TASK-15

15. Use State to Toggle an Element:

– Add a toggle button to mark tasks as completed or not.

Step 1: Update TodoApp.jsx to Include Task Completion State:

```
// /src/components/TodoApp.jsx

import React, { Component } from 'react';
import '../styles/TodoApp.css'; // Import the CSS file for styling

class TodoApp extends Component {
  constructor(props) {
    super(props);
    this.state = {
      tasks: [], // List of tasks
      currentTask: '', // Current task input value
    };

    // Bind the methods to 'this'
    this.addTask = this.addTask.bind(this);
    this.removeTask = this.removeTask.bind(this);
    this.toggleCompletion = this.toggleCompletion.bind(this); // For toggling completion status
    this.handleInputChange = this.handleInputChange.bind(this);
  }

  // Method to handle input changes
  handleInputChange(event) {
    this.setState({ currentTask: event.target.value });
  }

  // Method to add a task
  addTask() {
    if (this.state.currentTask.trim() !== '') {
      this.setState((prevState) => ({
        tasks: [
          ...prevState.tasks,
          { name: prevState.currentTask, isCompleted: false },
        ],
        currentTask: '', // Reset the input field
      }));
    }
  }
```

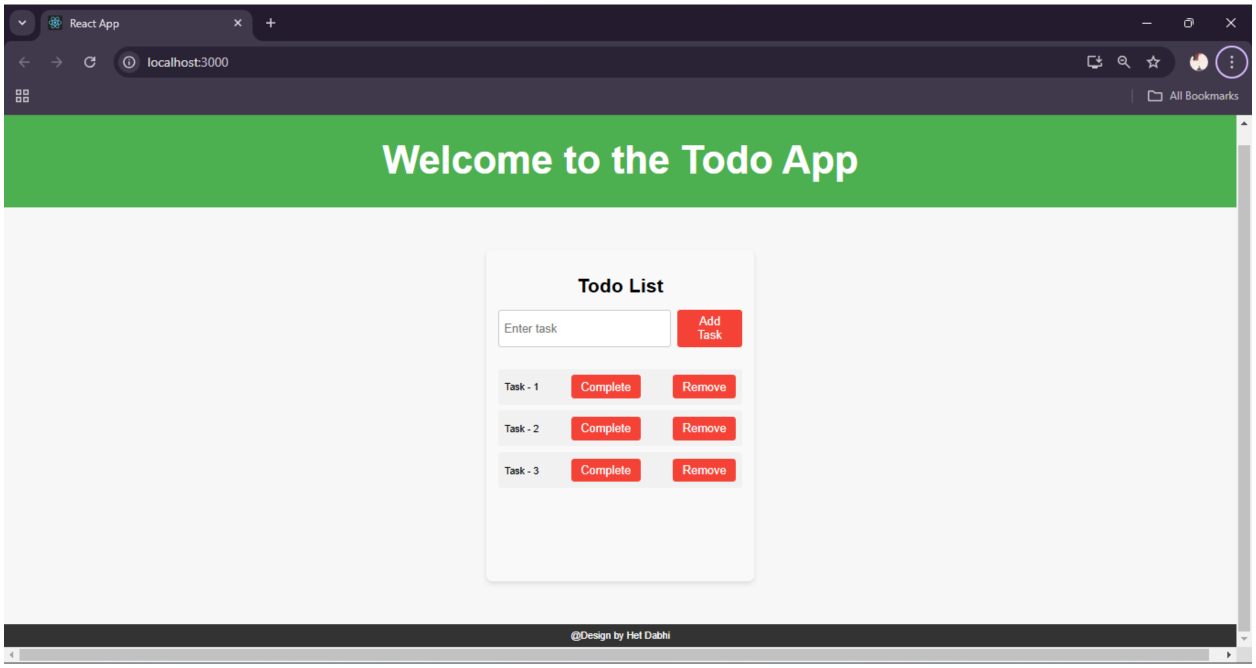
```
// Method to remove a task by index
removeTask(taskIndex) {
  this.setState((prevState) => ({
    tasks: prevState.tasks.filter((_, index) => index !== taskIndex),
  }));
}

// Method to toggle completion of a task
toggleCompletion(taskIndex) {
  this.setState((prevState) => {
    const updatedTasks = [...prevState.tasks];
    updatedTasks[taskIndex].isCompleted = !updatedTasks[taskIndex].isCompleted;
    return { tasks: updatedTasks };
  });
}

render() {
  return (
    <div className="todo-app">
      <h1>Todo List</h1>

      {/* Input field for adding tasks */}
      <div className="input-section">
        <input
          type="text"
          value={this.state.currentTask}
          onChange={this.handleInputChange}
          placeholder="Enter task"
        />
        <button onClick={this.addTask}>Add Task</button>
      </div>

      {/* Task list display */}
      <div className="task-list">
        <ul>
          {this.state.tasks.map((task, index) => (
            <li key={index} className={task.isCompleted ? 'completed' : ''}>
              {task.name}
              <button onClick={() => this.toggleCompletion(index)}>
                {task.isCompleted ? 'Undo' : 'Complete'}
              </button>
              <button onClick={() => this.removeTask(index)}>Remove</button>
            </li>
          ))}
        </ul>
      </div>
    </div>
   );
  }
}

export default TodoApp;
```

Step 2: Style Completed Tasks: (Add CSS in TodoApp.css)

```
/* /src/styles/TodoApp.css */

.todo-app {
  display: flex;
  flex-direction: column;
  align-items: center;
  width: 100%;
  max-width: 400px;
  margin: 50px auto;
  padding: 20px;
  background-color: #f9f9f9;
  border-radius: 10px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}

h1 {
  font-size: 2em;
  margin-bottom: 20px;
}

.input-section {
  display: flex;
  justify-content: space-between;
  width: 100%;
}

input[type='text'] {
  width: 70%;
  padding: 8px;
  font-size: 1.2em;
  margin-right: 10px;
  border: 1px solid #ccc;
  border-radius: 5px;
}
```

```
button {
  padding: 8px 16px;
  font-size: 1.2em;
  cursor: pointer;
  background-color: #4caf50;
  color: white;
  border: none;
  border-radius: 5px;
}

button:hover {
  background-color: #45a049;
}

.task-list {
  width: 100%;
  margin-top: 20px;
}

ul {
  list-style-type: none;
  padding: 0;
}

li {
  display: flex;
  justify-content: space-between;
  align-items: center;
  background-color: #f1f1f1;
  padding: 10px;
  margin-bottom: 10px;
  border-radius: 5px;
}

li.completed {
  text-decoration: line-through;
  background-color: #e0e0e0;
}

button {
  background-color: #f44336;
  color: white;
}

button:hover {
  background-color: #e53935;
}
```

Output:


React App
localhost:3000
All Bookmarks
Welcome to the Todo App
Todo List
Enter task
Add Task
Task - 1
Complete
Remove
Task - 2
Complete
Remove
Task - 3
Complete
Remove
@Design by Het Dabhi