

REACT: Introdução e Renderização de Componentes



DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

AGENDA

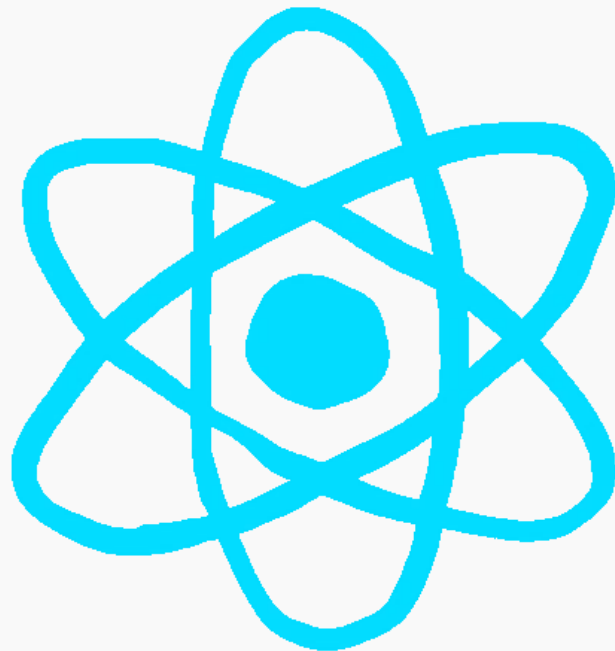
- Ambiente de desenvolvimento
- Introdução a React
- Criar um projeto React
- Renderização de Componentes
- JSX

INTRODUÇÃO

O que é?

React é uma **biblioteca JavaScript**
(um compilado de código JavaScript pré-escrito)
que **facilita** a construção de
interfaces visuais complexas.

Essa biblioteca foi **criada pelo Facebook**
e hoje é usada em muitas empresas, como
Netflix, Instagram, Twitter, etc.



Porque estudamos um jeito “+ difícil” primeiro?

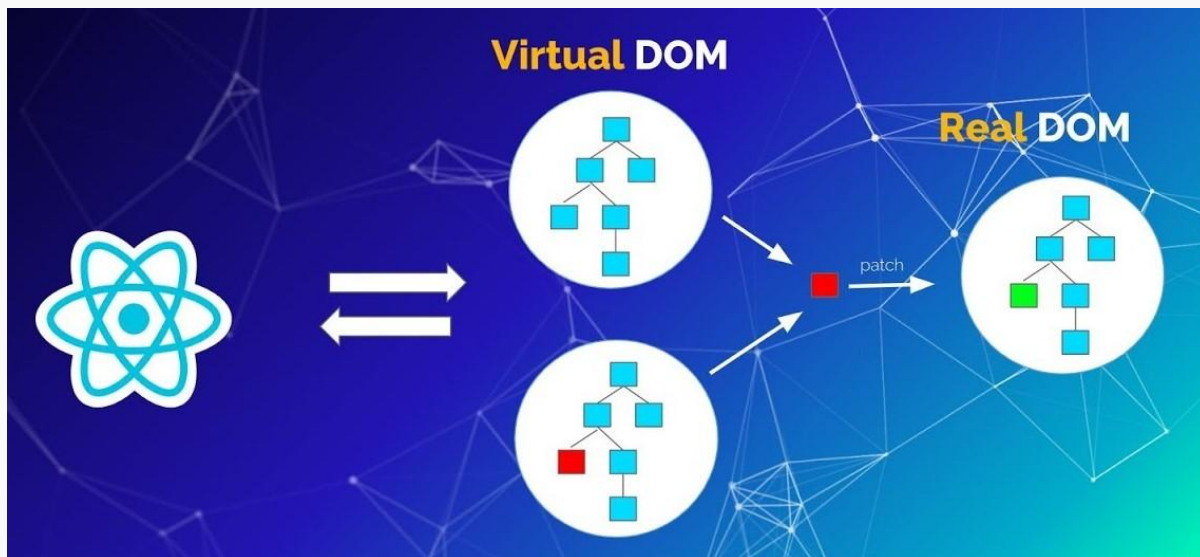
Para entender React (*de verdade!*) é muito importante **que já tenhamos aprendido** (*pelo menos*):



- Manipulação de DOM
Precisamos entender como o JS consegue **manipular elementos** que já estão ou que precisamos criar na interface através de **eventos do usuário** (clique de botão, envio de formulário,...).
O que o React faz, afinal, é manipular o HTML a partir do JS.
- Estrutura de dados
React utiliza muito de **array** e **objetos** para fazer armazenar e trabalhar com dados. Então é preciso saber usar métodos de array (como map, filter, reduce, find, includes,...) além de entender JSON, que será super importante na comunicação entre frontend e backend.
- Assincronismo
É importante conhecer o funcionamento de **ações que demoram** um tempo para acontecer (desde setTimeout a chamadas para uma API externa), assim como **Promises** (e a sintaxe async/await) e o uso da **API fetch** do navegador (uma das grandes responsabilidades na construção de aplicações com React é integrar/fazer requisições com backends já existentes).

