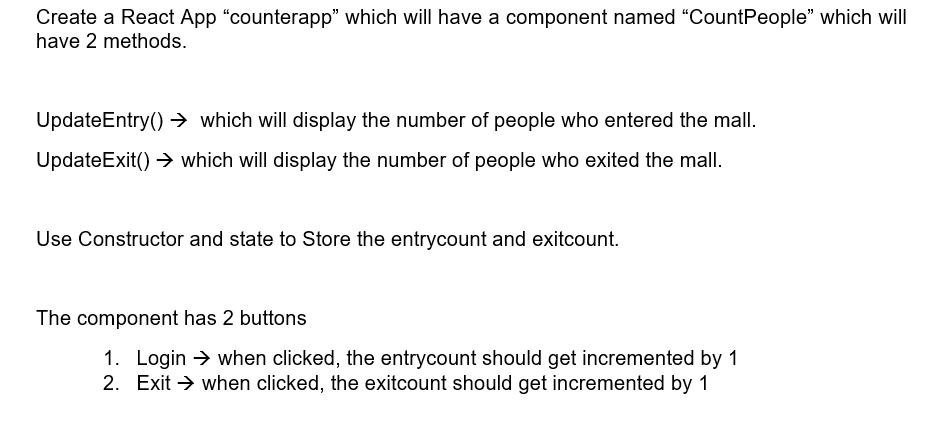
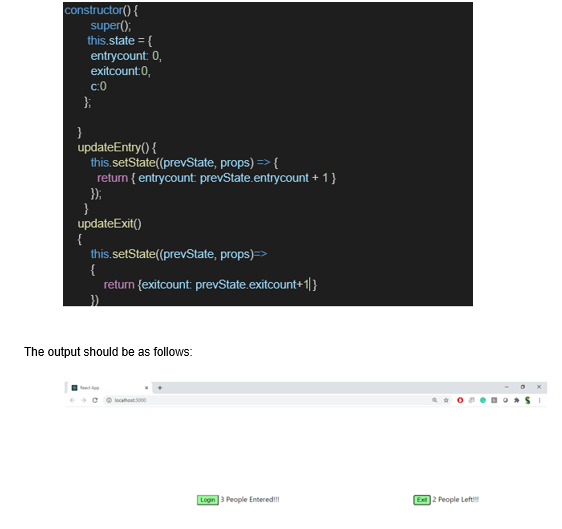


React state is a special built-in object in React components that is used to store data that may change over time. It allows a component to keep track of information between renders, such as user input, button clicks, or API responses. State is managed internally by the component and can be updated using the setState method in class components or the useState hook in functional components. When the state is updated, React automatically re-renders the component to reflect the new data in the user interface. Unlike props, which are passed from parent to child and are read-only, state is mutable and belongs to the component itself.





**Solution**

**CountPeople.js**

import React from 'react';

import './CountPeople.css';

class CountPeople extends React.Component {

constructor() {

super();

this.state = {

entrycount: 0,

exitcount: 0,

c: 0

};

}

updateEntry() {

this.setState((prevState, props) => {

return { entrycount: prevState.entrycount + 1 };

});

}

updateExit() {

this.setState((prevState, props) => {

return { exitcount: prevState.exitcount + 1 };

});

}

render() {

return (

<div className="container">

<div className="box">

<button className="btn login" onClick={() => this.updateEntry()}>Login</button>

<span className="text">{this.state.entrycount} People Entered!!!</span>

</div>

<div className="box">

<button className="btn exit" onClick={() => this.updateExit()}>Exit</button>

<span className="text">{this.state.exitcount} People Left!!!</span>

</div>

</div>

);

}

}

export default CountPeople;

**CountPeople.css**

.container {

display: flex;

justify-content: center;

align-items: center;

height: 90vh;

gap: 350px;

}

.btn {

padding: 5px 10px;

font-weight: bold;

border: 2px solid black;

border-radius: 5px;

background-color: #54fd54;

margin-right: 10px;

cursor: pointer;

}

.login {

color: rgb(0, 0, 0);

border-radius: 0px;

}

.exit {

color: rgb(0, 0, 0);

}

.text {

font-size: 16px;

}

**App.js**

import React from 'react';

import CountPeople from './CountPeople';

function App() {

return (

<div className="App">

<CountPeople />

</div>

);

}

export default App;

**Output**

