

SuperProf - Plano de aula.

- Fundamentos do React.
 - Conceito SPA (Single Page Application)
 - Bundlers & Compilers
 - Babel (Compiler)
 - Webpack (Bundle)
 - Can I Use? (ES MODULE)
 - Hoje em dia não há tanto a necessidade de uso de webpack, pois a maioria dos navegadores vem se adaptando ao ECMA SCRIPTS MODULE.
- Estrutura de pastas do React
 - Entendendo cada arquivo
 - DOM? Document Object Model
 - Representação do HTML dentro do Javascript: Possibilita manipulação de códigos HTML com javascript. (createRoot).
- Components
 - O que são componentes react?
 - Conceito de componentes em react
 - JSX = Javascript + XML = HTML
 - React Fragment
 - Export default X Named export

```
export default function Post() {  
  return (  
    // Todo componente precisa retornar o conteúdo em HTML + JS  
    <h1>Olá, sou um componente Post!</h1>  
    // Dentro do return usando () posso colocar minhas tags HTML.  
  )  
}
```

- Propriedades | props

- O que são componentes props?
- Como passar valores a componentes?
- Como pegar esses valores dentro do componente?

```
export default function Post({ name }) {  
  // Através da propriedade "props" eu consigo fazer a desestruturação.  
  return (  
    <h1>{ name }</h1>  
    // Para indicar um conteúdo em javascript eu preciso passar dentro de { }.  
  )  
}
```

- CSS no React

- CSS global
- CSS escoped ou css module
- Como pegar esses valores dentro do componente?

- Hooks

- useState

```
const [name, setName] = useState("")  
  
// const [get, set]= useState("")  
// A primeira propriedade sempre irá armazenar o valor do meu estado.  
// A segunda propriedade é uma função "set" que armazena o valor dentro do estado.
```

- useEffect

```
useEffect(() => {  
  const resultado = fetch() // Realiza um fetch  
}, [])  
  
// A utilização do [] é para que seja passada as dependencias de component did mount
```

```
// Quando um valor da dependencia é atualizado o useEffect realizará um novo fetch
// Caso seja passado vazio, ele apenas irá fazer uma requisição ao inicializar o app
```

- React Router Dom
 - Criação de rotas
 - O que são providers?
 - Layouts reutilizáveis

```
npm install react-router-dom

# react-router-dom é uma biblioteca externa
# precisa ser instalada manualmente.
```

```
graph LR
  pages-->Home
  pages-->MovieDetails
```

```
// Home.jsx

export default function Home() {
  return (
    <h1>Pagina Home</h1>
  )
}
```

```
// MovieDetails.jsx

export default function MovieDetails() {
  return (
    <h1>Pagina Filme</h1>
  )
}
```

Vamos criar um componente Router, ele será responsável pelo gerenciamento das nossas rotas.

```
//Router.jsx

import { Routes, Route } from 'react-router-dom';

export default function Router() {
```

```
    return (  
      <Routes>  
        <Route path="/" element={<Home />} />  
        <Route path="/" element={<FilmeDetails />} />  
      </Routes>  
    )  
  }  
}
```

Precisamos importar agora o nosso BrowserRouter (Nosso provider das rotas)

```
// App.jsx  
  
import { BrowserRouter } from 'react-router-dom'  
  
export default function App() {  
  return (  
    <BrowserRouter>  
      <Router />  
    </BrowserRouter>  
  )  
}
```