Nested Routes with outlet

The <Outlet> element is used as a placeholder. In this case an <Outlet>enables the Users component to render its child routes. Thus the<Outlet> element will render either a <Profile> or <Account> element depending on the current location.

Let's see the same example as before with outlet:

```
import { Link, Outlet, 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>
      <div>
        {/*
        This outlet will serve as a placeholder
        until the nested component will be rendered
         */}
        <Outlet />
      </div>
    </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 path="profile" element={<Profile />} />

          <Route path="account" element={<Account />} />

        </Route>

      </Routes>

    </>

  </div>

  );

}


export default App;
```