

### 3.3 获取路由的id(获取url的id值)

2018年11月23日 16:27

#### 获取路由的id

问题1: 路由形式:localhost:3000/#/order/ 123		往往需要取得
123 这个id的值		

(router\_demo/router4)

问题2:如果访问到了不存在的页面时, 如何匹配到 404 页面 (router\_demo/router5)

- 在router.js中添加动态路由即path:"/main/:value"用冒号定义的路由内容
- src\pages\route\_demo\router4\router.js

```
1 import React from "react";
2 import { HashRouter as Router, Route } from "react-router-dom"; //导入路由, 为HashRouter起别名: Router
3 import Info from "./info";
4 import Topic from "../router1/Topic";
5 import About from "../router1/About";
6 import Main from "./Main";
7 import Home from "./Home"; //4.0允许在路由中嵌套标签组件 这里是嵌套<Home>组件
9 /* 嵌套路由中添加 render方法加载
10    在子组件中进行嵌套路由
11 */
12 export default class IRouter extends React.Component {
13   render() {
14     return (
15       <Router>
16         <Home>
17         <Route
18           path="/main"
19           render={() =>
20             <Main>
21               { /* <div>this is a subchild element</div> */ }
22             <Route path="/main/:value" component={Info} />
23             </Main>
24           )
25         </Route>
26         <Route path="/about" component={About} />
27         <Route path="/topics" component={Topic} />
28       </Home>
29     </Router>
30   );
31 }
```

- Home1
- About1
- Topics1

this is Main page.

嵌套路由1

嵌套路由2

这里是测试动态路由功能 动态路由的值是: test-id

- src\pages\route\_demo\router4\Main.js

```
1 import React from 'react'
2 import {Link} from 'react-router-dom'
3 export default class Main extends React.Component {
4   render() {
5     return (
6       <div>
7         this is Main page.
8         <br/>
9         <Link to = "/main/test-id">嵌套路由1</Link>
10        <br/>
11        <Link to = "/main/456">嵌套路由2</Link>
12        <br/>
13        <hr/>
14        {this.props.children}
15      </div>
16    );
17  }
18 }
```

- src\pages\route\_demo\router4\info.js

- 在Info.js中获取定义的动态路由内容信息, 通过 {this.props.match.params. 路由的名称}

```
1 import React from 'react'
2 /*
3 {this.props.match.params.xxx}
4 xxx:是:后面的value <Route path="/main/:value" component={Info} />
5 此处写作: {this.props.match.params.value}
6 */
7 export default class Info extends React.Component {
8   render() {
```

```

5  此处写作: {this.props.match.params.value}
6
7  */
8  export default class Info extends React.Component {
9    render() {
10     return (
11       <div>
12         这里是测试动态路由功能
13         动态路由的值是: {this.props.match.params.value}
14       </div>
15     );
16   }
17 }

```

#### • src\pages\route\_demo\router4\Home.js

```

1  import React from "react";
2  import { Link } from "react-router-dom"; //引入路由
3
4
5  export default class Home extends React.Component {
6    //默认输出 对象 {}
7    render() {
8      return (
9        <div>
10          <ul>
11            <li>
12              <Link to="/main">Home1</Link>
13            </li>
14            <li>
15              <Link to="/about">About1</Link>
16            </li>
17            <li>
18              <Link to="/topics">Topics1</Link>
19            </li>
20          </ul>
21          <hr />
22          {this.props.children}
23        </div>
24      );
25    }
26  }

```

#### src\index.js

```

1  import React from 'react';
2  import ReactDOM from 'react-dom';
3  import './index.css';
4  import App from './App';
5  import * as serviceWorker from './serviceWorker';
6  import Life from './pages/demo/Life';
7  import Admin from './admin';
8
9  // import Home from './pages/route_demo/router1/Home';
10 // ReactDOM.render(<Home />, document.getElementById('root'));
11
12
13
14 // import Router from './pages/route_demo/router2/router';
15 // ReactDOM.render(<Router />, document.getElementById('root'));
16
17
18 // import Router from './pages/route_demo/router3/router';
19 // ReactDOM.render(<Router />, document.getElementById('root'));
20
21
22 import Router from './pages/route_demo/router4/router';
23 ReactDOM.render(<Router />, document.getElementById('root'));
24 // ReactDOM.render(<Admin />, document.getElementById('root'));
25
26 // If you want your app to work offline and load faster, you can change
27 // unregister() to register() below. Note this comes with some pitfalls.
28 // Learn more about service workers: http://bit.ly/CRA-PWA
29 serviceWorker.unregister();

```

#### src\pages\route\_demo\router5\Home.js

```

1  import React from "react";
2  import { Link } from "react-router-dom"; //引入路由
3
4  /* 1. 添加不存在的路由 imooc
5
6   2. 配置404页
7     <Route component={NoMatch} />
8   3. 定义NoMatch(组件) 即404页

