

# ReactJS - React Router

# O que vamos ver hoje?

- Routes
- Route
- Link
- Slug

# O que é React Router

# React Router

- Para navegar entre as páginas de uma aplicação React precisaremos criar rotas, onde cada rota vai representar uma página.
- O React Router é a biblioteca de roteamento mais popular do React.

# Iniciando com React Router


# Iniciando

- Por se tratar de uma biblioteca, precisamos antes de tudo instalá-la:
  - `npm install react-router-dom`

# Iniciando

- Após a instalação precisamos configurá-lo.
- A primeira coisa que faremos é englobar todas as diferentes rotas que teremos em nossa aplicação. Para isso iremos utilizar o BrowserRouter que armazena a localização atual na barra de endereço do navegador usando URLs limpas e navega usando a pilha de histórico integrada do navegador.

# Iniciando



```
1 import React from "react";
2 import ReactDOM from "react-dom/client";
3 import Inicial from "../pages/inicial/index.jsx";
4 import { BrowserRouter } from "react-router-dom";
5
6 ReactDOM.createRoot(document.getElementById("root")).render(
7   <React.StrictMode>
8     <BrowserRouter>
9       <Inicial />
10    </BrowserRouter>
11  </React.StrictMode>
12 );
```



# Rotas

# Rotas

- Após englobar o BrowserRouter em nossa aplicação, a próxima etapa é definir suas rotas. Isso geralmente é feito no nível superior da sua aplicação mas podemos criar em um arquivo separado.
- É o que faremos agora.

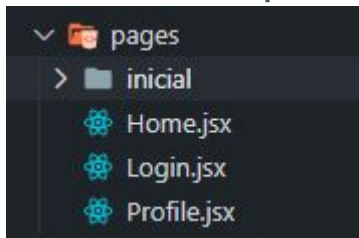
# Rotas

- Na pasta src crie um arquivo “Routes.jsx”.
- No arquivo “main.jsx” importe o arquivo de rotas e insira no componente:

```
1 import React from "react";
2 import ReactDOM from "react-dom/client";
3 import Inicial from "../pages/inicial/index.jsx";
4 import { BrowserRouter } from "react-router-dom";
5 import AppRoutes from "../Routes.jsx";
6
7 ReactDOM.createRoot(document.getElementById("root")).render(
8   <React.StrictMode>
9     <BrowserRouter>
10       <AppRoutes />
11       <Inicial />
12     </BrowserRouter>
13   </React.StrictMode>
14 );
```

# Rotas


- Antes de configurar as rotas vamos criar novas páginas que serão acessadas via rota.
- Vamos criar os arquivos Login, Profile e a Home dentro de src/pages. Todos eles terão a mesma estrutura, um componente React com um h1.
- Por enquanto vamos esquecer a page inicial.




# Rotas

- Antes de configurar as rotas vamos criar novas páginas que serão acessadas via rota.
- Vamos criar os arquivos Login, Profile e a Home dentro de src/pages.
- Todos eles terão a mesma estrutura, um componente React com um h1.
- Por enquanto vamos esquecer a page inicial.


# Rotas



```
1 import React from "react";
2
3 const Login = () => {
4   return <h1>Login</h1>;
5 };
6
7 export default Login;
```



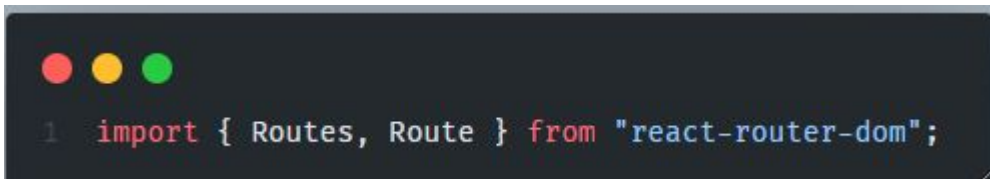
```
1 import React from "react";
2
3 const Profile = () => {
4   return <h1>Profile</h1>;
5 };
6
7 export default Profile;
```



```
1 import React from "react";
2
3 const Home = () => {
4   return <h1>Home</h1>;
5 };
6
7 export default Home;
```

# Rotas


- Agora já podemos configurar as nossas rotas.
- No arquivo “Routes.jsx” irems importar os componentes routes e route do react-router-dom:



```
1 import { Routes, Route } from "react-router-dom";
```

# Rotas

- Agora, importe as páginas Home, Login e Profile no arquivo "Routes.jsx":



```
1 import Home from "./pages/Home";  
2 import Login from "./pages/Login";  
3 import Profile from "./pages/Profile";
```



# Rotas

- O componente conterá todas as possíveis rotas de nossa aplicação.
- O componente Route aceita duas props:
  - path: é a URL para qual o link nos levará.
  - element: qual componente será renderizado.

