

React Router Practice Activity: "MyBooks App"

Build a simple book management app with multiple pages using React Router and TypeScript. The app should demonstrate routing between different views like a home page, book list, book detail page, and a "Not Found" page.

Requirements

Your app should include the following pages and routes:

1. **Home Page** (/)
 - A welcome message.
 - A link to the books page.
2. **Books Page** (/books)
 - A list of books (hardcoded array is fine).
 - i. author
 - ii. title
 - iii. description
 - Each book should be a link to a detailed view.
3. **Book Detail Page** (/books/:id)
 - Show the book's title and description, other fields.
 - If the book is not found, show a message like "Book not found".
4. **Not Found Page** (*)
 - A fallback route for undefined paths with a custom 404 message.

Suggested File Structure

```
src/  
├── components/  
│   └── BookList.tsx  
├── pages/  
│   ├── Home.tsx  
│   ├── Books.tsx  
│   ├── BookDetail.tsx  
│   └── NotFound.tsx  
├── types/  
│   └── Book.ts  
├── App.tsx  
└── main.tsx
```

Tasks

1. Set up routing:

Install `react-router-dom` and set up `BrowserRouter` in `main.tsx`.

2. Create a book type:

```
// types/Book.ts  
export interface Book {  
  id: string;  
  title: string;  
  author: string;  
  description: string;  
}
```

3. Create a hardcoded array of books (in a separate file or within a page).

4. Implement navigation links using `<Link>` or `<NavLink>`.

5. Use `useParams` to get the book ID from the URL and show its detail.

6. Use `<Routes>` and `<Route>` to define all your app routes.

Bonus

- Use `useNavigate` to redirect to the 404 page when a book ID is invalid.