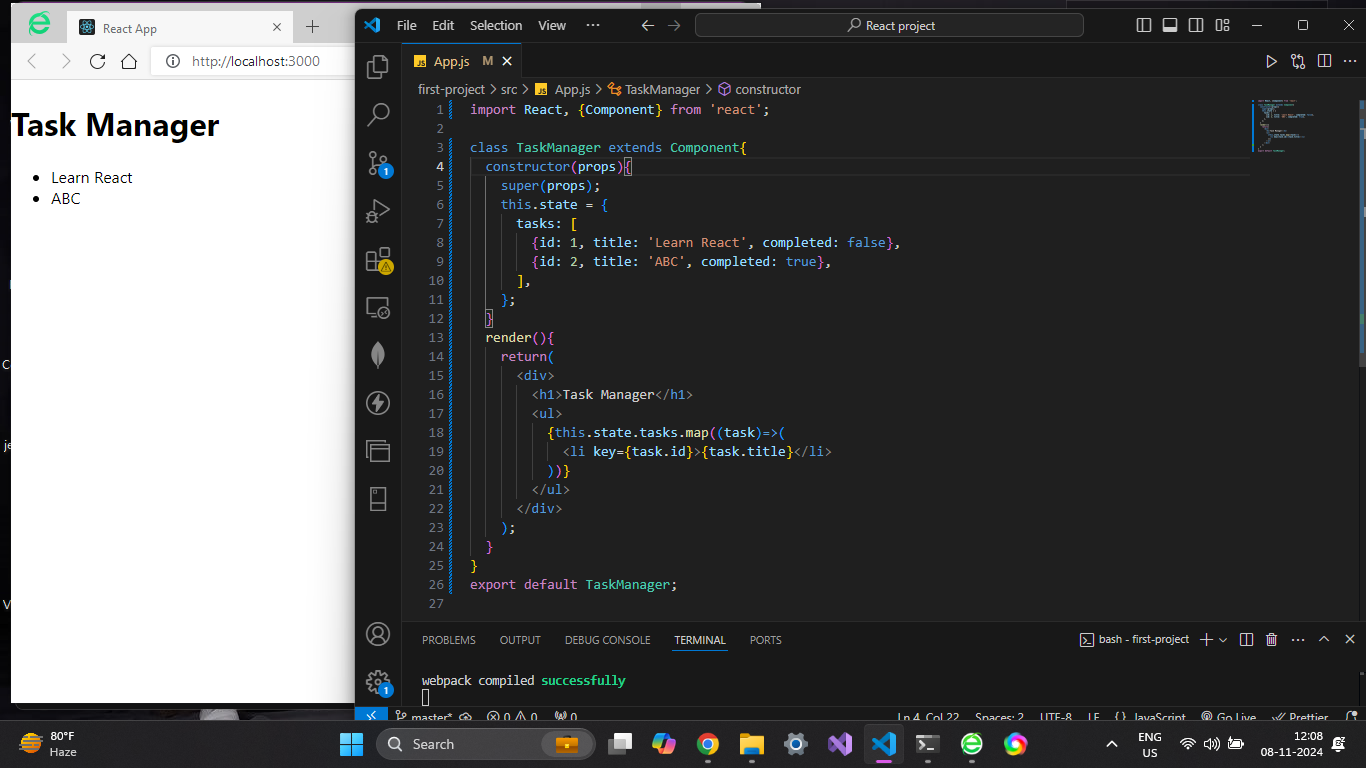
|  |  |  |
| --- | --- | --- |
| **Name** | **:** | **Patel Jenishkumar Kishorbhai** |
| **Cource** | **:** | **M.Sc. ICT(Sem-1)** |
| **Roll No** | **:** | **50** |
| **Subject** | **:** | **Application Development Using React.js** |
| **Practical Assignment 3** | | |

