**Explain React State**

In React, **State** refers to an object that **stores dynamic data** of a component and **determines how that component behaves or appears** at any point in time.

**Key Points about State:**

* State is **mutable**, meaning it can change over time.
* When state changes, **React re-renders the component** automatically.
* State is **local to the component** unless passed via props.

Example (Class Component):

import React, { Component } from 'react';

class Counter extends Component {

constructor(props) {

super(props);

// Initializing state

this.state = {

count: 0

};

}

// Method to update state

increment = () => {

this.setState({ count: this.state.count + 1 });

};

render() {

return (

<div>

<p>Count is: {this.state.count}</p>

<button onClick={this.increment}>Increase</button>

</div>

);

}

}

export default Counter;

**Create a React App “counterapp” which will have a component named “CountPeople” which will have 2 methods.**

**Index.js :-**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import CountPeople from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<CountPeople />);

**App.js :-**

import React from 'react';

import './App.css';

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 className="main-container">

        <div className="entry-section">

          <button className="green-button" onClick={this.UpdateEntry}>Login</button>

          <span className="label">{this.state.entryCount} People Entered!!!</span>

        </div>

        <div className="exit-section">

          <button className="green-button" onClick={this.UpdateExit}>Exit</button>

          <span className="label">{this.state.exitCount} People Left!!!</span>

        </div>

      </div>

    );

  }

}

export default CountPeople;

**app.css :-**

.main-container {

  display: flex;

  justify-content: space-around;

  align-items: center;

  margin-top: 100px;

  font-family: Arial, sans-serif;

}

.entry-section,

.exit-section {

  display: flex;

  align-items: center;

  gap: 10px;

}

.green-button {

  background-color: white;

  border: 2px solid green;

  color: green;

  font-weight: bold;

  padding: 8px 16px;

  cursor: pointer;

  border-radius: 4px;

}

.green-button:hover {

  background-color: #eaffea;

}

.label {

  font-size: 18px;

  color: #333;

}

Output :-



