## **Objectives**

**Explain React components**

React components are the building blocks of a React application. They are reusable pieces of code that return HTML-like structures (called JSX), which define how a selection of the UI should appear. Components help developers organize code into smaller, manageable parts. For example, a website may have components like Header, Footer, Home, and Contact, each handling its own layout and functionality.

**Identify the differences between components and JavaScript functions**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **JavaScript Function** | **React Component** |
| Purpose | Performs a task or returns a value | Returns JSX to build UI |
| Return Type | Primitive values (like number, string, etc.) | JSX (HTML-like code) |
| Usage | Called directly in code | Used inside JSX like <ComponentName /> |
| Integration with react | Not Inherently connected | Can use React features like hooks |

**Identify the types of components**

There are two main types of components in React:

1. Class Components – These are ES6 classes that extend React.Component and include a render() method.
2. Function Components – There are simple JavaScript functions that return JSX and can use React hooks for added features like state and lifecycle.

**Explain class component**

A class component is a React component defined using ES6 class. It provides more features that a function component by default, such as managing state and using lifecycle methods. Every class component must have a render() method which return JSX.

**Explain function component**

A function component is a simplest way to create a component in React. It is written using a regular JavaScript function and return JSX directly. With the introduction of hooks, functional components can now also manage state and lifecycle events, making them just as powerful as class components.

**Define component constructor**

The constructor is a special methos used inside a class component. It is called automatically when the component is created. It is commonly used to initialize state and bind members. The constructor must call super(props) before using this.

**Define render() function**

The render() function is required method in a class components. It defines what the component should display on the screen. It must return JSX (React’s HTML-like syntax). React will call this method whenever the component needs to update the UI.

**Create a react app for Student Management Portal named StudentApp and create a component named Home which will display the Message “Welcome to the Home page of Student Management Portal”. Create another component named About and display the Message “Welcome to the About page of the Student Management Portal”. Create a third component named Contact and display the Message “Welcome to the Contact page of the Student Management Portal”. Call all the three components.**

**Codes:**

**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**;

**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**;

**App.js:**

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**/>

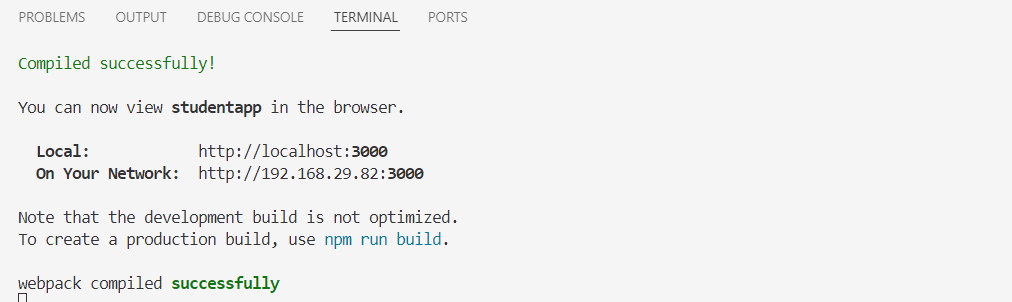
    </div>

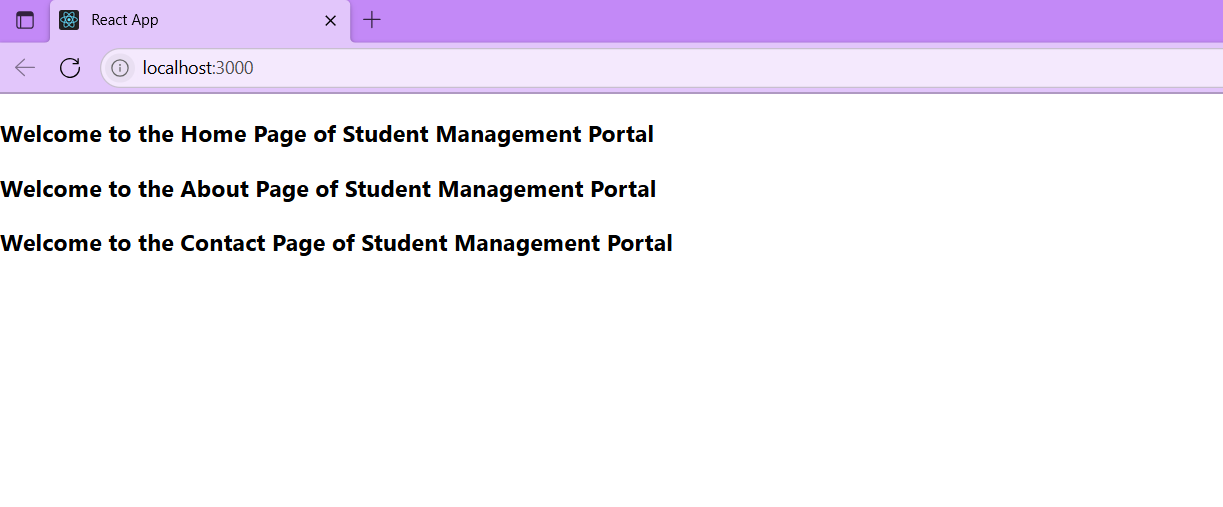
  );

}

export default **App**;

**Output:**

****

****