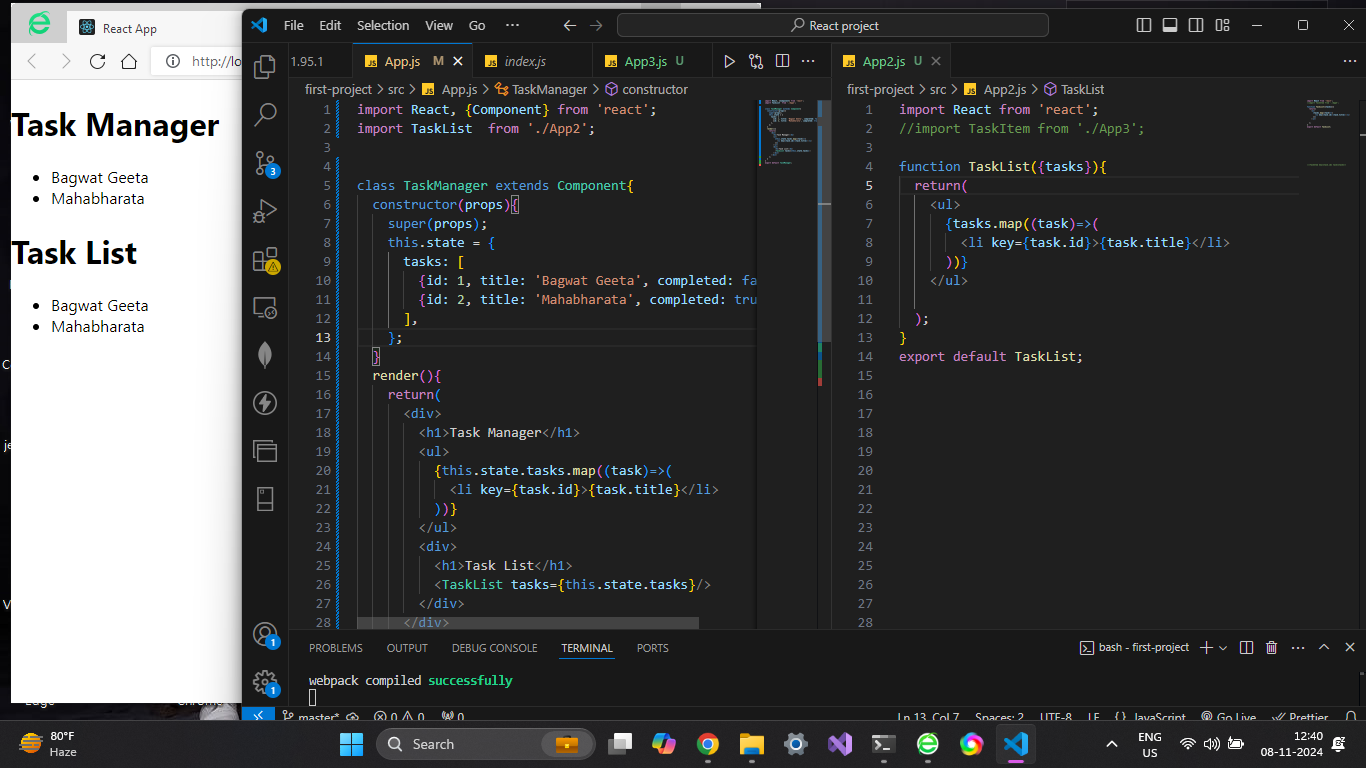
**Build a Small Task Management Website in React.js**

