

| Title: Demonstrate React Hooks |
| --- |

**AIM:** To Implement the React Axios

**Problem Definition:**

-Demonstrate the Concept of React JS With the help of Example.

\*(Students have to perform the task assigned within group and demonstrate the same).

**Resources used:**

* Google
* VS code

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected OUTCOME of Experiment:**

**CO 1:**.Build full stack applications in React using the MERN technologies.

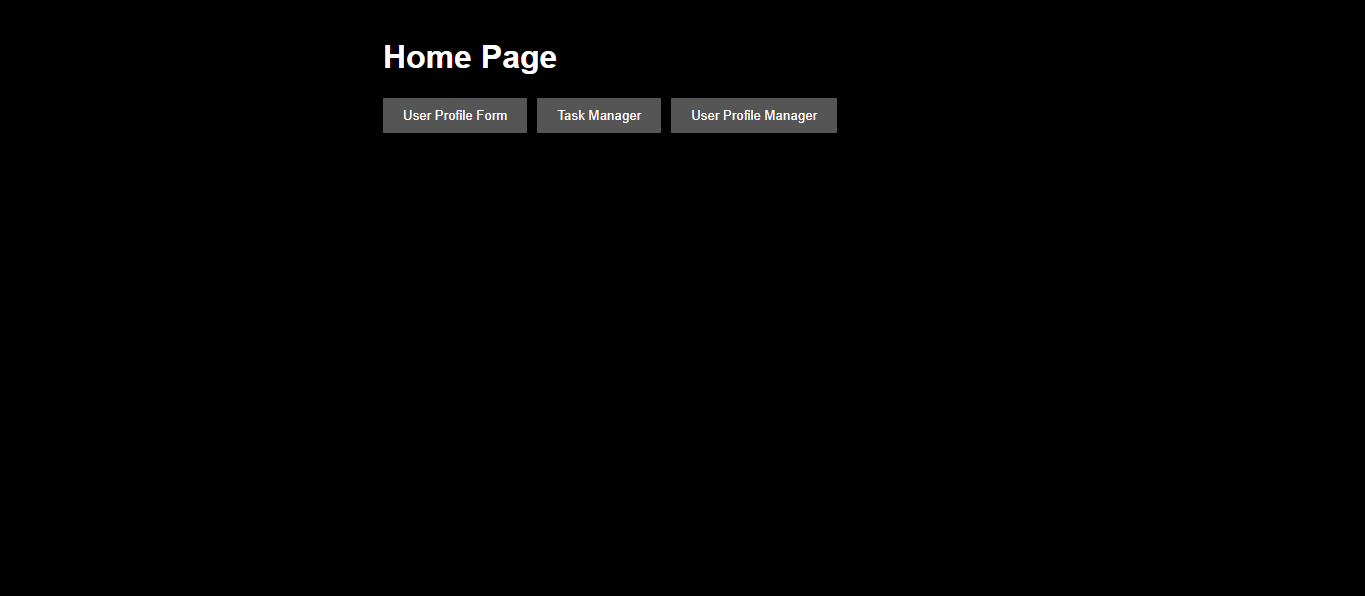
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

1. Shelly Powers Learning Node O’ Reilly 2 nd Edition, 2016.

**Pre Lab/ Prior Concepts:**

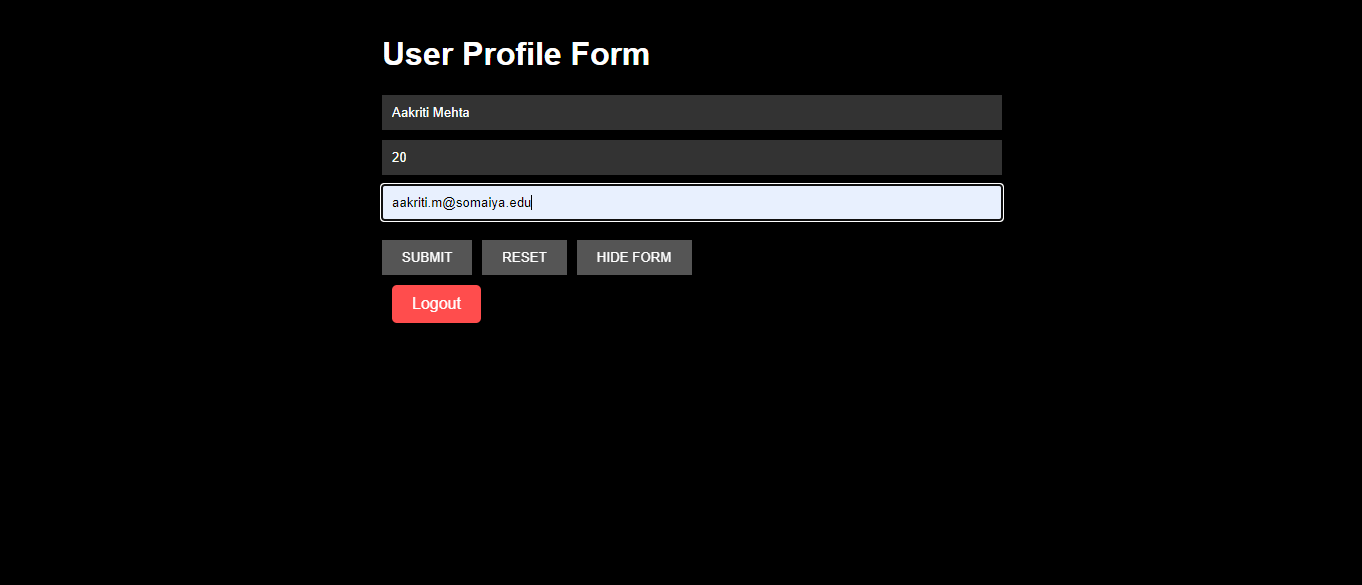
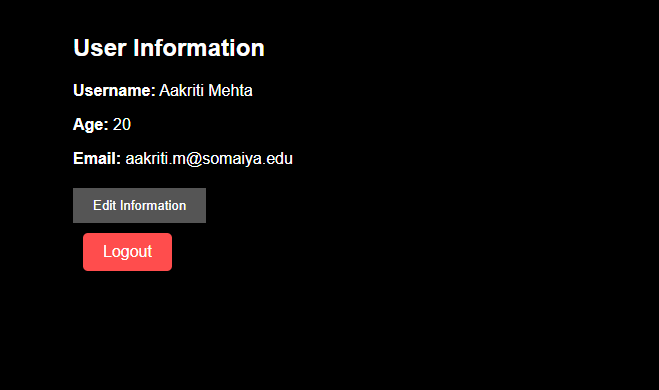
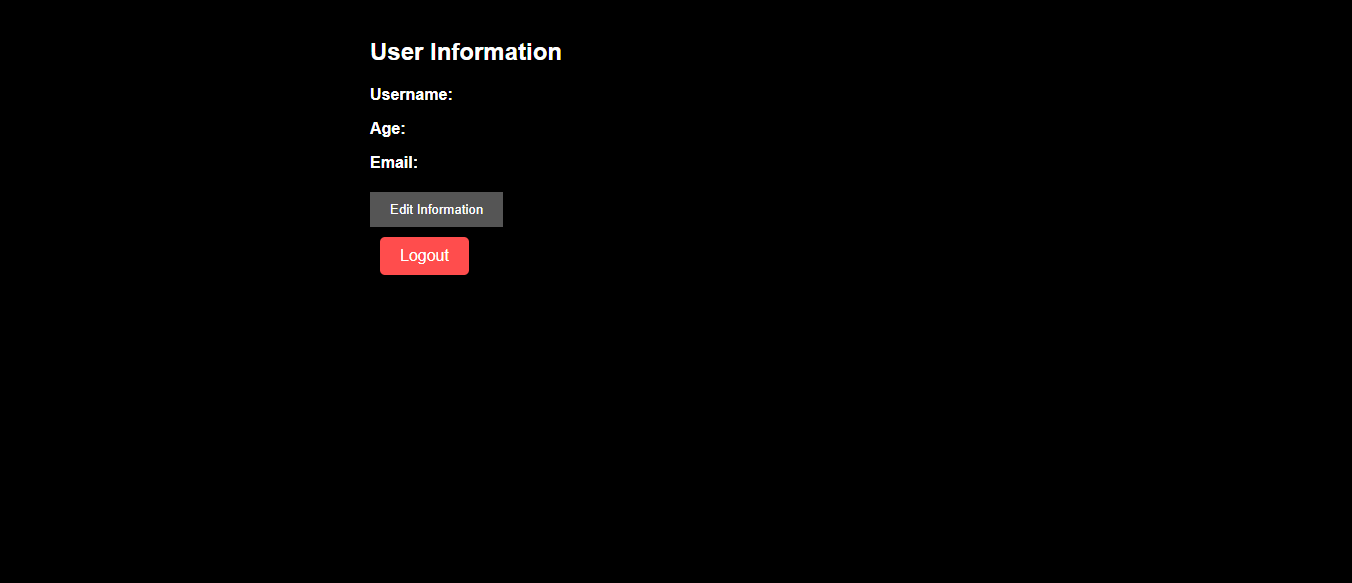
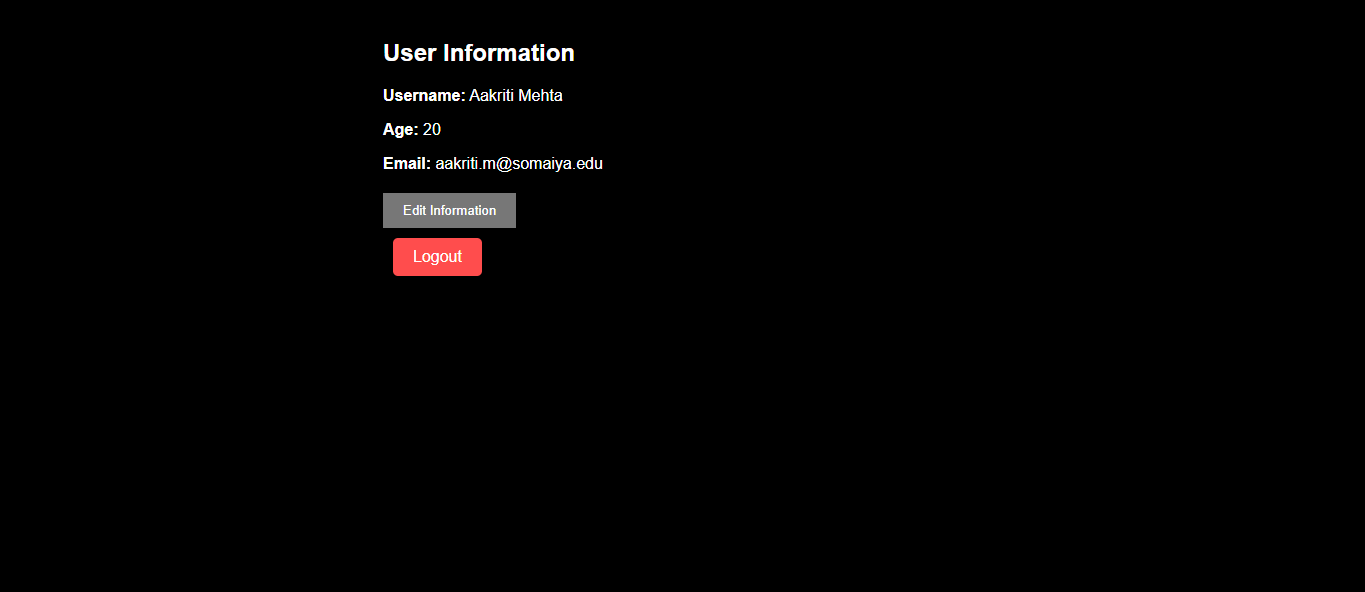
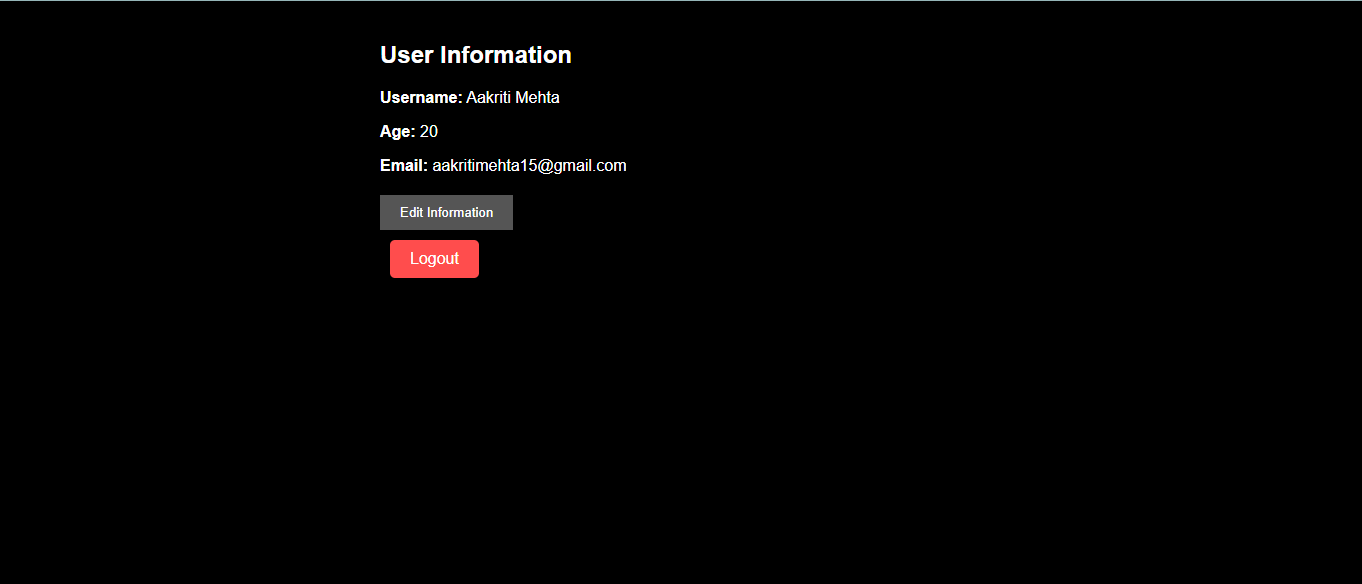
Node.js is a powerful, open-source runtime environment that enables JavaScript to be used for server-side development. Built on Chrome's V8 JavaScript engine, Node.js allows developers to build scalable and high-performance applications using JavaScript. Its non-blocking, event-driven architecture makes it particularly well-suited for I/O-intensive tasks, such as handling multiple simultaneous connections. Node.js comes with a rich ecosystem of libraries and frameworks accessible through npm (Node Package Manager), which simplifies the development of web servers, real-time applications, and other networked software. Its single-threaded nature, combined with asynchronous processing, offers efficient performance and scalability, making Node.js a popular choice for modern web development.

