

1. A React component is a reusable piece of code that returns a portion of the user interface, like a JavaScript function or class that renders HTML elements. It is a building block that that encapsulates a piece of the user interface. It defines the structure and behavior of the UI, either by managing its own state or by receiving data through props and rendering content accordingly.

|  |  |
| --- | --- |
| React Component | JavaScript Function |
| A React component is used to build user interfaces by returning JSX. | A JavaScript function is used to perform logic or return a value. |
| It returns JSX to be rendered in the browser. | It returns any data type such as a number, string, or object. |
| It is used directly in the React app to display visual elements. | It is not used for rendering UI directly. |
| It works with React features like state, props, and lifecycle methods. | It does not interact with React-specific features unless wrapped inside a component. |

1. There are two primary types of react component which are listed below:

* Functional Component
* Class Component

1. Class Components: These are ES6 classes that extend React.Component. They include additional features like state management and lifecycle methods.

Example:

function App(props) {

return <h1>Hello, {props.name}!</h1>;

}

1. Functional Components: These are simpler and preferred for most use cases. They are JavaScript functions that return React elements. With the introduction of React Hooks, functional components can also manage state and lifecycle events.

Example:

function App(props) {

return <h1>Welcome, {props.username}!</h1>;

}

<App username="User" />;

1. In a class component, the constructor(props) is a special method where we call super(props) and can initialize state or bind methods before the component mounts.

Example:

import React, { Component } from 'react';

class Welcome extends Component {

constructor(props) {

super(props);

this.state = {

message: 'Hello, React!'

};

}

render() {

return <h1>{this.state.message}</h1>;

}

}

export default Welcome;

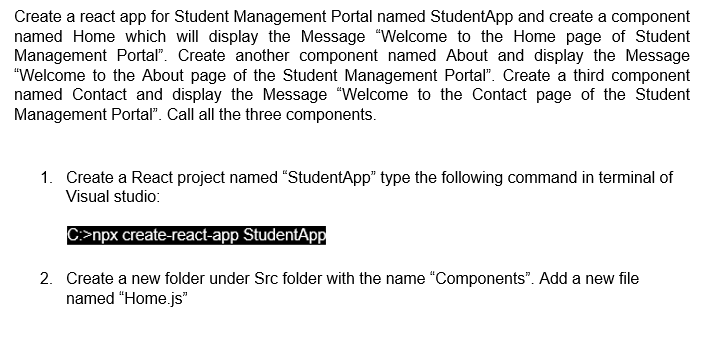
1. In a class component, render() is the method that returns the JSX for React to display; it runs every time the component’s props or state change.

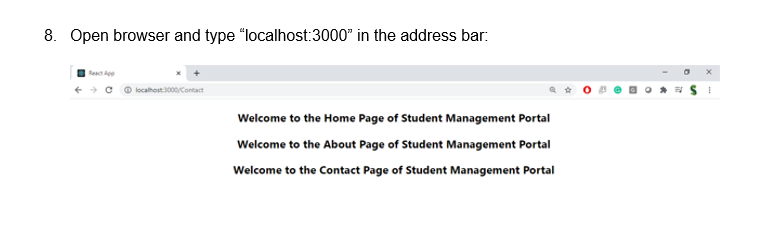
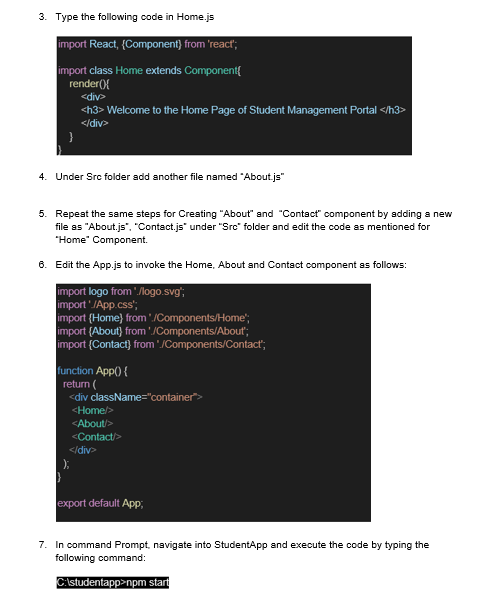
Example:

render() {

return <h1>{this.state.message}</h1>;

}





**Output**

**Home.js**

import React, { Component } from 'react';

class Home extends Component {

render() {

return (

<div>

<h3>Welcome to the Home page of Student Management Portal</h3>

</div>

);

}

}

export default Home;

**About.js**

import React, { Component } from 'react';

class About extends Component {

render() {

return (

<div>

<h3>Welcome to the About page of Student Management Portal</h3>

</div>

);

}

}

export default About;

**Contact.js**

import React, { Component } from 'react';

class Contact extends Component {

render() {

return (

<div>

<h3>Welcome to the Contact page of Student Management Portal</h3>

</div>

);

}

}

export default Contact;

**App.js**

import logo from './logo.svg';

import './App.css';

import Home from './Components/Home';

import About from './Components/About';

import Contact from './Components/Contact';

function App() {

return (

<div className="container">

<Home />

<About />

<Contact />

</div>

);

}

export default App;

**Output**