```

#### (router\_demo/router5)

- Home1
- About1
- Topics1
- Imooc
- Imooc

```

9 4. 在路由中导入 NoMatch组件
10
11
12 定义NoMatch(组件) 即404页
13 */
14 export default class Home extends React.Component {
15   render() {
16     return (
17       <div>
18         <ul>
19           <li>
20             <Link to="/main">Home1</Link>
21           </li>
22           <li>
23             <Link to="/about">About1</Link>
24           </li>
25           <li>
26             <Link to="/topics">Topics1</Link>
27           </li>
28           <li>
29             <Link to="/imooc1">Imooc</Link>
30           </li>
31           <li>
32             <Link to="/imooc2">Imooc</Link>
33           </li>
34         </ul>
35         <hr />
36         {this.props.children}
37       </div>
38     );
39   }
40 }

```

#### src\pages\route\_demo\router5\info.js

```

1 import React from 'react'
2 /*
3 {this.props.match.params.xxx}
4 xxx:是后面的value <Route path="/main/:value" component={Info} />
5 此处写作: {this.props.match.params.value}
6 */
7 export default class Info extends React.Component {
8   render() {
9     return (
10       <div>
11         这里是测试动态路由功能
12         动态路由的值是: {this.props.match.params.value}
13       </div>
14     );
15   }
16 }

```

#### src\pages\route\_demo\router5\Main.js

```

1 import React from 'react'
2 import {Link} from 'react-router-dom'
3 export default class Main extends React.Component {
4   render() {
5     return (
6       <div>
7         this is Main page.
8         <br/>
9         <Link to = "/main/test-id">嵌套路由1</Link>
10        <br/>
11        <Link to = "/main/456">嵌套路由2</Link>
12        <br/>
13        <hr/>
14        {this.props.children}
15      </div>
16    );
17  }
18 }

```

#### src\pages\route\_demo\router5\NoMatch.js

```

1 import React from "react";
2
3 /* 1. 添加不存在的路由 imooc
4    2. 配置404页
5       <Route component={NoMatch} />
6    3. 定义NoMatch(组件) 即404页
7    4. 在路由中导入 NoMatch组件

```

```

5      <Route component={NoMatch} />
6
7      3. 定义NoMatch(组件) 即404页
8      4. 在路由中导入 NoMatch组件
9
10     定义NoMatch(组件) 即404页
11     */
12     export default class Home extends React.Component {
13       render() {
14         return (
15           <div>
16             404 No Pages.
17           </div>
18         );
19       }
20     }

```

src\pages\route\_demo\router5\router.js

```

1 import React from "react";
2 import { HashRouter as Router, Route ,Switch} from "react-router-dom";
3 import Info from "../info";
4 import Topic from "../router1/Topic";
5 import About from "../router1/About";
6 import Main from "../Main";
7 import Home from "../Home";
8 import NoMatch from "../NoMatch";
9 /* 1. 添加不存在的路由 imooc
10    2. 配置404页
11      <Route component={NoMatch} />
12    3. 定义NoMatch(组件) 即404页
13    4. 在路由中导入 NoMatch组件
14      + import NoMatch from "../NoMatch";
15
16    5. 添加Switch防止出现多个页面
17    + import {HashRouter as Router, Route ,Switch} from "react-router-dom";
18
19    定义NoMatch(组件) 即404页
20    */
21    export default class IRouter extends React.Component {
22      render() {
23        return (
24          <Router>
25            <Home>
26              <Switch>
27                <Route
28                  path="/main"
29                  render={() => {
30                    <Main>
31                      { /* <div>this is a subchild element</div> */ }
32                      <Route path="/main/:value" component={Info} />
33                    </Main>
34                  }
35                />
36              </Switch>
37            </Home>
38            <Route path="/about" component={About} />
39            <Route path="/topics" component={Topic} />
40
41            <Route component={NoMatch} />
42          </Router>
43        );
44      }
45    }

```

src\index.js

```

1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import './index.css';
4 import App from './App';
5 import * as serviceWorker from './serviceWorker';
6 import Life from './pages/demo/Life';
7 import Admin from './admin';
8
9 // import Home from './pages/route_demo/router1/Home';
10 // ReactDOM.render(<Home />, document.getElementById('root'));
11
12
13
14 // import Router from './pages/route_demo/router2/router';
15 // ReactDOM.render(<Router />, document.getElementById('root'));

```

```
16
17
18 // import Router from './pages/route_demo/router3/router';
19 // ReactDOM.render(<Router />, document.getElementById('root'));
20
21
22 // import Router from './pages/route_demo/router4/router';
23 // ReactDOM.render(<Router />, document.getElementById('root'));
24
25
26 import Router from './pages/route_demo/router5/router';
27 ReactDOM.render(<Router />, document.getElementById('root'));
28 // ReactDOM.render(<Admin />, document.getElementById('root'));
29
30 // If you want your app to work offline and load faster, you can change
31 // unregister() to register() below. Note this comes with some pitfalls.
32 // Learn more about service workers: http://bit.ly/CRA-PWA
33 serviceWorker.unregister();
```