

Aula 12 - Revisão e Introdução ao React com Vite

Objetivo da Aula

Nesta aula, faremos uma **revisão** dos conceitos básicos já estudados, como **componentes**, **páginas**, **CSS em módulos e React Router**, antes de configurar um novo projeto utilizando **Vite**. O objetivo é reforçar a base e preparar para conteúdos mais avançados.

Revisão dos Conceitos Anteriores

Componentes no React

- **Componentes funcionais:** Como criar e estruturar.
- **Reutilização e organização** de componentes.

Pages e React Router

- **React Router:** Criando navegação entre páginas.
- Uso de `<BrowserRouter>`, `<Routes>` e `<Route>`.
- **Exemplo de estrutura de rotas:**

```
import { BrowserRouter, Routes, Route } from 'react-router-dom';
import Home from './pages/Home';
import About from './pages/About';

function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
      </Routes>
    </BrowserRouter>
  );
}

export default App;
```

CSS em Componentes (CSS Modules)

- O que são CSS Modules?
- Como criar um arquivo de estilo modularizado (`.module.css`).
- Importando estilos dentro do componente.
- Exemplo de uso:

```
import styles from './MeuComponente.module.css';

function MeuComponente() {
  return <h1 className={styles.titulo}>Olá, mundo!</h1>;
}

export default MeuComponente;
```

Exemplo de **MeuComponente.module.css**:

```
.titulo {
  color: blue;
  font-size: 2rem;
}
```

O Que é React?

React é uma biblioteca JavaScript para a criação de interfaces de usuário interativas. Criada pelo Facebook, ela permite desenvolver aplicações baseadas em **componentes reutilizáveis**.

Principais vantagens do React:

- Criação de **componentes reutilizáveis**.
- Atualização eficiente da interface com o **Virtual DOM**.
- Grande comunidade e suporte.
- Facilita a construção de **Single Page Applications (SPAs)**.

O Que é Vite?

Vite é uma ferramenta moderna de build que permite configurar projetos React de maneira rápida e otimizada. Ele substitui o tradicional **Create React App (CRA)**, oferecendo tempos de inicialização mais rápidos e recarregamento instantâneo durante o desenvolvimento.

Por que usar o Vite?

- **Inicialização rápida.**
- **Hot Module Replacement (HMR)**, permitindo recarregamento instantâneo.
- **Build mais eficiente e leve.**
- Suporte nativo ao **ESModules**.

Criando um Projeto React com Vite

Vamos criar nosso primeiro projeto React utilizando Vite. Para isso, siga os passos abaixo:



1. Instalar o Vite e Criar o Projeto

No terminal, execute:

```
npm create vite@latest meu-projeto-react -- --template react
```

Substitua **meu-projeto-react** pelo nome do seu projeto.

✂ 2. Acessar o Diretório do Projeto

```
cd meu-projeto-react
```



3. Instalar as Dependências

```
npm install
```



4. Rodar o Servidor de Desenvolvimento

```
npm run dev
```

Isso iniciará o projeto e exibirá um link no terminal (geralmente <http://localhost:5173/>) para acessar no navegador.



Entendendo a Estrutura do Projeto

Após criar o projeto, temos a seguinte estrutura de arquivos:

```
meu-projeto-react/  
├── node_modules/  
├── public/  
├── src/  
│   ├── App.jsx  
│   ├── main.jsx  
│   ├── assets/  
│   ├── components/  
│   └── pages/  
├── index.html  
├── package.json  
├── vite.config.js  
└── README.md
```

Principais Arquivos:

- `index.html` → Arquivo principal da aplicação.
- `src/main.jsx` → Ponto de entrada do React.
- `src/App.jsx` → Componente principal da aplicação.
- `src/components/` → Diretório para armazenar componentes.
- `vite.config.js` → Arquivo de configuração do Vite.



Criando Nosso Primeiro Componente React

Vamos modificar o arquivo `App.jsx` para exibir uma mensagem na tela:

1. Abrir o Arquivo `src/App.jsx` e Editar o Código:

```
function App() {  
  return (  
    <div>  
      <h1> Olá, mundo! ✨ </h1>  
      <p>Este é meu primeiro componente React com Vite!</p>  
    </div>  
  );  
}  
  
export default App;
```

2. Salvar e Atualizar o Navegador

Com o servidor rodando (`npm run dev`), basta atualizar o navegador para ver a alteração.



Conclusão

Nesta aula, revisamos:

- ☒ **Componentes;**
- ☒ **React Router e navegação entre páginas;**
- ☒ **CSS em Componentes utilizando CSS Modules;**
- ☒ O que é **React** e suas vantagens;
- ☒ O que é **Vite** e por que usá-lo;
- ☒ Como **criar um projeto React com Vite;**
- ☒ Estrutura de arquivos de um projeto React;
- ☒ Criamos um **componente React**.

Na próxima aula, vamos aprofundar em Props no React! 🚀