

Frontend & Backend разработка

Лекция 3

Маршрутизация в ReactJS

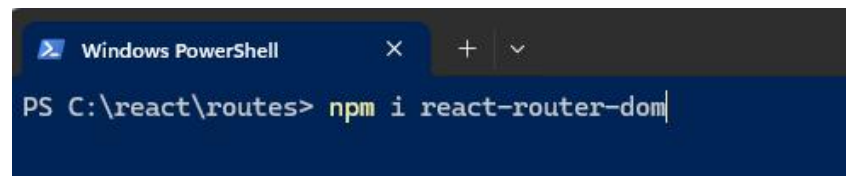
Р.В. Шамин

профессор кафедры индустриального программирования

Что такое маршрутизация?

Маршрутизация в ReactJS позволяет осуществлять навигацию между различными компонентами без перезагрузки всей страницы. Кроме того, маршрутизация позволяет передавать с помощью ссылок различные параметры.

В ReactJS для организации маршрутизации используется модуль `react-router-dom`, который необходимо установить следующей командой.



```
Windows PowerShell
PS C:\react\routes> npm i react-router-dom
```

Создадим новое React-приложение, которое будет содержать два компонента Home и About и на строим маршрутизацию, чтобы компонент Home вызывался по адресу "/", а About по адресу "/about".

Простейшая маршрутизация

```
1 import About from "../components/About";
2 import Home from "../components/Home";
3 import { BrowserRouter, Routes, Route } from 'react-router-dom';
4
5 function App() {
6
7   return (
8     <BrowserRouter>
9       <Routes>
10        <Route path="/" element={<Home />} />
11        <Route path="/about" element={<About />} />
12      </Routes>
13    </BrowserRouter>
14  );
15 }
16
17 export default App;
```



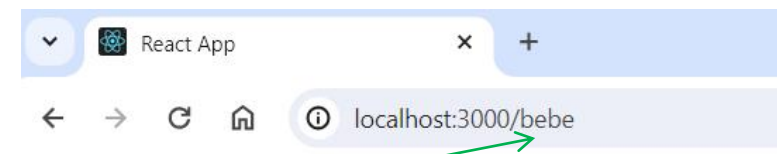
Главная страница



О компании...

«Не найдена страница»

```
1  import About from "../components/About";
2  import Home from "../components/Home";
3  import { BrowserRouter, Routes, Route } from 'react-router-dom';
4  import Not from "../components/Not";
5
6  function App() {
7
8    return (
9      <BrowserRouter>
10        <Routes>
11          <Route path="/" element={<Home />} />
12          <Route path="/about" element={<About />} />
13          <Route path="*" element={<Not />} />
14        </Routes>
15      </BrowserRouter>
16    );
17  }
18
19  export default App;
```



Страница не найдена!

```

1 export default function Nav() {
2   return (
3     <div>
4       | <a href="/">Home</a> | <a href="/about">О компании</a> |
5     </div>
6   )
7 }

```

```

1 import Nav from "../Nav";
2
3 export default function Home() {
4   return (
5     <div>
6       | <Nav/>
7       | Главная страница
8     </div>
9   )
10 }

```

← → ↺ 🏠 ⓘ localhost:3000

| [Home](#) | [О компании](#) |
Главная страница

```

1 import Nav from "../Nav";
2
3 export default function About () {
4   return (
5     <div>
6       | <Nav/>
7       | О компании...
8     </div>
9   )
10 }

```

← → ↺ 🏠 ⓘ localhost:3000/about

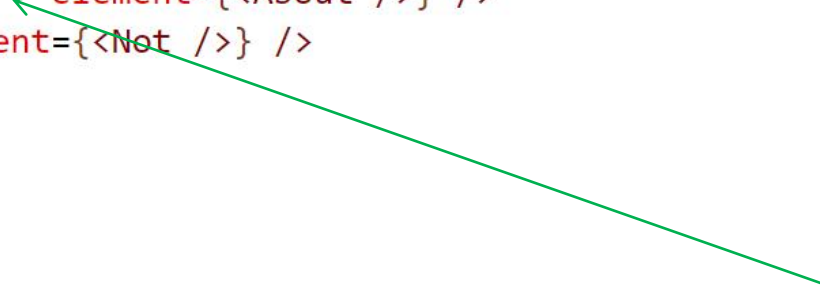
| [Home](#) | [О компании](#) |
О компании...

