

## Step 1: Setting Up the Project

Create a simple React application using React Router 6. This app will have multiple pages, including a home page, an about page, and a contact page. We'll use the main features of React Router 6, including nested routes, route parameters, and programmatic navigation.

```
npx create-react-app react-router-demo
cd react-router-demo
npm install react-router-dom@6
```

## Step 2: Basic Setup

Let's create the basic structure of our app. Create the following components in the src folder:

- Home.js
- About.js
- Contact.js
- Layout.js

### Home.js

```
import React from 'react';

const Home = () => {
  return <h2>Home Page</h2>;
};

export default Home;
```

### About.js

```
import React from 'react';

const About = () => {
  return <h2>About Page</h2>;
};
```

```
export default About;
```

## **Contact.js**

```
import React from 'react';

const Contact = () => {
  return <h2>Contact Page</h2>;
};

export default Contact;
```

## **Layout.js**

```
import React from 'react';
import { Outlet, Link } from 'react-router-dom';

const Layout = () => {
  return (
    <div>
      <nav>
        <ul>
          <li>
            <Link to="/">Home</Link>
          </li>
          <li>
            <Link to="/about">About</Link>
          </li>
          <li>
            <Link to="/contact">Contact</Link>
          </li>
        </ul>
      </nav>
      <Outlet />
    </div>
  );
};

export default Layout;
```

## **Step 3: Configuring the Router**

Now let's configure the router in App.js:

### App.js

```
import React from 'react';
import { BrowserRouter as Router, Routes, Route } from
'react-router-dom';
import Home from './Home';
import About from './About';
import Contact from './Contact';
import Layout from './Layout';

function App() {
  return (
    <Router>
      <Routes>
        <Route path="/" element={<Layout />}>
          <Route index element={<Home />} />
          <Route path="about" element={<About />} />
          <Route path="contact" element={<Contact />} />
        </Route>
      </Routes>
    </Router>
  );
}

export default App;
```

### Step 4: Adding Nested Routes

Let's add a nested route under the About page. Create a new component Team.js:

### Team.js

```
import React from 'react';

const Team = () => {
  return <h2>Our Team</h2>;
};

export default Team;
```

Modify About.js to include a nested route:

### About.js

```
import React from 'react';
import { Outlet, Link } from 'react-router-dom';

const About = () => {
  return (
    <div>
      <h2>About Page</h2>
      <nav>
        <ul>
          <li>
            <Link to="team">Our Team</Link>
          </li>
        </ul>
      </nav>
      <Outlet />
    </div>
  );
};

export default About;
```

Update the routes in App.js:

### App.js

```
import React from 'react';
import { BrowserRouter as Router, Routes, Route } from
'react-router-dom';
import Home from './Home';
import About from './About';
import Contact from './Contact';
import Team from './Team';
import Layout from './Layout';

function App() {
  return (
    <Router>
      <Routes>
```

```

    <Route path="/" element={<Layout />}>
      <Route index element={<Home />} />
      <Route path="about" element={<About />}>
        <Route path="team" element={<Team />} />
      </Route>
      <Route path="contact" element={<Contact />} />
    </Route>
  </Routes>
</Router>
);
}

export default App;

```

## Step 5: Using Route Parameters

Let's create a page that displays a user profile based on a dynamic user ID. Create a new component `Profile.js`:

### Profile.js

```

import React from 'react';
import { useParams } from 'react-router-dom';

const Profile = () => {
  const { userId } = useParams();
  return <h2>Profile Page of User {userId}</h2>;
};

export default Profile;

```

Add a route for the profile page:

### App.js

```

import React from 'react';
import { BrowserRouter as Router, Routes, Route } from
'react-router-dom';
import Home from './Home';
import About from './About';
import Contact from './Contact';
import Team from './Team';

```

```

import Profile from './Profile';
import Layout from './Layout';

function App() {
  return (
    <Router>
      <Routes>
        <Route path="/" element={<Layout />}>
          <Route index element={<Home />} />
          <Route path="about" element={<About />}>
            <Route path="team" element={<Team />} />
          </Route>
          <Route path="contact" element={<Contact />} />
        <Route path="profile/:userId" element={<Profile />} />
      </Route>
    </Routes>
  </Router>
  );
}

export default App;

```

## Step 6: Programmatic Navigation

Add a button in the Home component to navigate to the profile page programmatically:

### Home.js

```

import React from 'react';
import { useNavigate } from 'react-router-dom';

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

  const goToProfile = () => {
    navigate('/profile/123');
  };

  return (
    <div>
      <h2>Home Page</h2>
      <button onClick={goToProfile}>Go to Profile</button>
    </div>
  );
}

```

```
        </div>
    );
};

export default Home;
```

## Step 7: Styling the App

You can add some basic CSS to make the app look nicer. Create a styles.css file in the src folder:

### styles.css

```
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

nav {
  background-color: #333;
  padding: 1rem;
}

nav ul {
  list-style: none;
  padding: 0;
}

nav ul li {
  display: inline;
  margin-right: 1rem;
}

nav ul li a {
  color: white;
  text-decoration: none;
}

nav ul li a:hover {
  text-decoration: underline;
}
```

```
h2 {  
  color: #333;  
}
```

Import the CSS in index.js:

### **index.js**

```
import React from 'react';  
import ReactDOM from 'react-dom';  
import './styles.css';  
import App from './App';  
  
ReactDOM.render(  
  <React.StrictMode>  
    <App />  
  </React.StrictMode>,  
  document.getElementById('root')  
>;
```