**WEEK 06**

**Hands-on in this document**

**Project Title: React Task Manager**

**Project Structure:**

react-task-manager/

├── public/

├── src/

│ ├── components/

│ │ ├── TaskList.js

│ │ ├── TaskForm.js

│ │ └── Navbar.js

│ ├── pages/

│ │ ├── Home.js

│ │ └── About.js

│ ├── App.js

│ ├── index.js

│ └── App.css

├── package.json

**Initialize App**

npx create-react-app react-task-manager

cd react-task-manager

npm install react-router-dom

**src/index.js**

import React from "react";

import ReactDOM from "react-dom/client";

import App from "./App";

import "./App.css";

const root = ReactDOM.createRoot(document.getElementById("root"));

root.render(<App />);

**src/App.js**

import React from "react";

import { BrowserRouter as Router, Routes, Route } from "react-router-dom";

import Navbar from "./components/Navbar";

import Home from "./pages/Home";

import About from "./pages/About";

function App() {

return (

<Router>

<Navbar />

<div className="container">

<Routes>

<Route path="/" element={<Home />} />

<Route path="/about" element={<About />} />

</Routes>

</div>

</Router>

);}

export default App;

**src/components/Navbar.js**

import React from "react";

import { Link } from "react-router-dom";

function Navbar() {

return (

<nav className="navbar">

<h2>Task Manager</h2>

<div>

<Link to="/">Home</Link> | <Link to="/about">About</Link>

</div>

</nav>

);

}

export default Navbar;

**src/pages/Home.js**

import React, { useState, useEffect } from "react";

import TaskForm from "../components/TaskForm";

import TaskList from "../components/TaskList";

function Home() {

const [tasks, setTasks] = useState([]);

useEffect(() => {

fetch("https://jsonplaceholder.typicode.com/todos?\_limit=5")

.then((res) => res.json())

.then((data) => {

const cleaned = data.map((item) => ({

id: item.id,

title: item.title,

completed: item.completed,

}));

setTasks(cleaned);

});

}, []);

const addTask = (title) => {

const newTask = {

id: Date.now(),

title,

completed: false,

};

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

};

const toggleTask = (id) => {

setTasks(

tasks.map((task) =>

task.id === id ? { ...task, completed: !task.completed } : task

) );};

const deleteTask = (id) => {

setTasks(tasks.filter((task) => task.id !== id));

};

return (

<div>

<TaskForm addTask={addTask} />

<TaskList tasks={tasks} toggleTask={toggleTask} deleteTask={deleteTask} />

</div>

);}

export default Home;

**src/components/TaskForm.js**

import React, { useState } from "react";

function TaskForm({ addTask }) {

const [input, setInput] = useState("");

const handleSubmit = (e) => {

e.preventDefault();

if (!input.trim()) return;

addTask(input);

setInput("");

};

return (

<form onSubmit={handleSubmit}>

<input

type="text"

placeholder="Enter task..."

value={input}

onChange={(e) => setInput(e.target.value)}

/>

<button type="submit">Add</button>

</form>

);

}

export default TaskForm;

**src/components/TaskList.js**

import React from "react";

function TaskList({ tasks, toggleTask, deleteTask }) {

if (tasks.length === 0) return <p>No tasks yet!</p>;

return (

<ul>

{tasks.map((task) => (

<li key={task.id} style={{ marginBottom: "8px" }}>

<input

type="checkbox"

checked={task.completed}

onChange={() => toggleTask(task.id)}

/>

<span

style={{

textDecoration: task.completed ? "line-through" : "none",

marginLeft: "10px",

}}

>

{task.title}

</span>

<button

onClick={() => deleteTask(task.id)}

style={{ marginLeft: "10px", color: "red" }}

>

Delete

</button>

</li>

))}

</ul>

);}

export default TaskList;

**src/pages/About.js**

import React from "react";

function About() {

return (

<div>

<h2>About This App</h2>

<p>This is a basic React task manager to practice hands-on concepts.</p>

</div>

);

}

export default About;

**src/App.css**

body {

font-family: Arial, sans-serif;

background-color: #f3f3f3;

margin: 0;

padding: 0;

}

.navbar {

background-color: #333;

padding: 1rem;

color: white;

display: flex;

justify-content: space-between;

}

.navbar a {

color: white;

margin-left: 1rem;

text-decoration: none;

}

.container {

padding: 2rem;

}

input[type="text"] {

padding: 0.5rem;

margin-right: 1rem;

width: 200px;

}

button {

padding: 0.5rem 1rem;

}

**To Run the App:**

npm start