Дочерние маршруты

Маршрутизация React включает в себя и дочерние маршруты. Например, у нас может быть в странице «О компании» два дочерних маршрута - «Руководство» и «Контакты»: `/about/managment/` и `/about/contact/`

```
1  import About from "../components/About";
2  import Home from "../components/Home";
3  import { BrowserRouter, Routes, Route } from 'react-router-dom';
4  import Not from "../components/Not";
5
6  function App() {
7
8      return (
9          <BrowserRouter>
10             <Routes>
11                 <Route path="/" element={<Home />} />
12                 <Route path="/about/*" element={<About />} />
13                 <Route path="*" element={<Not />} />
14             </Routes>
15         </BrowserRouter>
16     );
17 }
18
19 export default App;
```

Добавляем *



Дочерние маршруты

```
1 export default function AboutNav() {
2   return (
3     <div>
4       <ol>
5         <li><a href="/about/managment">Руководство</a></li>
6         <li><a href="/about/contact">Контакты</a></li>
7       </ol>
8     </div>
9   )
10 }
```

```
6
7 export default function About () {
8   return (
9     <div>
10      <Nav/>
11      О компании...
12      <AboutNav/>
13      <Routes>
14        <Route path="/managment" element={<Managment/>}/>
15        <Route path="/contact" element={<Contact/>}/>
16      </Routes>
17    </div>
18  )
19 }
```

← → ↺ 🏠 ⓘ localhost:3000/about/managment

| [Home](#) | [О компании](#) |

О компании...

1. [Руководство](#)

2. [Контакты](#)

Руководство компанией.

Главный маршрут

```
10 <div>
11   <Nav/>
12   О компании...
13   <AboutNav/>
14   <Routes>
15     <Route index element={<AboutText/>}/>
16     <Route path="/managment" element={<Managment/>}/>
17     <Route path="/contact" element={<Contact/>}/>
18   </Routes>
19 </div>
```

Добавляем index

```
1 export default function AboutText() {
2   return (
3     <div>
4       Наша компания!
5     </div>
6   )
7 }
```

localhost:3000/about

| [Home](#) | [О компании](#) |
О компании...

1. [Руководство](#)
2. [Контакты](#)

Наша компания!

localhost:3000/about/contact

| [Home](#) | [О компании](#) |
О компании...

1. [Руководство](#)
2. [Контакты](#)

Контакты.

Ссылки

```
1 import { Link } from "react-router-dom";
2
3 export default function Nav() {
4   return (
5     <div>
6       | <Link to="/">Home</Link> | <Link to="/about">О компании</Link>
7     </div>
8   )
9 }
```

Nav.js

```
1 import { Link } from "react-router-dom";
2
3 export default function AboutNav() {
4   return (
5     <div>
6       <ol>
7         <li><Link to="/about/managment">Руководство</Link></li>
8         <li><Link to="/about/contact">Контакты</Link></li>
9       </ol>
10    </div>
11  )
12 }
```

AboutNav.js

Активные ссылки

Ссылки `<NavLink/>` могут использовать состояние ссылки, например, для отметания активной ссылки.

```
1 import { NavLink } from "react-router-dom";
2
3 export default function Nav() {
4   const setActive = ({isActive}) => ({fontWeight: isActive? 'bold': 'normal'});
5   return (
6     <div>
7       | <NavLink to="/" style={setActive}>Home</NavLink>
8       | <NavLink to="/about" style={setActive}>О компании</NavLink> |
9     </div>
10  )
11 }
```

← → ↻ 🏠 ⓘ localhost:3000

| Home | О компании |
Главная страница

← → ↻ 🏠 ⓘ localhost:3000/about

| Home | **О компании** |
О компании...

1. Руководство
2. Контакты

Наша компания!

Парсинг строки запроса

```
1 import { useParams, useSearchParams } from "react-router-dom";
2 import Nav from "./Nav";
3
4 export default function Home() {
5   const params = useParams();
6   const [searchParams, setSearchParams] = useSearchParams();
7   let x = searchParams.get("x");
8   let n = params.id;
9   let xn = Math.pow(x, n)
10
11   return (
12     <div>
13       <Nav/>
14       Степень = {n}
15       <br/>
16       x = {x}
17       <br/>
18       x в степени = {xn}
19     </div>
20   )
21 }
```

localhost:3000/3/?x=6

|Квадрат|Куб|

Степень = 3

x = 6

x в степени = 216

```
1 import Home from "../components/Home";
2 import { BrowserRouter, Routes, Route } from 'react-router-dom';
3 import Not from "../components/Not";
4
5 function App() {
6
7   return (
8     <BrowserRouter>
9       <Routes>
10         <Route path="/:id" element={<Home />} />
11         <Route path="*" element={<Not />} />
12       </Routes>
13     </BrowserRouter>
14   );
15 }
16
17 export default App;
```

