

REACT ROUTER

React Router là gì?

SPA (single page application) hiện nay được coi là một xu thế để xây dựng một trang web bởi nhiều tính năng ưu việt, có rất nhiều thư viện cho phép xây dựng một trang SPA phổ biến nhất đó là ReactJS. Khi một trang web được xây dựng theo hướng SPA thì tất cả các UI của trang web sẽ được render ra một trang duy nhất, tùy vào từng trường hợp mà component sẽ được render.

Ngoài ra, chúng ta có thể sử dụng URL làm điều kiện xem xét rằng liệu component nào sẽ được render. Trong ReactJS, React Router là thư viện được xây dựng để thực hiện điều này.

React Router là một thư viện cho việc điều hướng URL tiêu chuẩn trong React, Nó cho phép chúng ta có thể đồng bộ UI với URL. Được thiết kế với API đơn giản, từ đó cho phép giải quyết các vấn đề về URL một cách nhanh chóng.

Sử dụng React Router trong ReactJS?

Cài đặt

```
npm install react-router-dom
```

Sử dụng các chức năng trong react-router

Sau đây là phần giới thiệu cách dùng BrowserRouter, Route, Link và NavLink trong react-router. Khi sử dụng bạn nhớ import chúng vào ở đầu trang .js nhé.

```
import {  
  BrowserRouter,  
  Routes,  
  Route,  
  Link,  
  NavLink,  
  Redirect,  
} from "react-router-dom";
```

Route

Trong react-router, mỗi route là 1 đường dẫn đưa vào một chức năng nào đó trong ứng dụng, thường là dẫn vào một component. Nghĩa là khi user truy cập theo đường dẫn bạn khai báo thì component tương ứng sẽ được render ra trang web. Ví dụ:

```

<article className="main">
  <Routes>
    <Route path="/" exact element={<TrangChu />} />
    <Route path="/gioithieu" element={<GioiThieu />} />
    <Route path="/lienhe" element={<LienHe />} />
    <Route element={<NotFound />} />
  </Routes>
</article>

```

Trong đó `path` là đường dẫn trên URL tùy ý bạn đặt, `element` là component sẽ được render, còn `exact` để báo rằng user phải dùng chính xác route như khai báo URL thì mới hoạt động.

Trong một route, bạn có thể dùng render để thực thi 1 function xử lý trước khi trả kết quả cho user:

```

<Route
  exact
  path="/download"
  element={dangNhapChua ? <DownLoad /> : <Navigate to="/" />}
/>

```

Link

Khi đã cài react-router, bạn có thể dùng thẻ `<Link>` để tạo các link cho ứng dụng, lệnh sẽ tạo tag `<a>` cho bạn và khi nhấp vào sẽ không nạp lại toàn trang. Thường dùng `<Link>` để tạo thanh menu. link. Ví dụ

```

<Link to="/">Trang chủ</Link>
<Link to="/lienhe">Liên hệ </Link>
<Link to="/gioithieu">Giới thiệu </Link>

```

Trong đó giá trị trong `to` giống như thuộc tính `href` của tag `a`

Navlink

`NavLink` giống `Link` nhưng có thêm hai thuộc tính là `activeclassname` và `activestyle`. Các thuộc tính này giúp định dạng cho link trùng khớp với `path`.

```

<NavLink exact activestyle={{ color: "red" }} to="/cart" className="gh">
  Giỏ hàng
</NavLink>

```

BrowserRouter và HashRouter

Là thành phần giúp quy định kiểu địa chỉ. BrowserRouter dùng địa chỉ con kiểu như <http://localhost:3000/lienhe> còn HashRouter thì dùng địa chỉ con ở sau dấu # như <http://localhost:3000/#lienhe>

Demo sử dụng React Router

1. Tạo project mới
2. Chuyển vào folder và cài đặt thư viện react-router-dom

```
npm install react-router-dom
```

3. Tạo folder components và tạo 3 components trong đó như sau:

- components/TrangChu.js:

```
const TrangChu = () => (  
  <div>  
    {" "  
    <h1>Đây là thông tin ở trang chủ</h1>{" "  
  </div>  
);  
export default TrangChu;
```

- components/GioHang.js:

```
const GioHang = () => (  
  <div>  
    {" "  
    <h1>Đây là trang giỏ hàng</h1>{" "  
  </div>  
);  
export default GioHang;
```

- components/ListSach.js:

```
const ListSach = () => (  
  <div>  
    {" "  
    <h1>Đây là trang list sách</h1>{" "  
  </div>  
);  
export default ListSach;
```

4. Import vào src/App.js

```
import TrangChu from "../components/TrangChu";
import GioHang from "../components/GioHang";
import ListSach from "../components/ListSach";
```

5. Code lại function App() của App.js:

```
function App() {
  return (
    <BrowserRouter basename="/">
      <div className="container">
        <header>
          <nav>
            <Link to="/">Trang chủ</Link>
            <Link to="/sach/thieunhi">Thiếu nhi</Link>
            <Link to="/sach/ngethystuatsong">Nghệ thuật sống</Link>
            <NavLink
              exact
              activestyle={{ color: "red" }}
              to="/cart"
              className="gh"
            >
              Giỏ hàng
            </NavLink>
          </nav>
        </header>
        <main>
          <article>
            <Routes>
              <Route path="/" exact element={<TrangChu />} />
              <Route path="/sach/thieunhi" element={<ListSach />} />
              <Route path="/sach/ngethystuatsong" element={<ListSach />} />
              <Route path="/cart" element={<GioHang />} />
            </Routes>
            <input />
          </article>
          <aside>Thông tin bổ sung</aside>
        </main>
        <footer>FOOTER</footer>
      </div>
    </BrowserRouter>
  );
}
```

Định dạng : code lại App.css

```
* { margin: 0; }
.container { width: 1280px; margin: auto; }
header { height: 90px; background: darkcyan; position: relative; }
header nav { position: absolute; bottom: 5px; left: 10px }
header nav a { color: white; text-decoration: none; margin-right: 10px; }
header nav a:hover { color: yellow; }
main {
  min-height: 600px;
  display: grid; grid-template-columns: 75% 25%;
}
main article { background: wheat; }
main aside { background: darkkhaki; }
footer { background: gray; height: 45px;
  display: flex; justify-content: center; align-items: center;
}
```