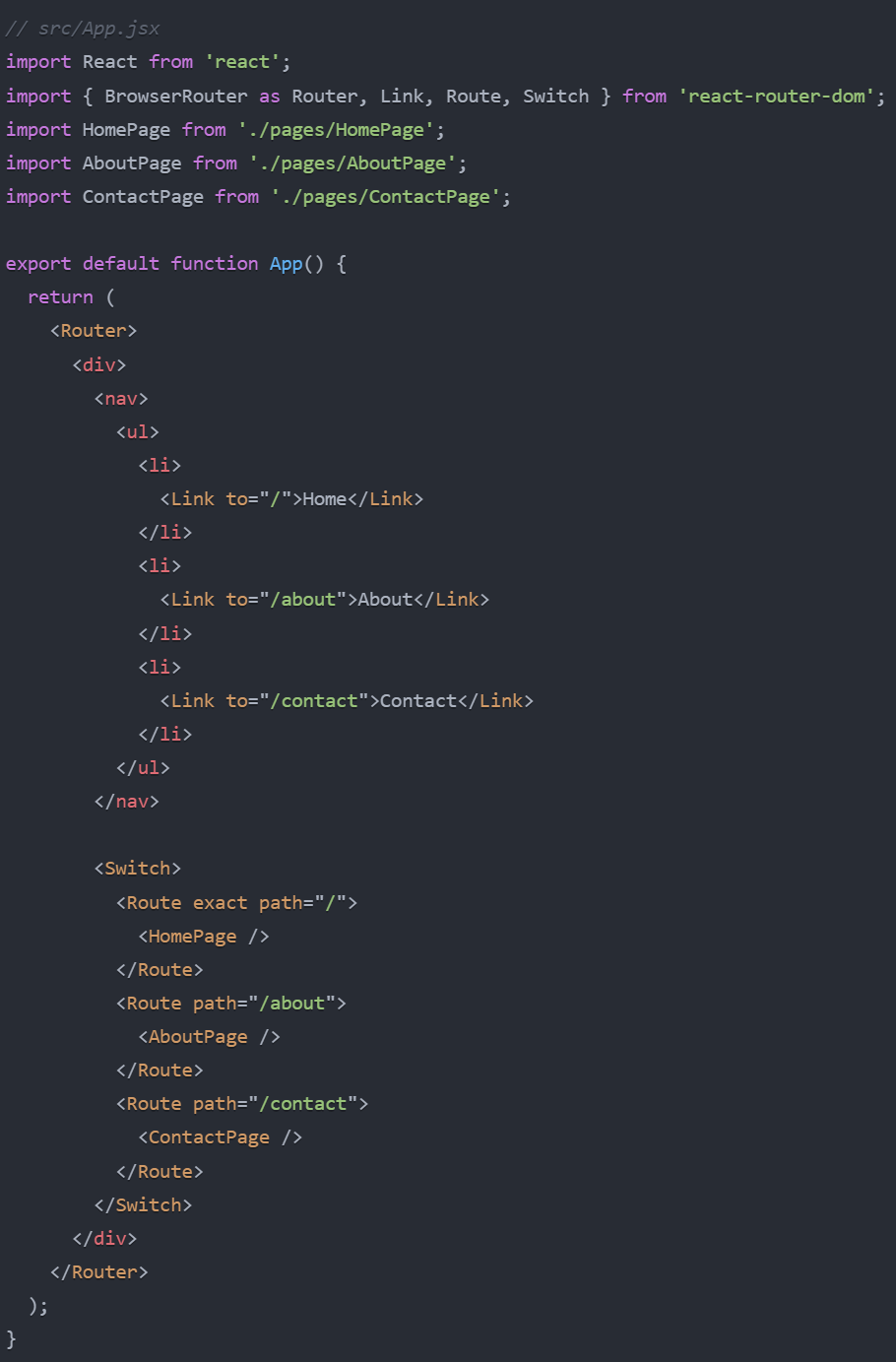
**Nested Routing with Module Federation**

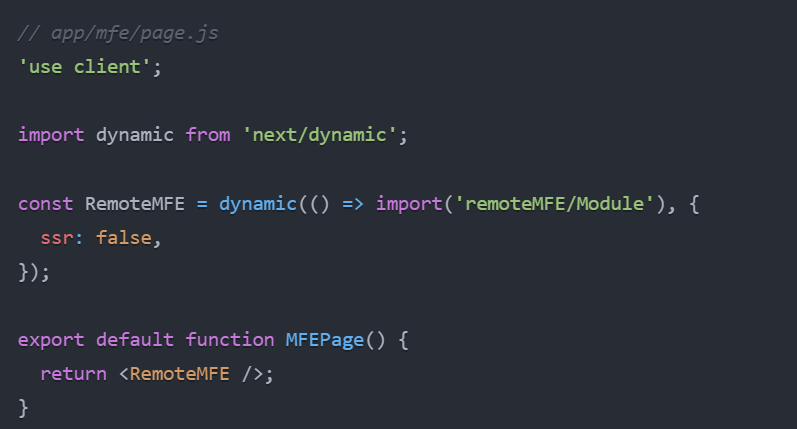
**Currently, the Next.js App Router does not have built-in support for nested routing within a federated module. In other words, if you have a federated module loaded into a parent application, the nested routes within that federated module may not work as expected.**

**Workarounds: There are a few workarounds and open discussions around this limitation. Some developers have suggested using shared layouts or wrapper components to handle nested routing within federated modules. Others have proposed using URL-based routing instead of file-based routing for federated modules.  
  
In summary, while Next.js App Router supports nested routing and module federation individually, the nested routing of module federated apps is currently limited.  
  
  
Code Samples:   
  
Here's an example that demonstrates the limitation of nested routing with module federation, where Next.js is the host and the child micro-frontend (MFE) is a React application with nested routes.**

**Child MFE   
  
**

**Host (Next.js Application)**

**In the Next.js host application, create a new page at app/mfe/page.js to render the federated React MFE:**

**  
  
  
Limitation**

**When you navigate to the /mfe route in the Next.js host application, the React MFE will be rendered. However, the nested routes within the MFE (/about and /contact) will not work as expected. This is because the Next.js App Router doesn't have built-in support for nested routing within a federated module.**

**The URLs /mfe/about and /mfe/contact in the host application will result in a 404 error, as Next.js expects these routes to be defined within the host application's file system structure.**

**The limitation of nested routing with module federation, where the Next.js host application cannot properly handle the nested routes defined within the child MFE.**