

Comando JS par criar um projeto

1. Criar sua pasta do projeto
2. Entrar na pasta do seu projeto
3. Dar o comando para criar um projeto React

```
npx create-react-app seu-app
```

4. Foram criadas várias pastas de um projeto em React

5. Iniciar a aplicação

```
npm start
```

6. Um componente padrão foi criado (<http://localhost:3000>)
7. Somente agora, na pasta src você poderá criar sua pasta componente
8. Dependendo do seu S.O. dá para criar com \$mkdir componente
9. Entrar nessa pasta com o comando (já no VSCode)

```
cd componente
```

10. Criar o arquivo PrimentoComponente.js (no VSCode)

11. Um componente possui (ao menos)

a) Um comando import, tipo: `import React from 'react'`

b) Uma função que vai definir o componente ex:

```
function MeuComponente() {  
  ...  
}
```

c) Dentro dessa função deve ter uma tag: `<div> ... </div>`

```
function MeuComponente() {  
  <div> ... </div>  
}
```

d) Um comando export, tipo: `export default MeuComponente;`

e) Exemplo completo:

```
import React from 'react';  
  
function MeuComponente() {  
  return (  
    <div>  
      Meu primeiro componente  
    </div>  
  );  
}  
  
export default MeuComponente;
```

Existe outra forma também de definir as funções, são o tipo Arrow Function ou

funções SETA "() =>"

Veja o exemplo:

```
const MeuComponente = () => {  
  return(  
    <div>  
      <h1>Meu primeiro componente</h1>  
    </div>  
  );  
};  
  
export default MeuComponente;
```

12. Para utilizar o componente criado termos que importá-lo para dentro do arquivo App.js

exemplo de um App.js

```
//componentes  
import MeuComponente from './componentes/MeuComponente';  
  
// styles /css  
import './App.css';  
  
function App() {  
  return (  
    <div className="App">  
      <h1>Fundamentos React</h1>  
      <MeuComponente/>  
    </div>  
  );  
}  
  
export default App;
```

Template Expressions:

```
const TemplateExpression = () => {  
  
const nome = "Meu nome aqui";  
  return (  
    <div>  
      <h1>OLÁ {nome}, tudo bem?</h1>  
    </div>  
  );  
};  
export default TemplateExperssion;
```

Então, para chamar o componente:

```
//componentes
import MeuComponente from './componentes/MeuComponente';
import TemplateExpression from './componentes/TemplateExpression';

// styles /css
import './App.css';

function App() {
  return (
    <div className="App">
      <h1>Fundamentos React</h1>
      <MeuComponente/> // ← anterior componente
      <TemplateExpression/> // ← chamamos o novo componente aqui
    </div>
  );
}
export default App;

///-----
```

Template Expression versão 2:

```
const TemplateExpressions = () => {
  const nome = "Marquinhos";
  const dados = {
    idade: 22,
    cargo: "Desenvolvedor",
  };

  return (
    <div>
      <h1>Olá {nome}, tudo bem?</h1>
      <p>Você atua como: {dados.cargo}</p>
    </div>
  );
};

export default TemplateExpressions;
```