Create react app

npx create-react-app my-app

cd my-app Code

// App.js

import React, { useState } from 'react';

const Greeting = ({ name }) => {

return <h2>Hello, {name ? name : 'Guest'}!</h2>;

};

const App = () => {

const [name, setName] = useState('');

const handleChange = (event) => {

setName(event.target.value);

};

return (

<div>

<h1>Welcome to React!</h1>

<input

type="text"

placeholder="Enter your name"

value={name}

onChange={handleChange}

/>

<Greeting name={name} />

</div>

);

};

export default App;

In React, **state** and **props** are two fundamental concepts that are crucial for managing data and building interactive user interfaces. Here's a brief overview of both, along with examples demonstrating their use.

**State**

* **Definition**: State is a built-in object in React that is used to manage and track data that can change over time. Each component can maintain its own state, which can be updated using the setState function (for class components) or the useState hook (for functional components).
* **Purpose**: State is used to store data that a component needs to render and respond to user actions.

**Props**

* **Definition**: Props (short for properties) are read-only data that are passed from a parent component to a child component. Props allow you to pass data and event handlers to child components.
* **Purpose**: Props are used to configure child components and share data between components without directly modifying their state.