# Rotas

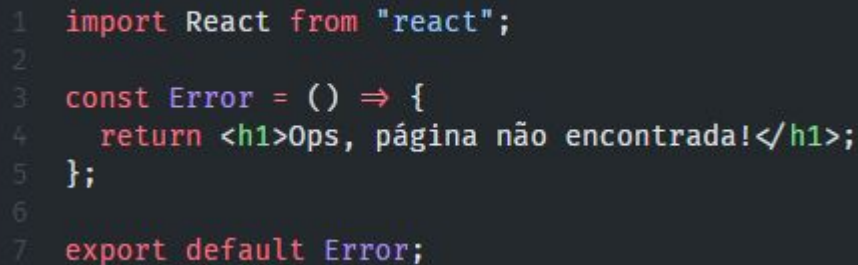
- Agora, instanciamos nossas rotas da seguinte forma:

```
1 import { Routes, Route } from "react-router-dom";
2 import Home from "../pages/Home";
3 import Login from "../pages/Login";
4 import Profile from "../pages/Profile";
5
6 const AppRoutes = () => {
7   return (
8     <Routes>
9       <Route path="/" element={<Home />} />
10      <Route path="login" element={<Login />} />
11      <Route path="profile" element={<Profile />} />
12    </Routes>
13  );
14 };
15
16 export default AppRoutes;
```

404

# 404

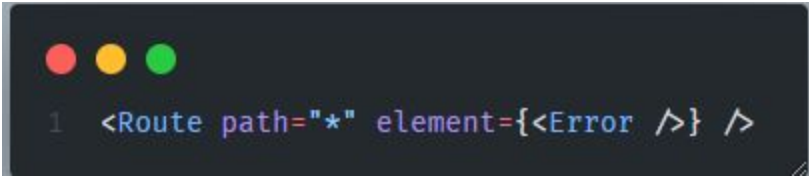
- Utilizada quando a url digitada é inválida.
- Vamos criar uma página chamada “Error.jsx”



```
1 import React from "react";  
2  
3 const Error = () => {  
4   return <h1>Ops, página não encontrada!</h1>;  
5 };  
6  
7 export default Error;
```

# 404

- Para configurar a rota, basta inserir um Route com path “\*”:



```
1 <Route path="*" element={<Error />} />
```

# Navegação por links

# Links

- Muitas vezes, em nossa aplicação podemos ter um menu de funcionalidades.
- Nesses casos utilizamos as rotas de navegação por links.
- Funciona de forma semelhante à utilizada no html.

# Links

- Muitas vezes, em nossa aplicação podemos ter um menu de funcionalidades.
- Nesses casos utilizamos as rotas de navegação por links.
- Funciona de forma semelhante à utilizada no html.



# Links

- Dentro de components vamos criar o arquivo “Menu.jsx”.
- Nele vamos importar o componente “Link” do react-route-dom.
- Vamos criar um menu com uma opção apontando para a página “Profile”.

# Links

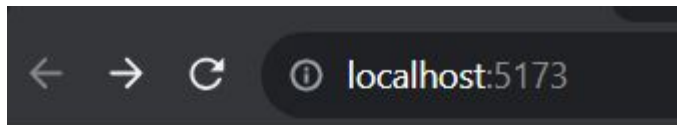
```
1 import React from "react";
2 import { Link } from "react-router-dom";
3
4 const Menu = () => {
5   return (
6     <nav>
7       <ul>
8         <li>
9           <Link to="/profile">Profile</Link>
10         </li>
11       </ul>
12     </nav>
13   );
14 };
15
16 export default Menu;
```

# Links

- No arquivo “Home.jsx” vamos importar o menu.

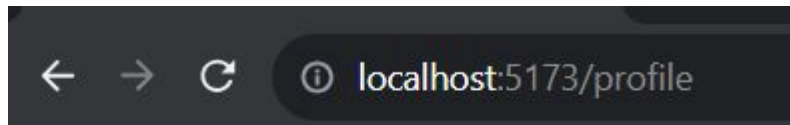
```
1 import React from "react";
2 import Menu from "../components/Menu";
3
4 const Home = () => {
5   return (
6     <
7       <Menu />
8       <h1>Home</h1>
9     </>
10   );
11 };
12
13 export default Home;
```

# Links



- [Profile](#)

**Home**



**Profile**

# Rotas dinâmicas

# Rotas dinâmicas

- Muitas vezes precisamos passar parâmetros de uma página a outra.
- Nesse caso usamos a funcionalidade “useParams” do react-router-dom.
- Usaremos o conceito de “slug” que é como uma props que terá seu valor recuperado da url.

# Rotas dinâmicas

- Muitas vezes precisamos passar parâmetros de uma página a outra.
- Nesse caso usamos a funcionalidade “useParams” do react-router-dom.
- Usaremos o conceito de “slug” que é como uma props que terá seu valor recuperado da url.
- Vamos modificar o arquivo “Profile.jsx”.

# Rotas dinâmicas

```
1 import React from "react";
2 import { useParams } from "react-router-dom";
3 const Profile = () => {
4   const { slug } = useParams();
5   return <h1>Profile de {slug}</h1>;
6 };
7
8 export default Profile;
```

← → ↻ ⓘ localhost:5173/profile/Rafael

## Profile de Rafael

```
1 <Route path="profile/:slug" element={<Profile />} />
```



# Resumo

# Resumo



- Utilizando React Router podemos criar rotas estáticas e dinâmicas por meio do uso de props na url.

# Dúvidas?



Programa  
**3000 TALENTOS TI**  
Obrigado(a)!