localhost:3000/10/?x=2

|Квадрат|Куб|

Степень = 10

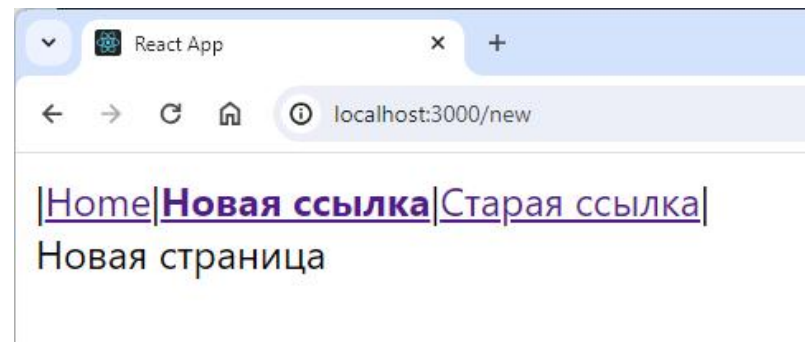
x = 2

x в степени = 1024

Переадресация

Ссылки внутри сайта могут меняться, но при этом старые ссылки могут храниться на внешних ресурсах и пользователь может переходить по старым ссылкам. Чтобы перенаправить пользователя на новые ссылки используется переадресация.

```
1  import Home from "../components/Home";
2  import { BrowserRouter, Routes, Route, Navigate } from 'react-router-dom';
3  import Not from "../components/Not";
4  import New from "../components/New";
5
6  function App() {
7
8    return (
9      <BrowserRouter>
10        <Routes>
11          <Route path="/" element={<Home />} />
12          <Route path="/new" element={<New/>} />
13          <Route path="/old" element={<Navigate to="/new" />} />
14          <Route path="*" element={<Not />} />
15        </Routes>
16      </BrowserRouter>
17    );
18  }
19
20  export default App;
```



Условная переадресация

```
8 function App() {
9
10   return (
11     <BrowserRouter>
12       <Routes>
13         <Route path="/" element={<Home />} />
14         <Route path="/green" element={<Green />} />
15         <Route path="/red" element={<Red />} />
16         <Route path="*" element={<Not />} />
17       </Routes>
18     </BrowserRouter>
19   );
20 }
```

App.js

```
1 import { Link } from "react-router-dom"
2
3 export default function Green() {
4   const styles = {
5     width: "100px",
6     height: "100px",
7     background: "green",
8   }
9
10  return (
11    <div>
12      <div style={styles}></div>
13      <Link to="/">Еще раз!</Link>
14    </div>
15  )
16 }
17 }
```

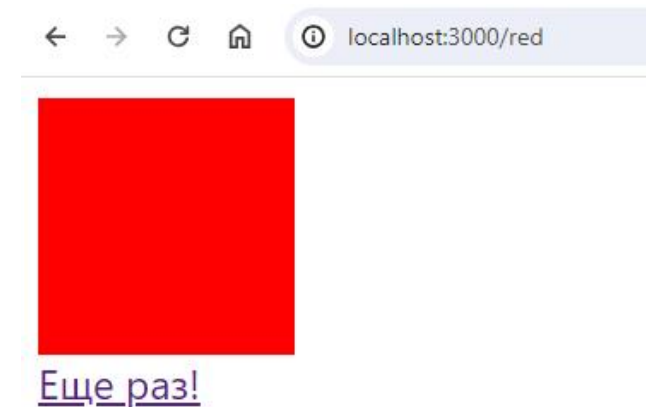
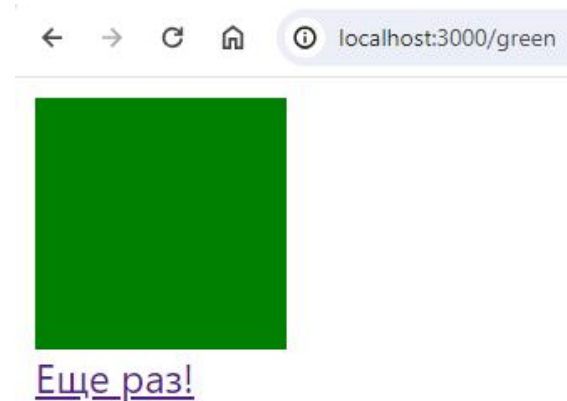
Green.js

```
1 import { Link } from "react-router-dom"
2
3 export default function Red() {
4   const styles = {
5     width: "100px",
6     height: "100px",
7     background: "red",
8   }
9
10  return (
11    <div>
12      <div style={styles}></div>
13      <Link to="/">Еще раз!</Link>
14    </div>
15  )
16 }
```

Red.js

```
1 import { Navigate } from "react-router-dom";
2
3 export default function Home() {
4   if (Date.now() % 2 === 0){
5     return <Navigate to="/green" />
6   } else {
7     return <Navigate to="/red" />
8   }
9
10 }
```

Home.js



Навигация в программе

```
1 import Home from "../components/Home";
2 import { BrowserRouter, Routes, Route } from 'react-router-dom';
3 import Not from "../components/Not";
4 import Calc from "../components/Calc";
5
6 function App() {
7
8   return (
9     <BrowserRouter>
10      <Routes>
11        <Route path="/" element={<Home />} />
12        <Route path="/calc" element={<Calc />} />
13        <Route path="*" element={<Not />} />
14      </Routes>
15    </BrowserRouter>
16  );
17 }
18
19 export default App;
```

App.js

```
1 import { useNavigate } from "react-router-dom";
2
3 export default function Home() {
4   const navigate = useNavigate();
5   return (
6     <button onClick={async event => {navigate(`/calc`)}}>Вычислить</button>
7   )
8 }
```

Home.js

