Practical 3:- Create a ToDo App using ReactJS.

App.js

```
import { useState } from 'react'
import reactLogo from './assets/react.svg'
import viteLogo from '/vite.svg'
import TodoList from './TodoList'
import TodoItem from './TodoItem'
import './App.css'

export default function App() {
  return (
    <>
    <div className='App'>
        <TodoList/>
    </div>

    </>
  );
}
```

TodoItem.js

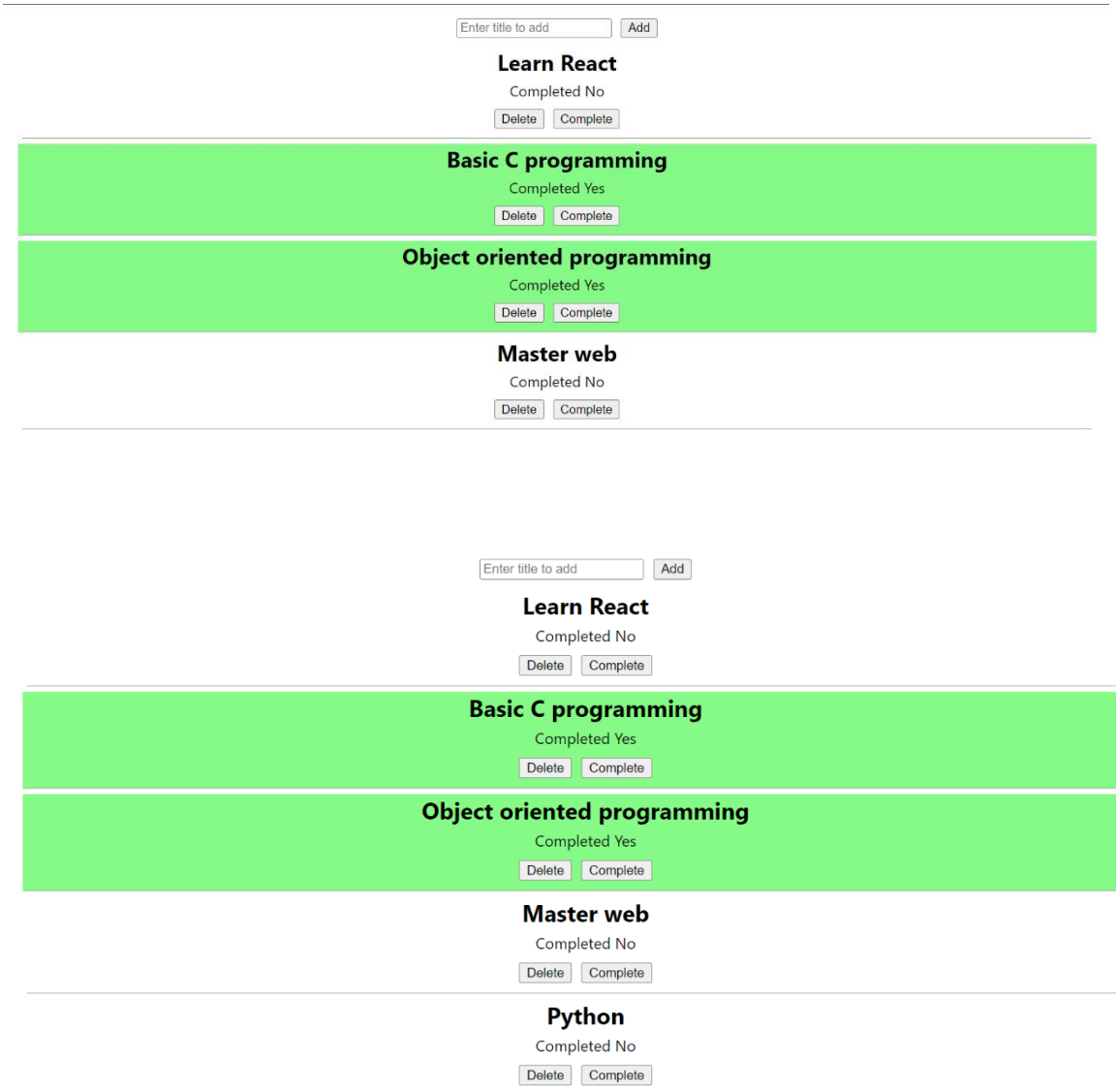
```
import './App.css';
export default function TodoItem(props)
{
     const completedStyle={
          color: props.task.completed ? 'green' : 'red',
     };
     const deleteButtonStyle = {
          backgroundColor: 'red',
          color: 'white',
          marginRight: '5px',
     };
     const completeStyle={
          backgroundColor:'#00cc33',
          color :'White',
          marginRight:'5px',
     }
     return(
          <div className="TodoItem">
               <h2>{props.task.title}</h2>
               <p style={completedStyle}>Completed
                         {props.task.completed?"Yes":"No"}</p>
               <button onClick={props.onDelete}
                    style={deleteButtonStyle}>Delete</button>  
               <button onClick={props.onComplete}
                     style={completeStyle}>Complete</button>
          </div>
     );
}
```

TodoList.js

```
import TodoItem from "./TodoItem";
import './App.css';
import { useState } from "react";
export default function TodoList() {
     const task = [ { id: 1, title: "Learn React", completed: false }, ]
     const [todo, setTodo] = useState(task);
     const [inputValue, setInputValue] = useState("");

     const deleteTodo = (keyId) => {
          const updateTodo = todo.filter(answer => answer.id !== keyId);
          setTodo(updateTodo);
     }
     const Complete = (keyId) => {
          const newTodo = todo.map(val => {
               if (val.id == keyId) {
                return { ...val, completed: !val.completed }}
               return val;
          })
          setTodo(newTodo);
     }
     const addTodo = () => {
          const newobj = {
               id: Date.now(),
               title: inputValue,
               completed: false
          }
          setTodo([...todo, newobj]);
          setInputValue("");
     }
     return ( <div className="TodoList">
               <h1><u>TODO APP</u></h1>
               {todo.map(row => (
                    <TodoItem key={row.id}
                              task={row}
                              onDelete={() => deleteTodo(row.id)}
                              onComplete={() => Complete(row.id)} />
                    ))}
               <br/>
               <div className="inputContainer">
                    <input
                         type="text"
                         className="TodoInput"
                         value={inputValue}
                         onChange={(e)=> setInputValue(e.target.value)}
                         placeholder="Please enter your tasks"
                    /><br/><br/>
                    <button onClick={addTodo} className="addTodoButton">
                         Add New Button
                    </button>
               </div>
          </div>
     );}
```

Output:-


Enter title to add
Add
Learn React
Completed No
Delete
Complete
Basic C programming
Completed Yes
Delete
Complete
Object oriented programming
Completed Yes
Delete
Complete
Master web
Completed No
Delete
Complete
Enter title to add
Add
Learn React
Completed No
Delete
Complete
Basic C programming
Completed Yes
Delete
Complete
Object oriented programming
Completed Yes
Delete
Complete
Master web
Completed No
Delete
Complete
Python
Completed No
Delete
Complete