**Create a small website for managing tasks. The website should have the following features:**

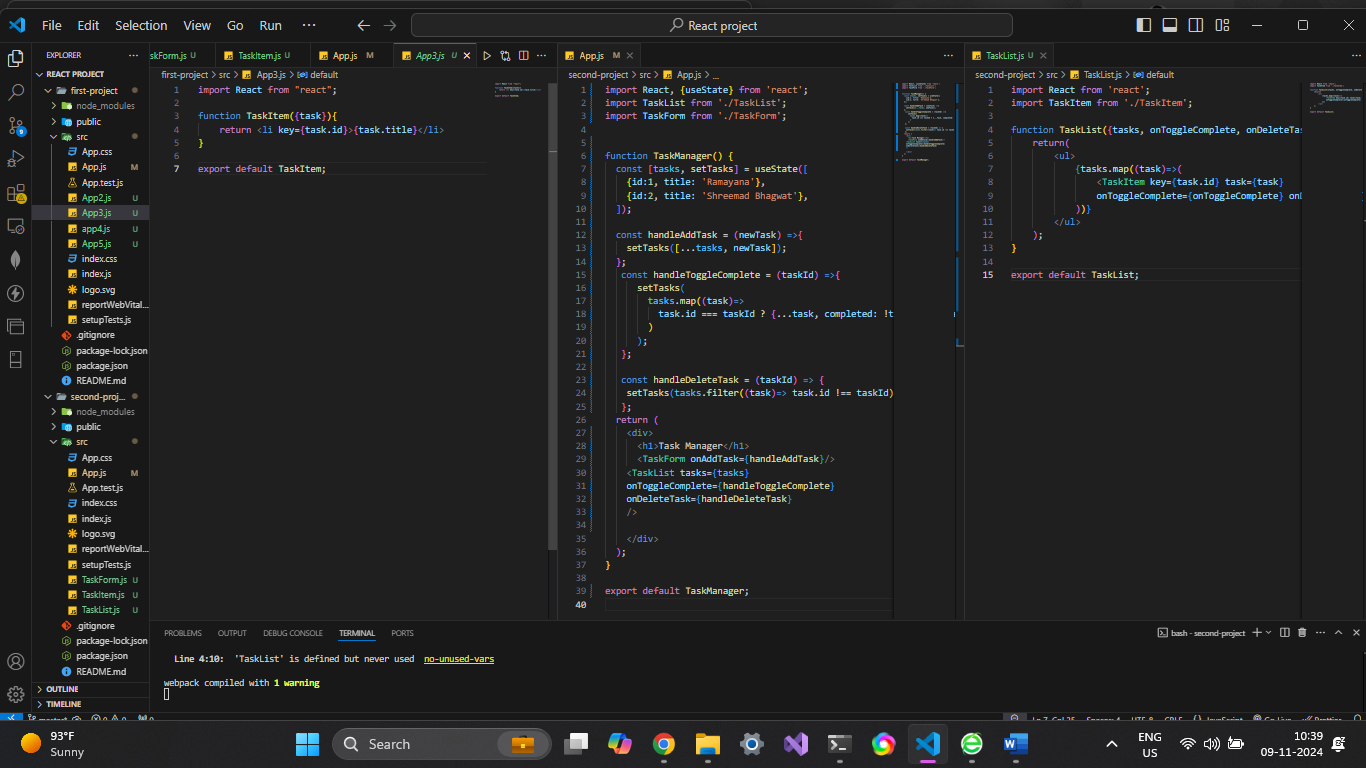
**1. Class Components: Implement a class component called TaskManager that will serve as the main container for managing tasks. This component should have a state to store the list of tasks.**

****

**2. Functional Components: Create a functional component called TaskList responsible for displaying a list of tasks. It should receive the list of tasks as props from the TaskManager component.**

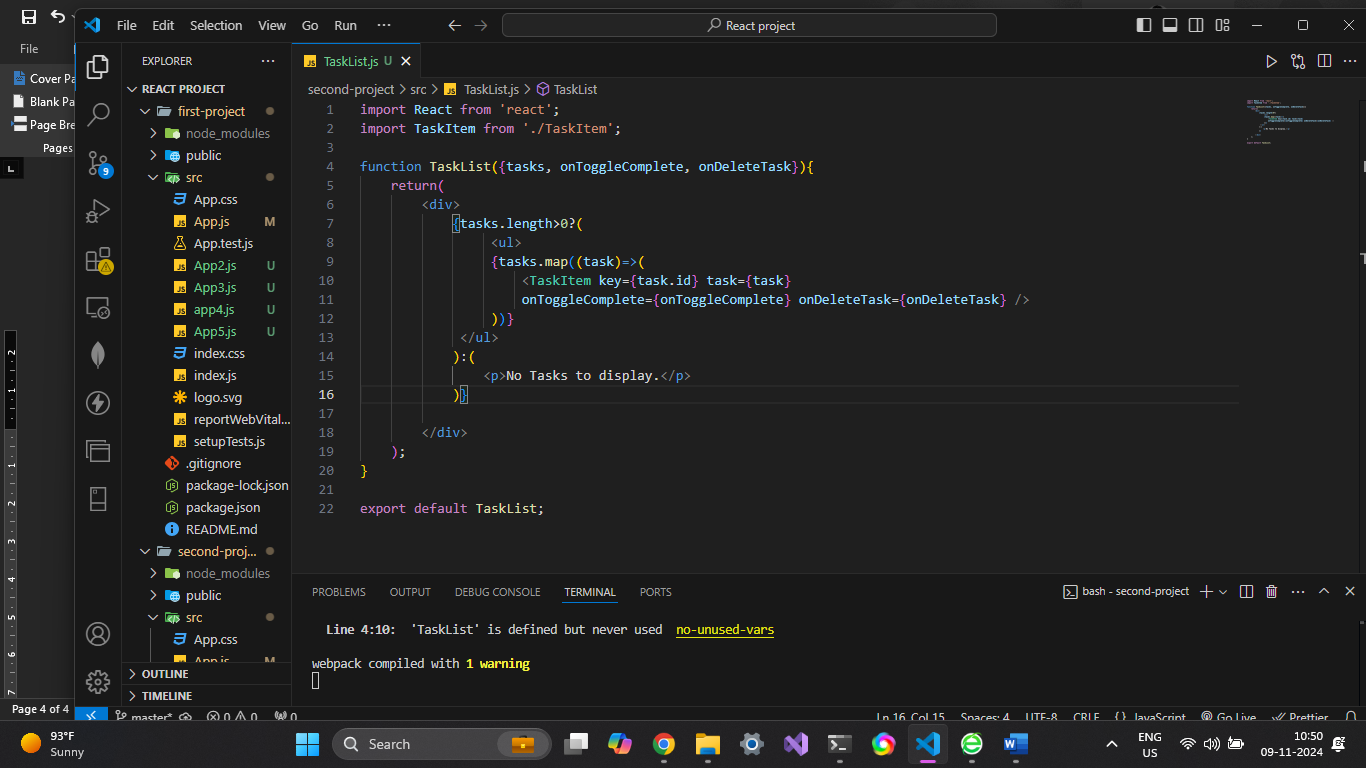
****

**3. Nested Components: Inside the TaskList component, create a child component called TaskItem that represents an individual task.**

