| Ex. No: 5 | |
|---|---|
| | **React based App Development** |
| **23.02.2023** | |

**Aim:**

To Create an App using React with Components, Rendering, and Data Sharing.

**Algorithm:**
1. Create a new React project using a tool like Create React App.
2. Build individual components to represent various parts of your app.
3. Arrange components hierarchically, defining parent-child relationships.
4. Define the UI using JSX within components for rendering.
5. Pass data between components using props or consider state management for more complex data sharing.
6. Implement local or global state management for dynamic data and user interactions.
7. Style components using CSS, CSS modules, or CSS-in-JS libraries while keeping styling separate from logic.

**Program:**

```
import React, { useState } from 'react';
import './App.css';
import TodoList from './components/TodoList';

function App() {
  const [todos, setTodos] = useState([]);
  const [text, setText] = useState('');

  const addTodo = () => {
    if (text.trim() !== '') {
      const newTodo = { id: Date.now(), text, completed: false };
      setTodos([...todos, newTodo]);
      setText('');
    }
  };

  const toggleComplete = (id) => {
    const updatedTodos = todos.map((todo) =>
      todo.id === id ? { ...todo, completed: !todo.completed } : todo
    );
    setTodos(updatedTodos);
  };

  const deleteTodo = (id) => {
    const updatedTodos = todos.filter((todo) => todo.id !== id);
    setTodos(updatedTodos);
  };

  return (
    <div className="App">
```

```jsx
      <h1>Todo List App</h1>
      <input
        type="text"
        value={text}
        onChange={(e) => setText(e.target.value)}
        placeholder="Enter a new task"
      />
      <button onClick={addTodo}>Add</button>
      <TodoList
        todos={todos}
        toggleComplete={toggleComplete}
        deleteTodo={deleteTodo}
      />
    </div>
  );
}
export default App;
```

**Todoitem.js Component**

```jsx
import React from 'react';

const TodoItem = ({ todo, toggleComplete, deleteTodo }) => {
  return (
    <div className={`todo-item ${todo.completed ? 'completed' : ''}`}>
      <input
        type="checkbox"
        checked={todo.completed}
        onChange={() => toggleComplete(todo.id)}
      />
      <p>{todo.text}</p>
      <button onClick={() => deleteTodo(todo.id)}>Delete</button>
    </div>
  );
};

export default TodoItem;
```

**TodoList.js Component**
```jsx
import React from 'react';
import TodoItem from './TodoItem';

const TodoList = ({ todos, toggleComplete, deleteTodo }) => {
  return (
    <div className="todo-list">
      {todos.map((todo) => (
        <TodoItem
          key={todo.id}
          todo={todo}
          toggleComplete={toggleComplete}
          deleteTodo={deleteTodo}
        />
      ))}
```
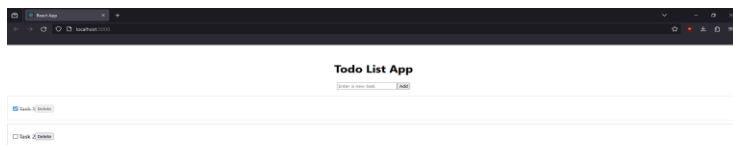
```
      </div>
   );
};

export default TodoList;
```
**Output:**

```
Github Link:
```

https://github.com/kavin-t28/CS3809-Web-Technologies-Lab





**Result:**
Therefore, we've successfully created a simple react app.