O que o React faz?

Ao invés de manipular o DOM do navegador,
o React cria um **DOM VIRTUAL** na memória,
onde faz toda a manipulação necessária,
e altera no DOM do navegador apenas o que for necessário.





AMBIENTE DE DESENVOLVIMENTO

Ambiente de desenvolvimento

Node.js

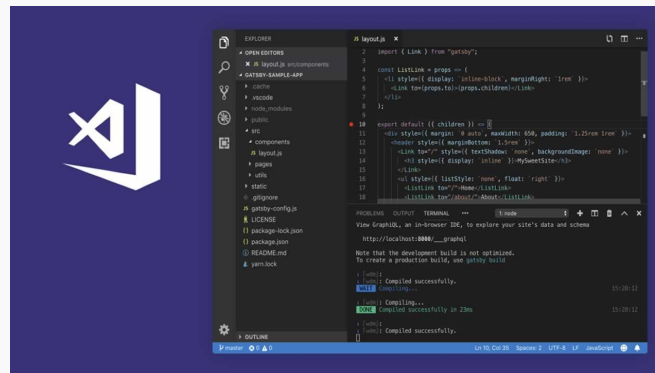
De forma resumida, Node.js é uma **ferramenta de execução de JavaScript** fora do contexto do navegador, **no desktop**.

<https://nodejs.org/en/>



OBS: há **alternativas** para quem não pode instalar Node na sua máquina, como:

<https://playcode.io/>
<https://codesandbox.io/>



Visual Studio Code

VS Code é um **editor de código** muito usado por desenvolvedores e empresas de software ao redor do mundo.

<https://code.visualstudio.com/>

Nós aprendemos a instalar o Node.js
e ele será muito útil na criação dos nossos projetos React.

Comandos úteis:

```
# Mostrar a versão do Node instalada na máquina
node --version

# Mostrar a versão do npm (gerenciador de pacotes do Node) instalada na máquina
npm -v

# Instalar pacotes JavaScript
npm install <nome_do_pacote>

# Executar um script de um pacote instalado ou disponível no repositório npm
npx <nome_do_executavel>
```

INTERVALO DE AULA

DEV!

Finalizamos o nosso primeiro período de hoje. Que tal descansar um pouco?!

Nos vemos em 20 minutos.