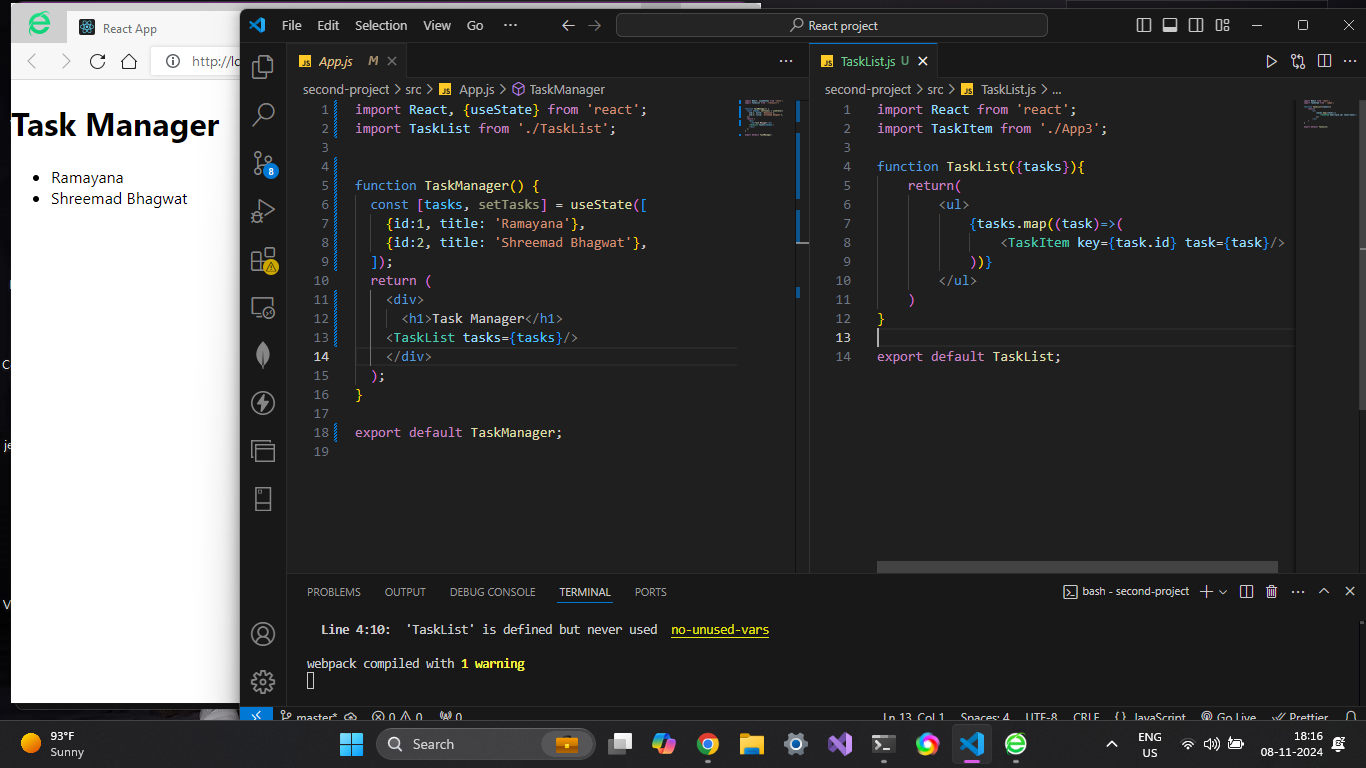
****

**4. Conditional and Looping Constructs: Implement conditional rendering in the TaskList**

**component. If there are no tasks, display a message like &quot;No tasks to display.&quot; Use a loop (e.g., map) to render each TaskItem.**

