

# Training Day26 Report

(18 July 2024)

## React Router

**React Router** is a collection of navigational components within your application. With the latest version, React Router v6, there are several new features and improvements over the previous versions. This article will help you understand how to use React Router in a modern React application.

### What is a React Router?

**React Router** is a standard library for routing in [React](#). It enables the navigation among views of various components in a React Application, allows changing the browser URL, and keeps the UI in sync with the URL.

### Features of React Router

- **Declarative Routing:** React Router v6 uses the `Routes` and `Route` components to define routes declaratively, making the routing configuration simple and easy to read.
- **Nested Routes:** It supports nested routes, allowing for complex and hierarchical routing structures, which helps in organizing the application better.
- **Programmatic Navigation:** The `useNavigate` hook enables programmatic navigation, allowing developers to navigate between routes based on certain conditions or user actions.
- **Route Parameters:** It provides dynamic routing with route parameters, enabling the creation of routes that can match multiple URL patterns.
- **Improved TypeScript Support:** Enhanced TypeScript support ensures that developers can build type-safe applications, improving development efficiency and reducing errors.

### Components of React Router

#### 1. **BrowserRouter and HashRouter**

- **BrowserRouter:** Uses the HTML5 history API to keep your UI in sync with the URL.

- **HashRouter:** Uses the hash portion of the URL (i.e., window.location.hash) to keep your UI in sync with the URL.

## 2. Routes and Route

- **Routes:** A container for all your route definitions.
- **Route:** Defines a single route with a path and the component to render.

## 3. Link and NavLink

- **Link:** Creates navigational links in your application.
- **NavLink:** Similar to Link but provides additional styling attributes when the link is active.

# Steps to Use React Router

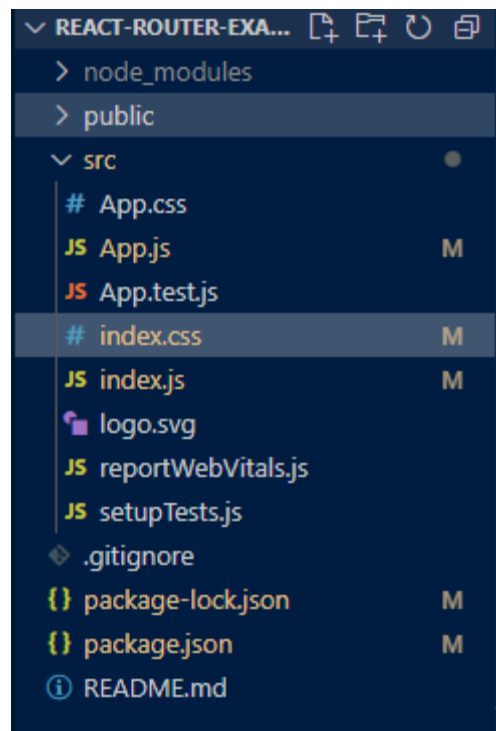
**Step 1:** Create React application using the following command.

```
npx create-react-app react-router-example
cd react-router-example
```

**Step 2:** Install react-router in your application write the following command in your terminal

```
npm install react-router-dom@6
```

## Folder Structure



*Folder Structure*

## Dependencies

```
"dependencies": {
  "@testing-library/jest-dom": "^5.17.0",
  "@testing-library/react": "^13.4.0",
  "@testing-library/user-event": "^13.5.0",
```

```
"react": "^18.3.1",
"react-dom": "^18.3.1",
"react-router-dom": "^6.24.1",
"react-scripts": "5.0.1",
"web-vitals": "^2.1.4"
}
```

## Example

*// src/App.js*

```
import React from "react";
import {
  BrowserRouter as Router,
  Routes,
  Route,
  Link,
  useNavigate,
  Outlet,
} from "react-router-dom";

const Home = () => {
  const navigate = useNavigate();

  return (
    <div>
      <h2>Home Page</h2>
      <button onClick={() =>
        navigate("/contact")}>Go to Contact</button>
    </div>
  );
};

const About = () => (
  <div>
    <h2>About Page</h2>
    <nav>
      <ul>
        <li>
          <Link to="team">Our Team</Link>
        </li>
        <li>
          <Link to="company">Our Company</Link>
        </li>
      </ul>
    </nav>
  </div>
);
```

```

    </nav>
    <Outlet />
  </div>
);

const Contact = () => <h2>Contact Page</h2>;
const Team = () => <h2>Team Page</h2>;
const Company = () => <h2>Company Page</h2>;

function App() {
  return (
    <Router>
      <nav>
        <ul>
          <li>
            <Link to="/">Home</Link>
          </li>
          <li>
            <Link to="/about">About</Link>
          </li>
          <li>
            <Link to="/contact">Contact</Link>
          </li>
        </ul>
      </nav>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
        <Route path="/team" element={<Team />} />
        <Route path="/company" element={<Company />} />
      </Routes>
      <Route path="/contact" element={<Contact />} />
    </Routes>
  </Router>
);
}

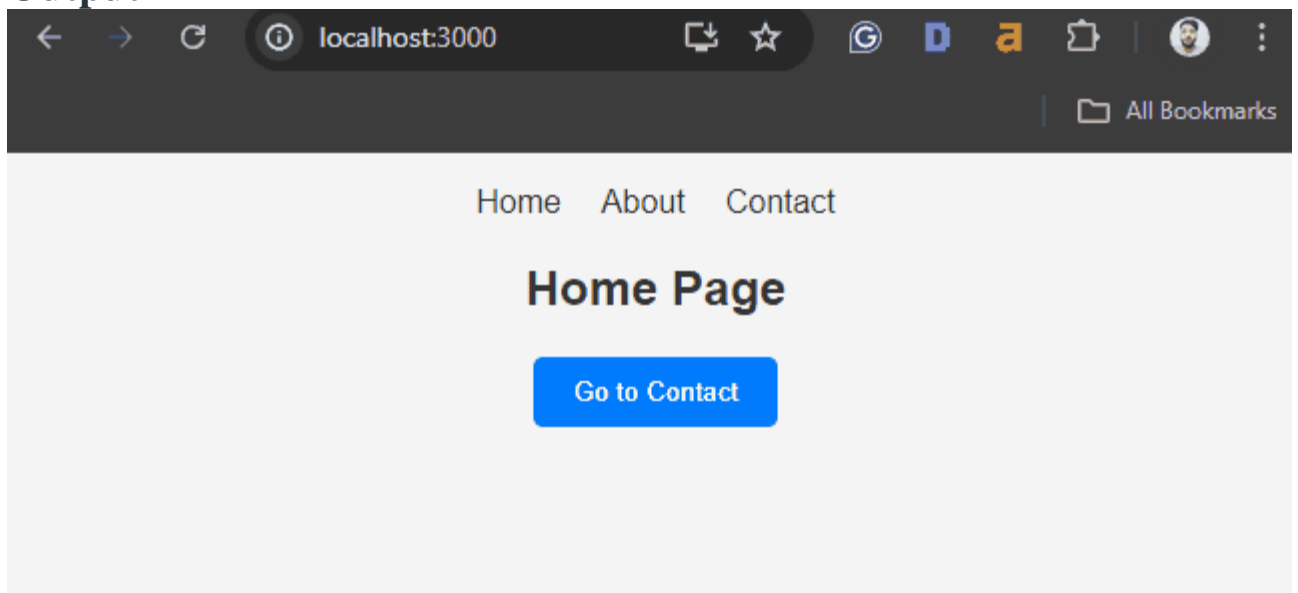
export default App;

```

**Step 4:** Run the application using the following command.

```
npm start
```

## Output



*React Router*