**Implementation Details:   
Task: User Profile Editor**

**1. Create a form with fields for the user's name, age, and email.**

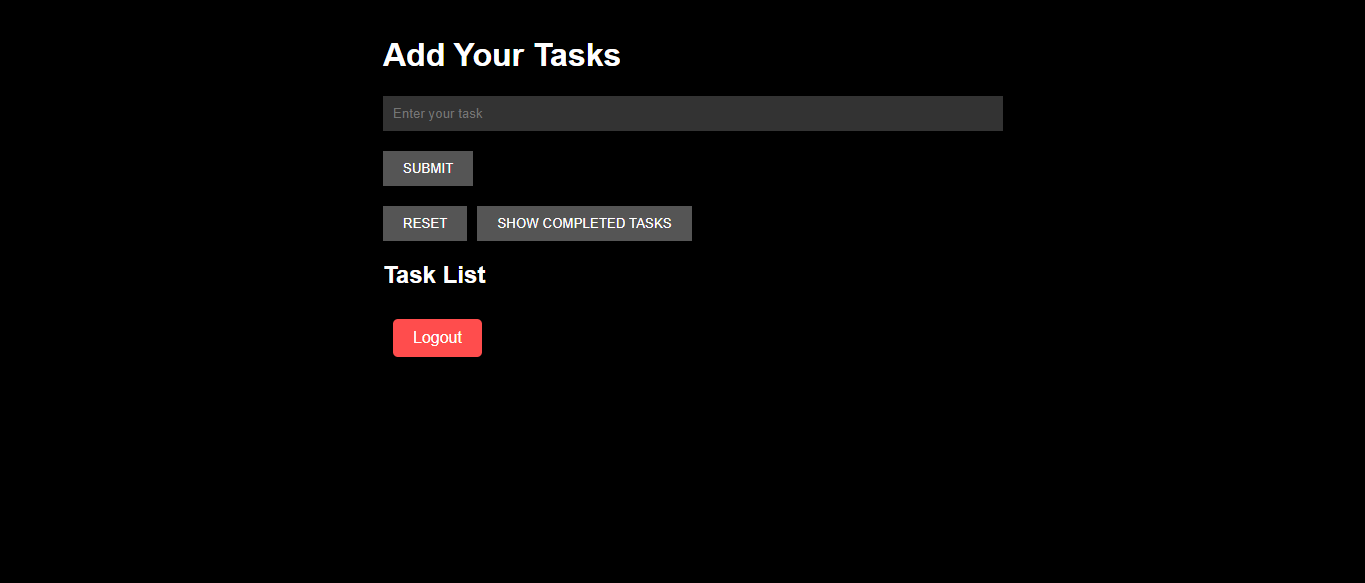
**2. Display the entered information below the form.**

