

🚩 Current Skill The Nested routes

The Nested routes

Up to this point, we have been working with routers that are small enough to be rendered entirely in a single file. But as an application grows in scope, it can be useful to split up the router and write routes nearer to where the related UI logic is written.

In other words, the nested routes allow us to, at the route level, have a parent component control the rendering of a child component.

The best example here is `facebook.com/messages/<whatever_path>`

If we open the `facebook.com/messages` and navigate between messages we'll notice that the first part of the url (`facebook.com/messages/t/`) is always fixed which we call the parent path while the rest of the url is called the child path.

The react router V6 library offer two way to implement the nested routes.

The first way is to use the `<Route>` component and `/*` in the path.

To make it easier to understand let's have an example.



```
import { Link, Route, Routes } from 'react-router-dom';
import './App.css';

// Component Home
const Home = () => (
  <>
    <h1>
      Home
    </h1>
  </>
)

// Component Profile
const Profile = () => (
  <>
    <h1>
      profile
```



```
    </h1>
  </>
)
// Component Account
const Account = () => (
  <>
    <h1>
      Account
    </h1>
  </>
)
```

```
// Component User
const User = () => {
  return (
    <div>
      <h1>User</h1>

      <nav>
        <Link to="profile">Profile</Link>
        <Link to="account">Account</Link>
      </nav>

      <Routes>
        <Route path="profile" element={<Profile />} />
        <Route path="account" element={<Account />} />
      </Routes>
    </div>
  );
};
```

```
const App = () => {
  return (
    <div className="App">
      <>
        <h1>React Router</h1>
```

```
<nav>

  <Link to="/home">Home</Link>

  <Link to="/user">User</Link>

</nav>

<Routes>

  <Route index element={<Home />} />

  <Route path="home" element={<Home />} />

  {/* The `/*` means every route that is relative to the user must render the */}

  <Route path="/user/*" element={<User />}>

    </Route>

  </Routes>

</>

</div>

);
}
```

```
export default App;
```

Nested and Dynamic Routes with React Router



In this video, you will find an overview about React Router and how we nest Routes

Nested and Dynamic Routes with React Router v6



< Previous

next >

