# 8. ReactJS-HOL

# Objective

• Explain React State

**React State**

In React, **state** is a built-in object used to contain data or information about the component that can change over time. When the state of a component changes, React **automatically re-renders** the component and updates the DOM.

**🧠 Key Points about React State**

1. **State is Mutable**  
   Unlike props (which are immutable and passed from parent to child), state can be modified using the setState() method.
2. **Used to Track Data Internally**  
   It is used for tracking things like form inputs, user interactions, toggling UI elements, etc.

**Hands-on Lab**

# Steps

## 1. Create React App

npx create-react-app counterapp  
cd counterapp

## 2. Create src/CountPeople.js

import React, { Component } from "react";  
  
class CountPeople extends Component {  
 constructor() {  
 super();  
 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={{ display: "flex", justifyContent: "space-around", marginTop: "100px" }}>  
 <div>  
 <button onClick={this.updateEntry}>Login</button>  
 <h3>{this.state.entrycount} People Entered!!!</h3>  
 </div>  
 <div>  
 <button onClick={this.updateExit}>Exit</button>  
 <h3>{this.state.exitcount} People Left!!!</h3>  
 </div>  
 </div>  
 );  
 }  
}  
  
export default CountPeople;

## 3. Update src/App.js

import React from "react";  
import CountPeople from "./CountPeople";  
  
function App() {  
 return (  
 <div className="App">  
 <CountPeople />  
 </div>  
 );  
}  
  
export default App;

## 4. Run the App

npm start

## Output

Clicking Login increases the "People Entered" count, and clicking Exit increases the "People Left" count.

