**Learner Assignment Submission Format**

**Learner Details**

* **Name:Nikhil k**
* **Enrollment Number: su625mr004**
* **Batch / Class: Mern Stack**
* **Assignment: todo list**
* **Date of Submission: 04-08-25**

**Problem Solving Activity 1.1**

**1. Program Statement**

To develop a basic React application that allows users to manage a to-do list. The application should let users add new tasks and delete existing ones with a simple and clean interface.

**2. Algorithm**

 Start the React application.

 Initialize a task list with some default tasks using useState.

 Display the list on the screen using map().

 Allow the user to type a new task in the input field.

 On clicking the "Add" button:

* Add the new task to the task list.
* Clear the input field.

 On clicking the "Delete" button next to a task:

* Remove that task from the task list using filter().

 Repeat steps 4–6 as the user interacts.

**3. Pseudocode**

START

SET initialTasks = ["Read a book", "Do homework", "Take a walk", "Drink water"]

SET newTask = ""

FUNCTION handleInput(event):

SET newTask = event.value

FUNCTION addTask():

IF newTask is not empty:

APPEND newTask to tasks

CLEAR newTask

FUNCTION deleteTask(index):

REMOVE task at given index from tasks

DISPLAY tasks in an ordered list

FOR each task:

DISPLAY task text and Delete button

END

**4. Program Code**

import React, { useState } from 'react';

function ToDoList() {

const [tasks, setTasks] = useState([

"Read a book",

"Do homework",

"Take a walk",

"Drink water"

]);

const [newTask, setNewTask] = useState("");

function handleInput(event) {

setNewTask(event.target.value);

}

function addTask() {

if (newTask !== "") {

setTasks([...tasks, newTask]);

setNewTask("");

}

}

function deleteTask(index) {

const updated = tasks.filter((task, i) => i !== index);

setTasks(updated);

}

return (

<>

<style>{`

.box {

max-width: 400px;

margin: 50px auto;

padding: 20px;

background: #fff8dc;

border-radius: 10px;

text-align: center;

font-family: Arial, sans-serif;

}

h1 {

margin-bottom: 20px;

}

input {

padding: 8px;

width: 60%;

border: 1px solid #ccc;

border-radius: 5px;

}

button {

padding: 8px 12px;

margin-left: 10px;

border: none;

background-color: green;

color: white;

border-radius: 5px;

cursor: pointer;

}

ol {

margin-top: 20px;

text-align: left;

padding-left: 20px;

}

li {

margin-bottom: 10px;

}

.delete-btn {

margin-left: 10px;

background-color: red;

}

`}</style>

<div className="box">

<h1>To Do List</h1>

<input type="text" value={newTask} onChange={handleInput} placeholder="New task" />

<button onClick={addTask}>Add</button>

<ol>

{tasks.map((task, index) => (

<li key={index}>

{task}

<button className="delete-btn" onClick={() => deleteTask(index)}>Delete</button>

</li>

))}

</ol>

</div>

</>

);

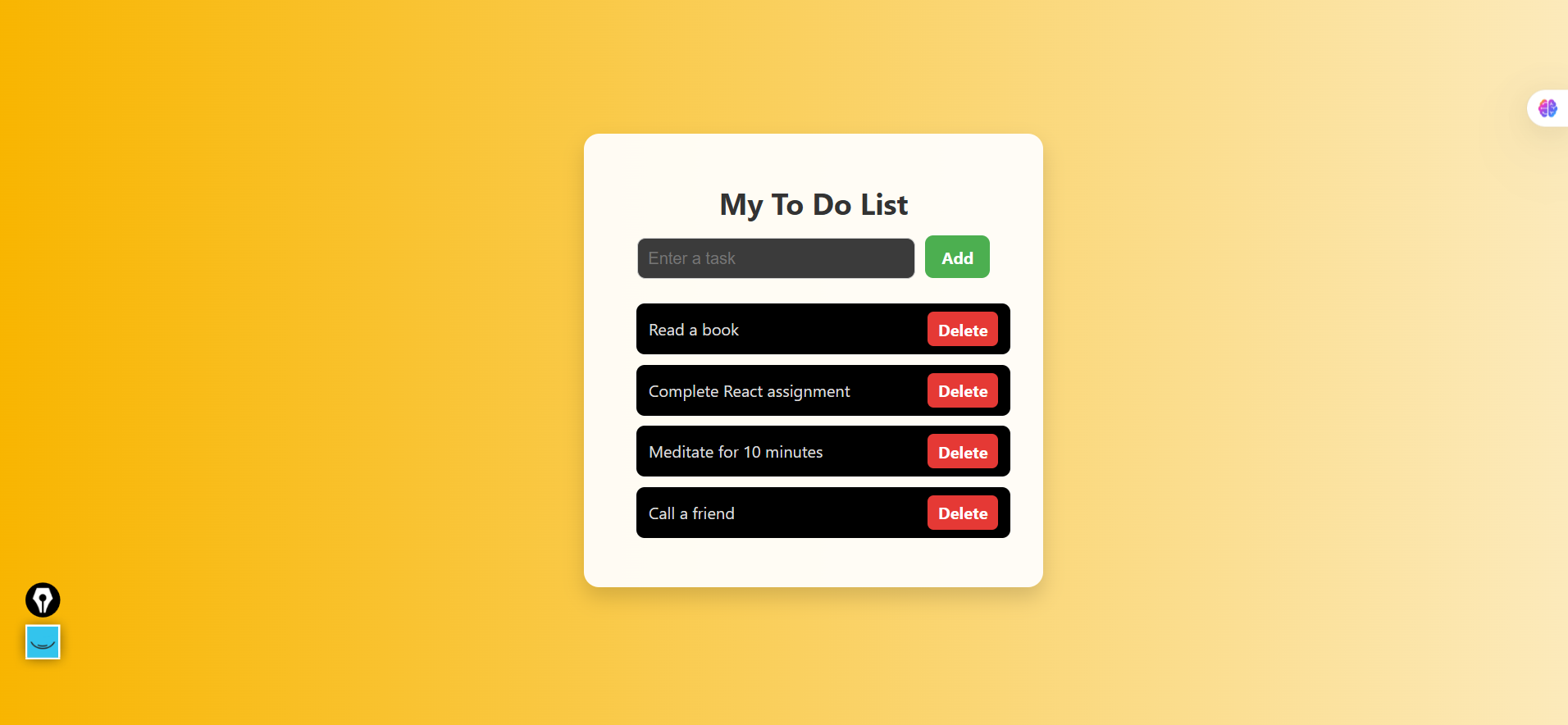
}

export default ToDoList;

**5. Test Cases**

| **Input Task** | **Expected Result** | **Actual Result** |
| --- | --- | --- |
| "Wash dishes" | Task added to the end of the list | Works |
| Empty input | No task added | Works |
| Delete index 1 | Task at position 1 removed | Works |
| Multiple additions | All tasks appear in the correct order | Works |

**6. Screenshots of Output**

****

**7. Observation / Reflection**

* Faced difficulty understanding useState in the beginning.
* Learned how to handle input, update state, and render lists in React.
* Would like to improve the design and also explore saving tasks in local storage next time.