Início: 20:20

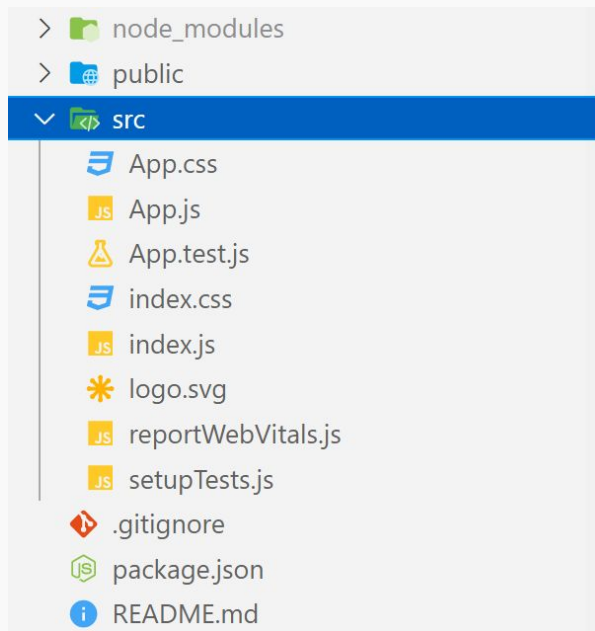
Retorno: 20:40



RENDERIZAÇÃO DE COMPONENTES

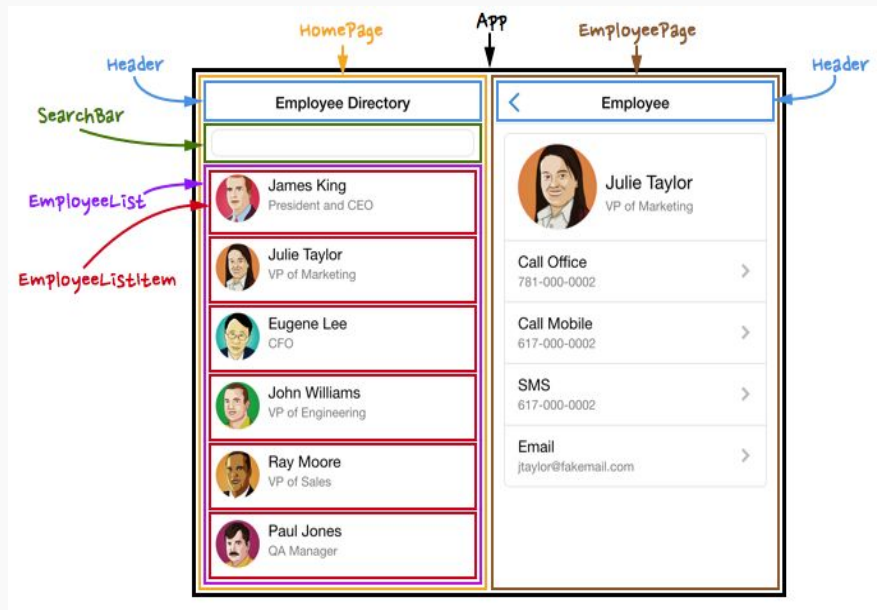
Criar projeto React

Essa é a estrutura inicial de um projeto React:



Renderização de componentes

O React permite criar UIs (*user interfaces*) utilizando **componentes isolados**.
Facilita a renderização e atualização de dados.



Renderização de componentes

Um componente React é, em essência, uma função JavaScript que **retorna** um conteúdo que será *renderizado na tela*.

Exemplo:

ES6 Arrow Function

```
const Botao = () => {  
  return <button>Clique no Botão</button>  
}  
  
export default Botao;
```

*Function declaration
(declaração de função)*

```
function Botao() {  
  return <button>Clique no Botão</button>  
}  
  
export default Botao;
```

JSX

Apesar de aparentemente usarmos um elemento HTML em nosso JS, na prática o que estamos utilizando é um **elemento JSX**.

JSX é uma **extensão** de linguagem JavaScript que possibilita a escrita de código com uma sintaxe mais amigável ao programador (com a mesma experiência de escrever HTML)

```
function Botao() {  
  return <button>Clique no Botão</button>  
}  
  
export default Botao;
```


CRIANDO UM PROJETO REACT

Criar projeto React

Para **criar** um projeto React, vamos executar um **script** de automação chamado ***create-react-app***:

```
# Executar o script de criação do React  
npx create-react-app <nome_do_projeto>
```

E para **inicializar** o projeto:

```
# Iniciar/rodar o projeto React  
npm start
```

Não precisamos instalar o pacote create-react-app localmente. Quando executamos o comando de criação do projeto, baixamos e executamos os *scripts* de criação e configuração sem instalar os arquivos na nossa máquina.

Vamos criar nosso **primeiro projeto React**
e adicionar **componentes** nele.



ESTILIZAÇÃO

Estilização Inline

Há ***muitas maneiras*** de aplicar estilização (usar CSS) no React, como a já conhecida folha de estilo CSS (arquivo com a extensão **.css**), que já estudamos, a estilização inline, entre outras.

Na **estilização inline** usamos o atributo **style** com um objeto JS.

```
export function Botao() {  
  return (  
    <button style={{ backgroundColor: "blue", color: "white" }}>  
      Sou um botão  
    </button>  
  );  
}
```

```
export function Botao() {  
  const meuEstilo = {  
    backgroundColor: "blue",  
    color: "white",  
  }  
  return (  
    <button style={meuEstilo}>  
      Sou um botão  
    </button>  
  );  
}
```

OBS: como a estilização é escrita dentro do JS é necessário usar camelCase

Crie um **projeto React**
que recebe dois **componentes**:

- um título: Meu primeiro projeto React
 - utilize ***folha de estilo CSS*** para deixá-lo centralizado e colorido
- um texto: Criar um projeto React é muito fácil <3
 - utilize ***estilização inline*** para deixá-lo em itálico e de outras maneiras que sua imaginação permitir

- [Componentes e Props – React](#)
- [Do Zero ao React | Como fazer o React e aprender a ideia do Virtual DOM na prática](#)
- [Renderização condicional – React](#)
- [Styling React Using CSS](#)
- [Instalando e gerenciando várias versões do Node.js com NVM | Blog TreinaWeb](#)

AVALIAÇÃO DOCENTE

O que você está achando das minhas aulas neste conteúdo?

[Clique aqui](#) ou escaneie o QRCode ao lado para avaliar minha aula.

Sinta-se à vontade para fornecer uma avaliação sempre que achar necessário.





DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>