Problem Statement 1: Task Manager App

Create a **Task Manager** application using React and Axios that allows users to perform CRUD operations on tasks. The app should have the following features:

1. **Fetch & Display Tasks**

o   Use Axios to fetch tasks from a mock API (e.g., JSONPlaceholder or a custom mock server).

o   Display tasks in a list with their title, description, and status (completed/pending).

2. **Add a New Task**

o   Provide a form to add a new task (title, description, status).

o   Use Axios POST to send the new task to the API.

3. **Update Task Status**

o   Allow users to toggle the status (completed/pending) of a task.

o   Use Axios PUT/PATCH to update the task on the server.

4. **Delete a Task**

o   Add a delete button for each task.

o   Use Axios DELETE to remove the task from the API.

5. **Error Handling & Loading States**

o   Show a loading spinner while fetching/updating data.

o   Display error messages if API requests fail.

API Endpoint (Example):

https://jsonplaceholder.typicode.com/todos

(Or use mockapi.io to create a custom endpoint.)

Problem Statement 2: User Profile Dashboard

Build a **User Profile Dashboard** where users can view, add, edit, and delete user profiles using React and Axios.

1. **Fetch & Display Users**

o Fetch a list of users from a mock API (e.g., JSONPlaceholder or `mockapi.io`).

o Display users in a table/card layout with `name`, `email`, and `avatar` (if available).

2. **Add a New User**

o Create a form to add a new user (`name`, `email`, `avatar URL`).

o Use Axios `POST` to add the user to the API.

3. **Edit User Details**

o Allow users to edit details (e.g., name or email) via a modal/form.

o Use Axios `PUT/PATCH` to update the user data.

4. **Delete a User**

o Add a delete button for each user.

o Use Axios `DELETE` to remove the user from the API.

5. **Search & Filter** (Bonus)

o Implement a search bar to filter users by name.

API Endpoint (Example):

https://jsonplaceholder.typicode.com/users