <h2>Nome: {props.nome}</h2>
6       <p>Idade: {props.idade}</p>
7     </div>
8   )
9 }
10
11 export default Pessoa
```

# Importando o componente Pessoa.js

```
1 import './App.css';
2 import HelloWorld from './components/HelloWorld';
3 import Frase from './components/Frase';
4 import DigaMeuNome from './components/DigaMeuNome';
5 import Pessoa from './components/Pessoa';
6
7 function App() {
8   const nome = "Beltrano da Silva"
9   return (
10     <div className="App">
11       <h1>MEU APP REACT</h1>
12       <HelloWorld />
13       <Frase />
14       <DigaMeuNome nome="Anderson" />
15       <DigaMeuNome nome="Maria" />
16       <DigaMeuNome nome={nome} />
17       <Pessoa
18         nome="Anderson"
19         idade="51"
20         foto = "https://placeholder.co/150"
21       />
22     </div>
23   );
24 }
25
26
27 export default App;
28
```



# Outra maneira de trazer vários valores (sem props)

primeiro-teste > src > components > JS Pessoa.js > [e] default

```
1  function Pessoa({nome, idade, foto}){
2      return(
3          <div>
4              <img src={foto} alt={nome} />
5              <h2>Nome: {nome}</h2>
6              <p>Idade: {idade}</p>
7          </div>
8      )
9  }
10
11  export default Pessoa
```

# Adicionando CSS

- O **CSS** pode ser **adicionado de forma global** na aplicação, por meio do arquivo `index.css` por exemplo;
- Porém é possível **estilizar a nível de componentes**;
- Utilizamos **CSS modules** para isso;
- Basta criar um arquivo como por exemplo: **`Componente.module.css`**
- Chamar este CSS no componente.



# Adicionando CSS

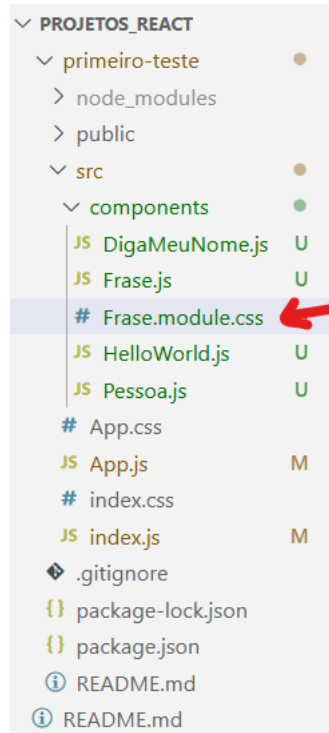
- O arquivo index.css afeta de maneira global todo o projeto. Portanto, toda alteração deste CSS irá afetar diretamente todos os componentes.



# Criando estilos individuais para componentes

- Vamos utilizar um componente já criado: **Frase.js**
- Para aplicar um estilo individualizado neste componente, precisamos criar um arquivo com `module.css`. Lembre-se de utilizar o mesmo nome do componente, neste caso ficará:

## Frase.module.css



```
primeiro-teste > src > components > # Frase.module.css > .fraseConteudo
1  /* OBS: NÃO É POSSÍVEL CRIAR NOMES DE CLASSES COM -
2  NESTE CASO PODE-SE USAR O _ */
3
4  .fraseContainer{
5      background-color: #333;
6      border: 1px solid #f00;
7  }
8
9  .fraseConteudo{
10     color: #fff;
11     background-color: #333;
12     margin: 0;
13 }
```

# Importando o estilo criado no componente Frase.js

```
primeiro-teste > src > components > JS Frase.js > default
1  import styles from './Frase.module.css'
2
3  function Frase(){
4      return(
5          <div className={styles.fraseContainer}>
6              <p className={styles.fraseConteudo}>Este é um componente de frase</p>
7          </div>
8      )
9  }
10
11  export default Frase
```