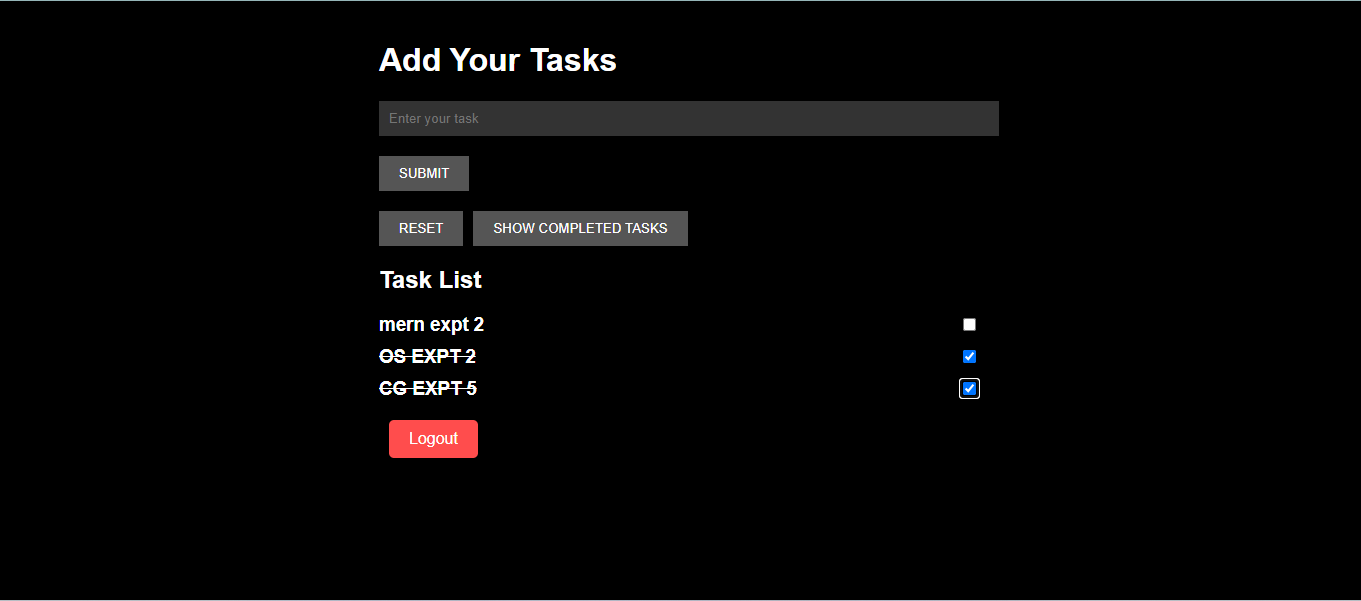
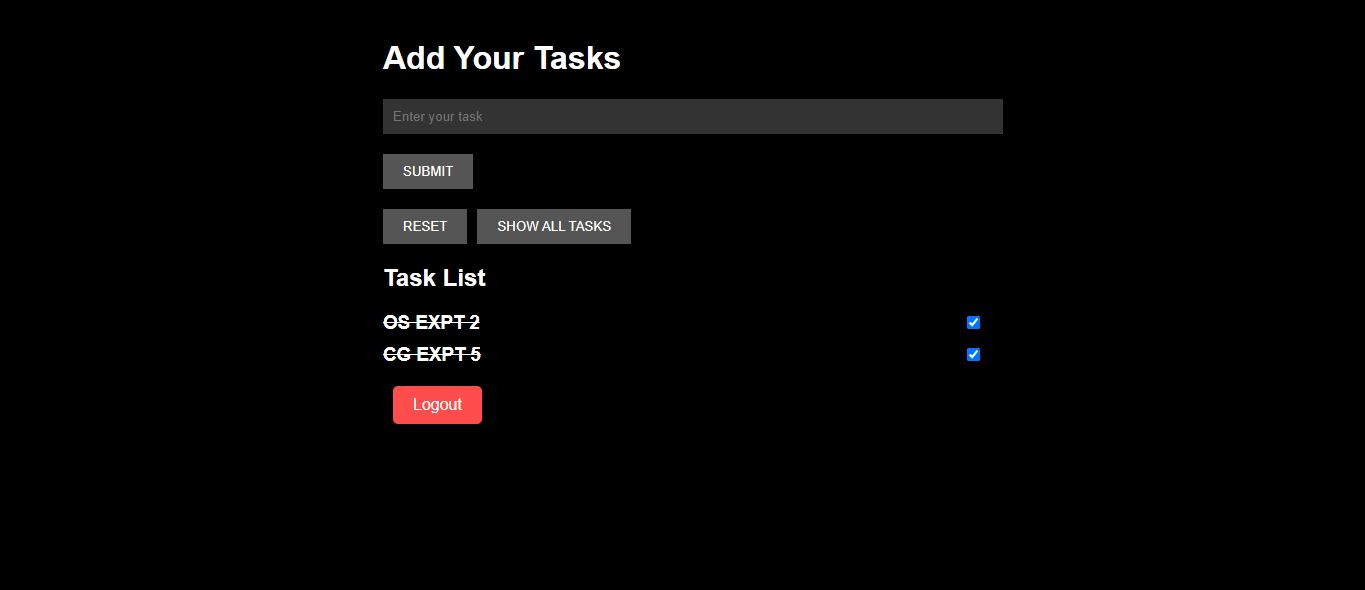
**3. Add a button to toggle the visibility of the form.**