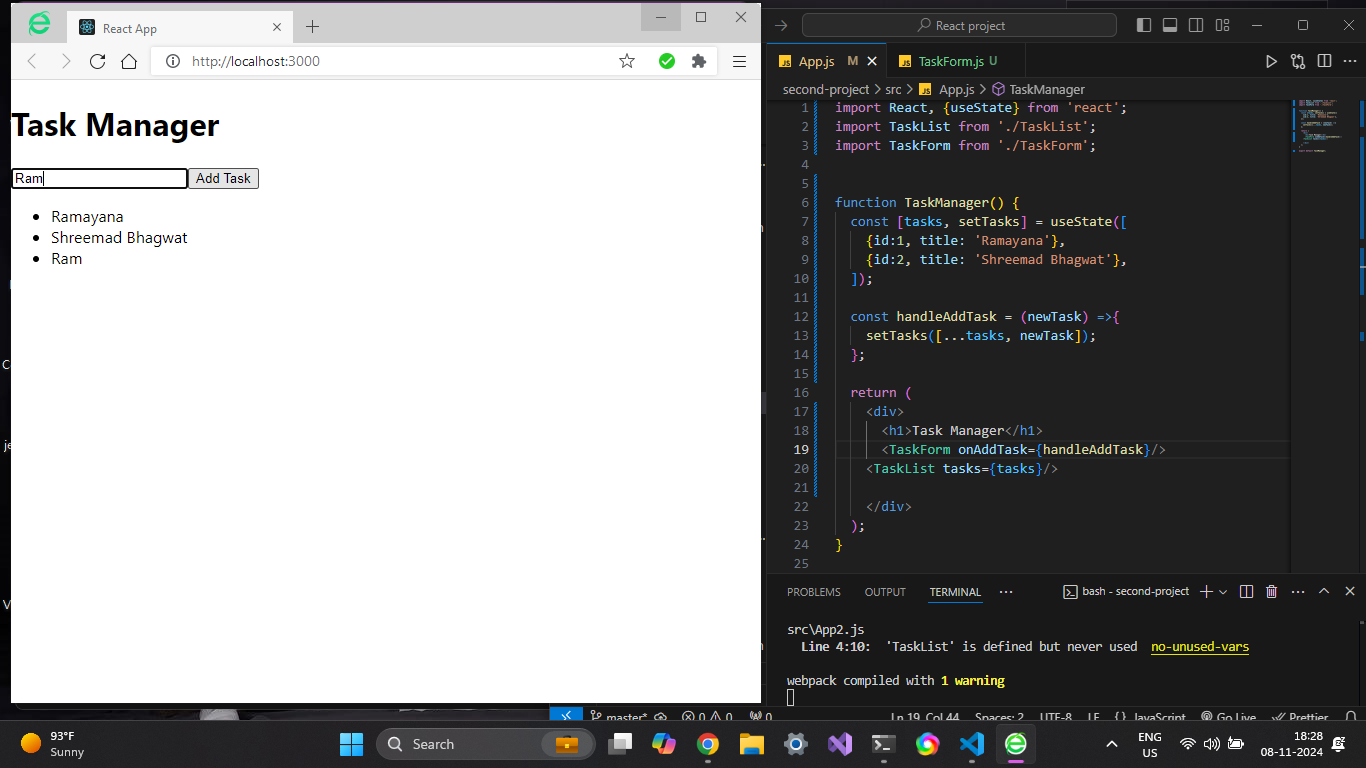
****

**5. State: In the TaskManager component, manage the list of tasks using state. Initially, you can provide some sample tasks within the state.**

