**Dynamic Tabs Project**

**Introduction:**

The Dynamic Tabs project is a React-based application that allows users to dynamically add and remove tabs, with each tab displaying content fetched from an external API. This project demonstrates modern React concepts including state management, component composition, and API integration.

**Project Aim:**

The aim of this project is to create a dynamic tab interface where users can:

1. Add new tabs that fetch and display product information

2. Remove existing tabs

3. Switch between active tabs

4. Maintain a default tab that cannot be removed

**Project Summary:**

The Dynamic Tabs application is built using React and utilizes the Fake Store API to fetch product data. The application features:

- Dynamic tab creation and removal

- Responsive design

- API integration for tab content

- State management using React hooks

- Component-based architecture

**Tools and Frameworks Used:**

- **React** (JavaScript library for building user interfaces)

- **React Hooks** (useState, useEffect)

- **Fake Store API** (for product data)

- **Vite** (build tool)

- **HTML5/CSS3** (for structure and styling)

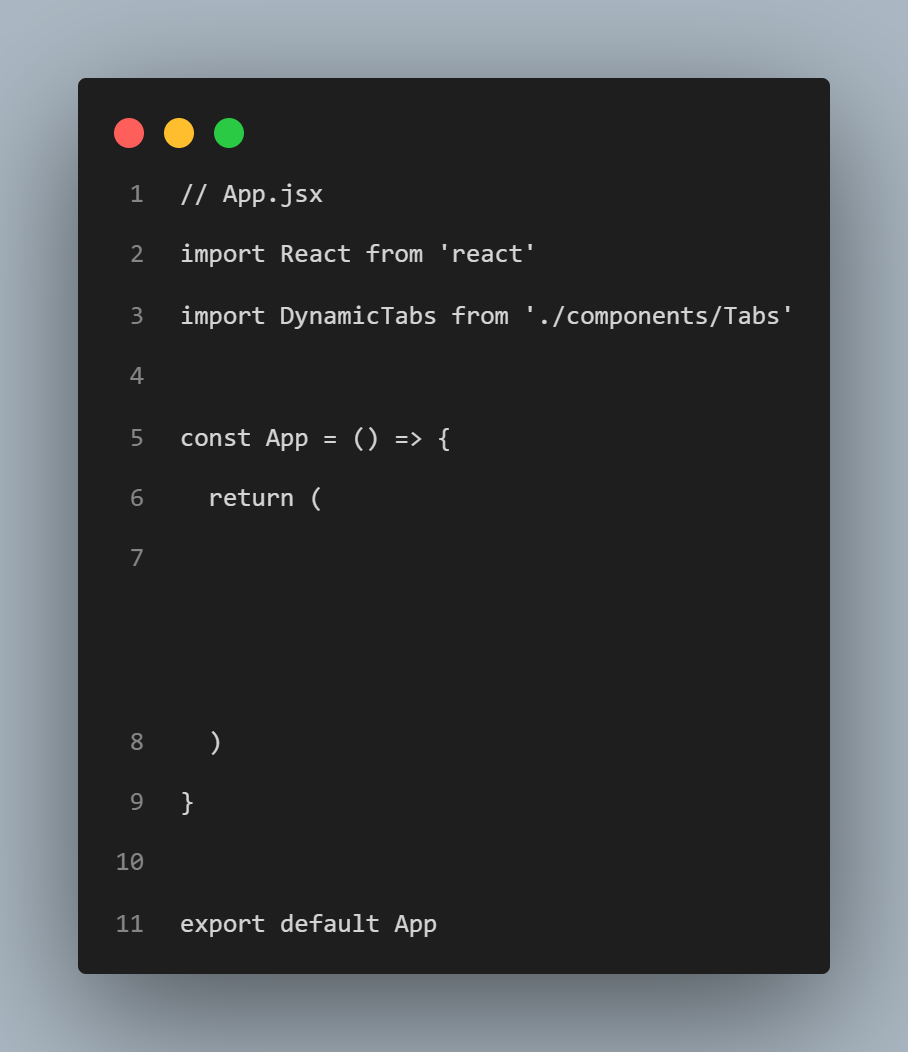
**Project Structure:**

The project follows a component-based architecture with the following structure:

A screenshot of a computer program

AI-generated content may be incorrect.

**- App.jsx:**



Explanation:

- The App component is the root component that renders the DynamicTabs component

- This simple structure allows for easy expansion of the application

**Project Implementation Process:**

**1. State Management:**

A computer screen with white text

AI-generated content may be incorrect.

- **tabs** stores an array of tab objects, each containing:

- **id**: Unique identifier for the tab

- **title:** Display text for the tab

- **content:** JSX content to be displayed in the tab

- **setTabs** is the function used to update the tabs array

- **activeTab** stores the id of the currently selected tab

- **setActiveTab** is the function used to change the active tab

- Initialized to 1 to show the first tab by default

**2. Adding Tabs:**

A screenshot of a computer

AI-generated content may be incorrect.

**Explanation:**

**3. Removing Tabs:**

A screenshot of a computer program

AI-generated content may be incorrect.

**Explanation: - Each new tab gets a unique ID based on current tab count**

**- Fetches product data from Fake Store API**

**- Creates a structured tab object with:**

**- Unique ID**

**- Title with tab number**

**-Content component displaying product details**

**- Updates state to include new tab**

**- Automatically activates the new tab**

**4. Rendering Tabs:**

A computer screen shot of a program code

AI-generated content may be incorrect.

**Results and Outputs:**

- Users can add new tabs that fetch and display product information

- Tabs can be removed except for the default tab

- The application maintains a clean and responsive interface

- Each tab displays product details including image, title, description, and price

**Conclusion:**

The Dynamic Tabs project successfully demonstrates modern React development practices. It showcases state management, component composition, and API integration in a practical, user-friendly interface. The project can be extended with additional features like tab reordering, persistent state, or integration with different APIs.