

Projeto ReactJS com TypeScript: Manipulação de Eventos

Passo 1: Configurar o Projeto com Vite e TypeScript

1. Instale o Node.js e npm:

Certifique-se de que você tem o Node.js e o npm instalados. Você pode baixar a versão LTS do Node.js no site oficial.

2. Crie um Novo Projeto com Vite e TypeScript:

- Abra o terminal ou o prompt de comando.
- Execute o comando abaixo para criar um novo projeto com Vite e TypeScript. Substitua 'meu-projeto' pelo nome do seu projeto.

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

3. Navegue até o Diretório do Projeto:

- Entre no diretório do seu projeto recém-criado.

```
cd meu-projeto
```

4. Instale as Dependências:

- Execute o comando abaixo para instalar todas as dependências necessárias.

```
npm install
```

5. Inicie o Servidor de Desenvolvimento:

- Após a instalação das dependências, inicie o servidor de desenvolvimento.

Projeto ReactJS com TypeScript: Manipulação de Eventos

`npm run dev`

- O Vite iniciará um servidor de desenvolvimento e fornecerá um link (geralmente `http://localhost:3000`) para visualizar seu projeto no navegador.

Passo 2: Implementação de Manipulação de Eventos

Agora que o projeto está configurado, vamos criar um componente que exemplifique a manipulação de eventos, como cliques em botões e mudanças em campos de entrada.

1. Modifique o arquivo 'src/App.tsx':

Abra o arquivo 'src/App.tsx' e modifique o código para incluir exemplos de manipulação de eventos:

```
import React, { useState } from 'react';

import './App.css';

const EventHandlingExample: React.FC = () => {

  const [inputValue, setInputValue] = useState<string>('');

  const [clickCount, setClickCount] = useState<number>(0);

  // Função para manipular mudanças no campo de entrada

  const handleInputChange = (event: React.ChangeEvent<HTMLInputElement>) => {

    setInputValue(event.target.value);

  };

};
```

Projeto ReactJS com TypeScript: Manipulação de Eventos

```
// Função para manipular cliques no botão

const handleButtonClick = () => {

  setClickCount(clickCount + 1);

};

return (

  <div>

    <h1>Exemplo de Manipulação de Eventos</h1>

    <div>

      <label htmlFor='inputField'>Digite algo: </label>

      <input

        type='text'

        id='inputField'

        value={inputValue}

        onChange={handleInputChange}

      />

      <p>Você digitou: {inputValue}</p>

    </div>

    <div>

      <button onClick={handleButtonClick}>

        Clique aqui

      </button>

      <p>O botão foi clicado {clickCount} vezes</p>

    </div>

  </div>

);
```

Projeto ReactJS com TypeScript: Manipulação de Eventos

```
        </div>

    </div>

    );
};

function App() {

    return (

        <div className='App'>

            <EventHandlingExample />

        </div>

    );
}

export default App;
```

Explicação do Código

- EventHandlingExample Componente Funcional:

- Estado (state): Utiliza o hook useState para gerenciar dois estados:

- inputValue: Armazena o valor atual do campo de entrada.

- clickCount: Conta o número de vezes que o botão foi clicado.

- Manipulação de Eventos:

- handleInputChange: Esta função é chamada toda vez que o valor do campo de entrada muda. Ela recebe um evento do tipo `React.ChangeEvent<HTMLInputElement>` e atualiza o estado `inputValue` com o valor atual do campo.

Projeto ReactJS com TypeScript: Manipulação de Eventos

- `handleButtonClick`: Esta função é chamada toda vez que o botão é clicado. Ela incrementa o contador de cliques (`clickCount`).
- **Renderização**: O componente renderiza um campo de entrada e um botão. Ele também exibe o valor digitado no campo de entrada e o número de vezes que o botão foi clicado.
- **App Componente**: O componente principal (App) simplesmente renderiza o componente `EventHandlingExample`.

Estrutura do Projeto

O projeto deve ter a seguinte estrutura após a modificação:

meu-projeto/

??? node_modules/

??? public/

? ??? vite.svg

??? src/

? ??? App.css

? ??? App.tsx

? ??? index.css

? ??? main.tsx

? ??? vite-env.d.ts

??? .gitignore

??? index.html

??? package.json

??? tsconfig.json

Projeto ReactJS com TypeScript: Manipulação de Eventos

??? vite.config.ts

Conclusão

Este exemplo demonstra como manipular eventos comuns em React com TypeScript. A aplicação permite que o usuário digite texto em um campo de entrada e clique em um botão, com o estado sendo atualizado em resposta a esses eventos e refletido na interface do usuário.