

DYNAMIC TASK MANAGER

TaskForm.jsx

```
import React, { useState } from 'react';
function TaskForm({ addTask }) {
  const [taskText, setTaskText] = useState("");
  const handleInputChange = (e) => {
    setTaskText(e.target.value);
  };
  const handleSubmit = (e) => {
    e.preventDefault();
    if (taskText.trim()) {
      addTask({ id: Date.now(), text: taskText, completed: false });
      setTaskText("");
    }
  };
  return (
    <form onSubmit={handleSubmit}>
      <input
        type="text"
        placeholder="Add a new task"
        value={taskText}
        onChange={handleInputChange}
      />
      <button type="submit">Add Task</button>
    </form>
  );
}
export default TaskForm;
```

TaskItem.jsx

```
import React from 'react';
function TaskItem({ task, removeTask, toggleComplete }) {
  return (
    <li>
      <input
        type="checkbox"
        checked={task.completed}
        onChange={() => toggleComplete(task.id)}
      />
      <span className={task.completed ? 'completed' : ''}>{task.text}</span>
      <button onClick={() => removeTask(task.id)}>Remove</button>
    </li>
  );
}
```

```
}  
export default TaskItem;
```

TaskList.jsx

```
import React from 'react';  
import TaskItem from './TaskItem';  
function TaskList({ tasks, removeTask, toggleComplete }) {  
  return (  
    <ul>  
      {tasks.map((task) => (  
        <TaskItem  
          key={task.id}  
          task={task}  
          removeTask={removeTask}  
          toggleComplete={toggleComplete}  
        />  
      ))}  
    </ul>  
  );  
}  
export default TaskList;
```

App.js

```
import React, { useState } from 'react';  
import TaskForm from './TaskManager/TaskForm';  
import TaskList from './TaskManager/TaskList';  
import './App.css';  
function App() {  
  const [tasks, setTasks] = useState([]);  
  const addTask = (task) => {  
    setTasks([...tasks, task]);  
  };  
  const removeTask = (taskId) => {  
    const updatedTasks = tasks.filter((task) => task.id !== taskId);  
    setTasks(updatedTasks);  
  };  
  const toggleComplete = (taskId) => {  
    const updatedTasks = tasks.map((task) =>  
      task.id === taskId ? { ...task, completed: !task.completed } : task  
    );  
    setTasks(updatedTasks);  
  };  
  return (  

```

```

<div className="App">
  <h1>Dynamic Task Manager</h1>
  <TaskForm addTask={addTask} />
  <TaskList
    tasks={tasks}
    removeTask={removeTask}
    toggleComplete={toggleComplete}
  />
</div>
);
}
export default App;

```

OUTPUT:

