**Explain React State**  
React state is an object managed within a component to store and track dynamic data. When state changes, the component re-renders automatically.

**Step-by-Step Implementation**

### ****Step 1: Create a React App****

npx create-react-app counterappcd counterapp

**Step 2: Create** CountPeople **Class Component**

Create a file CountPeople.js in the src folder.

// src/CountPeople.jsimport React from "react";

class CountPeople extends React.Component {

constructor(props) {

super(props);

this.state = {

entryCount: 0,

exitCount: 0,

};

}

UpdateEntry = () => {

this.setState((prevState) => ({

entryCount: prevState.entryCount + 1,

}));

};

UpdateExit = () => {

this.setState((prevState) => ({

exitCount: prevState.exitCount + 1,

}));

};

render() {

return (

<div style={{ padding: "20px", textAlign: "center" }}>

<h1>🛍️ Mall Entry Counter</h1>

<h2>People Entered: {this.state.entryCount}</h2>

<h2>People Exited: {this.state.exitCount}</h2>

<button onClick={this.UpdateEntry} style={{ margin: "10px", padding: "10px" }}>

Login

</button>

<button onClick={this.UpdateExit} style={{ margin: "10px", padding: "10px" }}>

Exit

</button>

</div>

);

}

}

export default CountPeople;

**Step 3: Render in** index.js

// src/index.jsimport React from "react";import ReactDOM from "react-dom";import CountPeople from "./CountPeople";

ReactDOM.render(

<React.StrictMode>

<CountPeople />

</React.StrictMode>,

document.getElementById("root")

);

**OUTPUT:**

