### ****Explain React Components****

ANS:

**React components** are **independent, reusable building blocks** used to create UI in a React application.

Each component:

Represents a part of the user interface (e.g., header, button, form).

Can **maintain its own state** (if needed).

Can **receive props** (input values) and **return JSX** to render UI.

Components make the UI modular, easier to maintain, and scalable.

### ****Differences Between React Components and JavaScript Functions****

ANS:

| **Feature** | **React Component** | **JavaScript Function** |
| --- | --- | --- |
| Purpose | Used to render UI in React | Performs logic or computation |
| Returns | JSX (UI structure) | Any value (number, string, object, etc.) |
| React Integration | Follows React rules (e.g., useState, props) | Regular JS, not tied to React |
| Lifecycle Methods | Available in Class Components | Not applicable |
| Usage Context | Invoked by JSX (e.g., <MyComponent />) |  |

### ****3.Types of React Components****

ANS:

There are **two main types** of React components:

* **Class Components**
* **Function Components (with or without Hooks)**

### ****4.Explain Class Component****

A **class component** is a React component defined using ES6 class syntax. It can use **state**, **props**, and **lifecycle methods**.

**Key features:**

* Uses render() method to return JSX
* Has a constructor() to initialize state
* Can use lifecycle methods like componentDidMount()

### ****Explain Function Component****

ANS:

A **function component** is a simpler way to write React components using JavaScript functions.

Before React 16.8, they couldn’t use state, but with **Hooks**, they can now use state and side effects.

### ****Define Component Constructor****

ANS:

In **class components**, the constructor() is a special method used to:

**Initialize state**

**Bind methods** (if needed)

**Pass props to the parent class (**super(props)**)**

### ****Define**** render() ****Function****

ANS:

The render() function is **mandatory** in class components and is responsible for returning the **JSX** (UI layout) of the component.

**Rules:**

* Must return **one root element** (can use <div> or <React.Fragment> if needed).
* Called **every time state or props change**.



import './App.css';

import React, { Component } from 'react';

import About from './About';

import Contact from './Contact';

**class** App **extends** Component {

  render() {

    return (

      <div>

        <h1>Welcome to the Home Page of Student Management Portal</h1>

        <About></About>

        <Contact></Contact>

      </div>

    );

  }

}

export default App;

import React from 'react';

**class** About **extends** React.Component {

    render(){

        return(

            <div>

                <h1>Welcome to the About Page of Student Management Portal</h1>

            </div>

        );

    }

}

export default About;

import React from 'react';

**class** Contact **extends** React.Component {

    render(){

        return(

            <div>

                <h1>Welcome to the Contact Page of Student Management Portal</h1>

            </div>

        );

    }

}

export default Contact;

