# React Assignment: Blog Management App

## Objective:

Build a simple React app that demonstrates the use of React Routers and React Hooks. The app will have two pages:  
1. A Home Page that lists blog titles.  
2. A Blog Details Page that displays the content of a selected blog.

---

## Assignment Requirements

### 1. Setup

Create a React app using `create-react-app` or `Vite`.  
Install `react-router-dom` for routing.

### 2. Data

Use a static array of blog objects for this assignment. Example:

const blogs = [  
 { id: 1, title: "React Routers Basics", content: "React Router is a standard library for routing in React..." },  
 { id: 2, title: "Understanding React Hooks", content: "React Hooks are functions that let you use state and other React features without writing a class..." },  
 { id: 3, title: "JavaScript ES6 Features", content: "ES6 brought significant changes to JavaScript like arrow functions, destructuring, and more..." },  
];

### 3. Pages

Home Page:  
- Display the list of blog titles.  
- Each title should be a clickable link that navigates to the Blog Details Page for that specific blog.  
  
Blog Details Page:  
- Display the title and content of the selected blog based on its `id`.

### 4. Routing

Use `react-router-dom` to set up routes:  
- `/` → Home Page  
- `/blog/:id` → Blog Details Page

### 5. React Hooks Usage

Use the `useState` hook to manage a 'favorites' feature:  
- Add a button on the Blog Details Page to 'Add to Favorites'.  
- Clicking the button updates a 'favorites' state.  
  
Use the `useEffect` hook to log a message to the console whenever a blog is added to favorites.

---

## Bonus (Optional)

Create a Favorites Page accessible via a `/favorites` route that lists all favorited blogs.

---

## Deliverables

1. A fully functional React app with:  
- Proper routing.  
- State management using hooks.  
2. Code organized into components (e.g., `Home.js`, `BlogDetails.js`, etc.).  
3. Styling (minimal but clear; optional use of CSS framework).

---

## Tips

Use `useParams` from `react-router-dom` to get the `id` from the URL.  
Pass the `blogs` array as props or use React Context for a more advanced approach.  
Ensure code is clean and components are reusable.