**4. Add a button to reset the form fields to their initial values.**  
  
  
  
HIDE FORM  
  
  
  


### Task: Task Manager

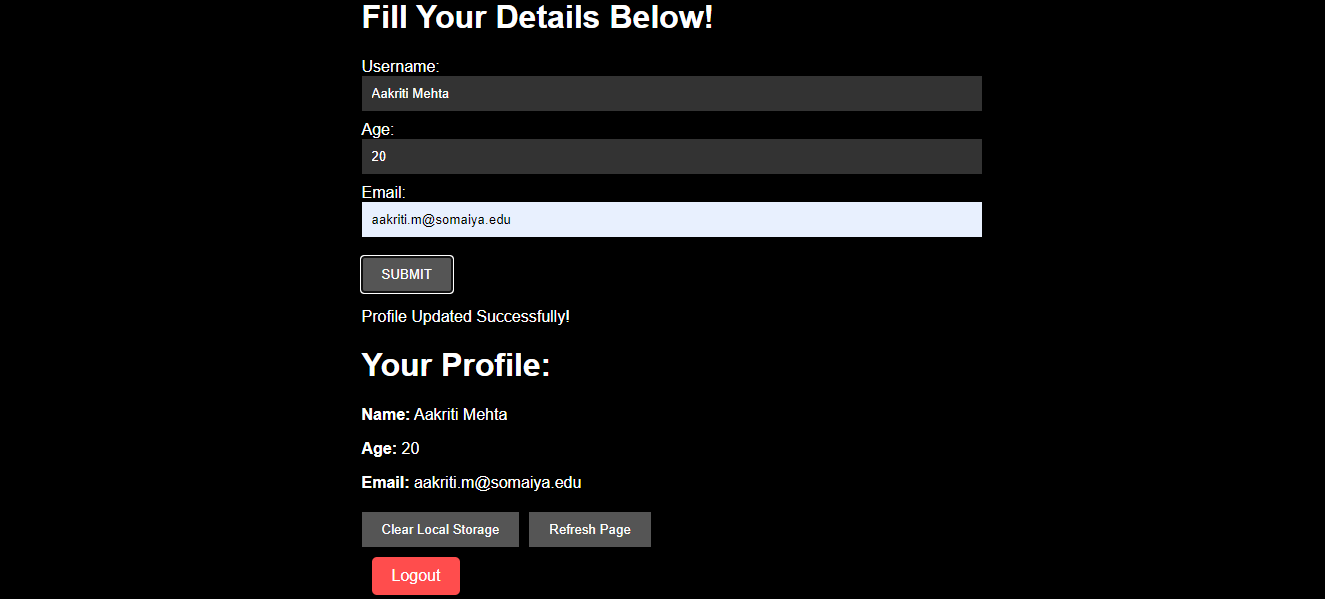
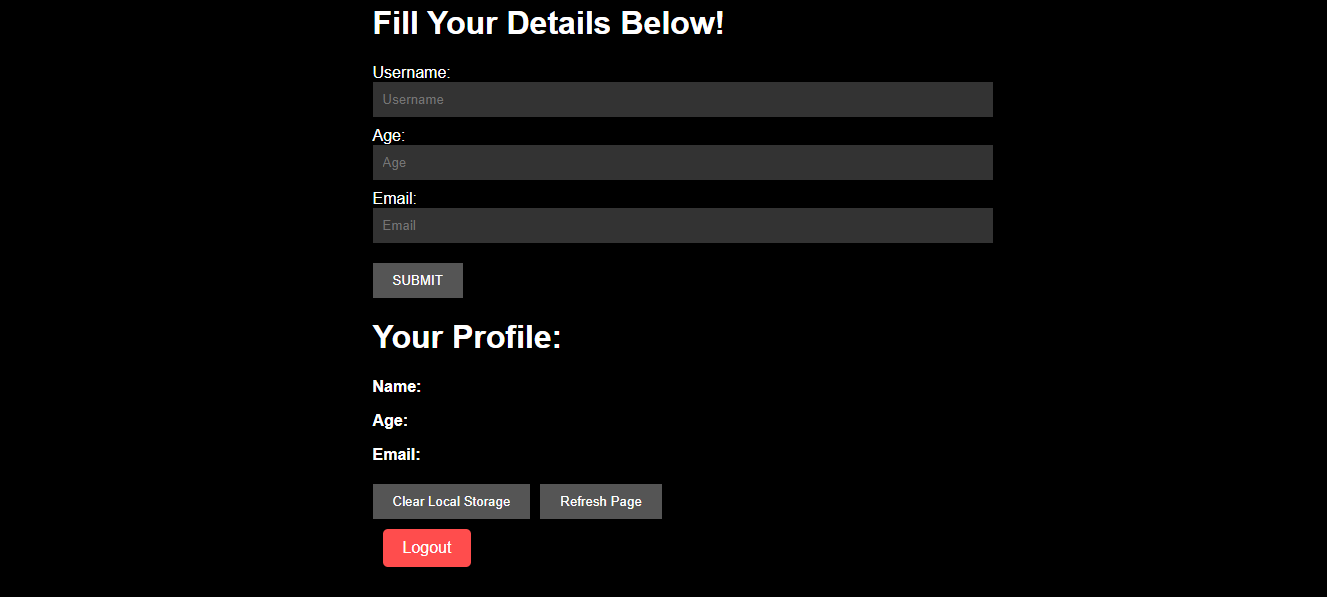
1. **Create a task manager with the following features:**
   * **An input field to add a new task.**
   * **A list to display added tasks.**
   * **A button to mark tasks as complete.**
   * **A button to toggle the visibility of completed tasks.**
   * **A button to reset the task list.**

****

****

### Task: User Profile Manager

1. **Create a form with fields for the user's name, age, and email.**
2. **Fetch initial profile data when the component mounts.**
3. **Display the fetched profile data in the form fields.**
4. **Allow the user to update their profile information.**
5. **Display the updated profile information below the form.**
6. **Log a message to the console whenever the profile data is updated.**

**  
  
**

**Conclusion:**

Learned basic react commands to build MERN application.