****

**6. Props: Pass the list of tasks as props from the TaskManager component to the TaskList**

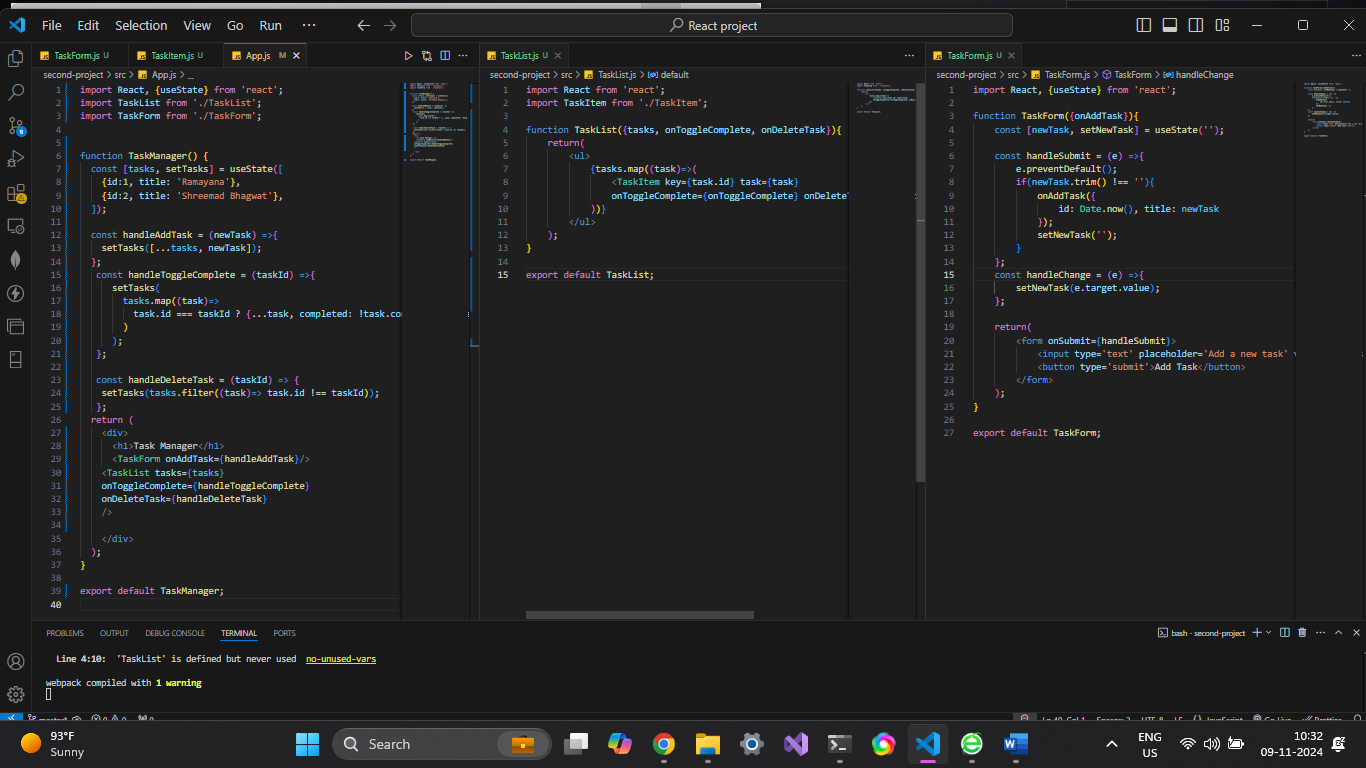
**component. Each TaskItem should receive a task object as a prop.**

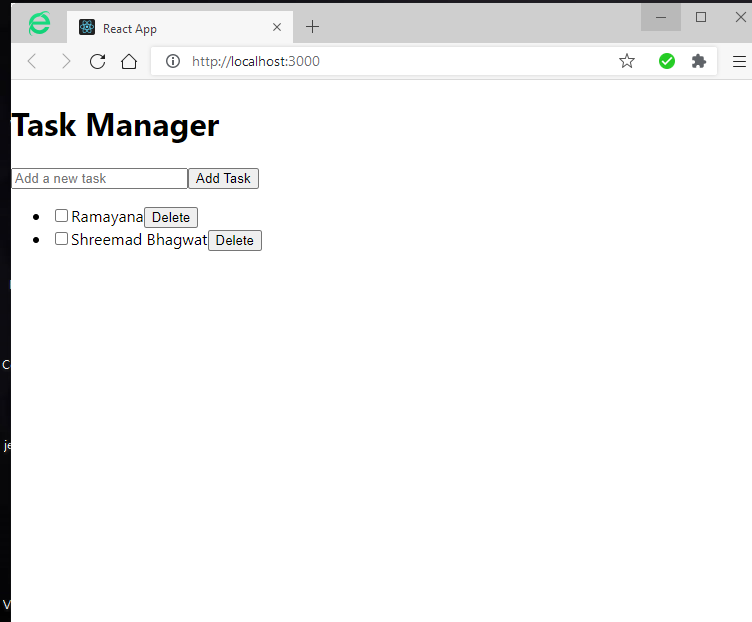
****

**7. Form (Controlled Component): Create a form component called TaskForm within the**

**TaskManager component. This form should allow users to add new tasks. Use controlled**

**components to manage form input values.**

****

****