**3.ReactJS-HOL**

**• Explain React Components**

React components are the basic parts of any React application. A component is a self-contained piece of code that represents a part of the user interface (UI). It can be as small as a button or as large as a whole web page.

* Components can be reused and placed inside other components.
* Each component can manage its own state and respond to props, which are input values from parent components.
* They help keep the code organized, clean, and easy to maintain.

**• Identify the Differences Between Components and JavaScript Functions**

| **Aspect** | **React Component** | **JavaScript Function** |
| --- | --- | --- |
| Purpose | Used to build UI in React | Performs a specific task or calculation |
| Return Value | Returns JSX (UI elements) | Returns a value (number, string, object) |
| React Features | Can use state, props, lifecycle methods | Cannot directly use React features |
| Integration with React | Rendered to the browser by React | Not rendered; used within logic |
| Reusability in UI | Yes, as part of React apps | Not used as UI elements directly |

**• Identify the Types of Components**

* Class Components
* Function Components

**• Explain Class Component**

A class component is defined using ES6 class syntax. It can manage its own state and use lifecycle methods.

Example:

class Welcome extends React.Component {

render() {

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

}

}

* Uses class and extends React.Component
* Must include a render() method
* Can hold internal state using this.state
* Can access lifecycle methods like componentDidMount()

**• Explain Function Component**

A function component is a simpler way to write components using regular JavaScript functions. With React Hooks, function components can also manage state and handle side effects.

Example:

function Welcome(props) {

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

}

* Uses a regular JavaScript function
* Returns JSX directly
* Can use useState, useEffect, and other React Hooks
* Preferred for most new React development because of its simplicity

**• Define Component Constructor**

The constructor in a class component is a special method used to set up the state and bind methods.

Syntax:

constructor(props) {

super(props);

this.state = { count: 0 };

}

* Initializes state
* Binds event handlers (e.g., this.handleClick = this.handleClick.bind(this))

Note: Function components don’t use constructors; they use the useState hook instead.

**• Define render() Function**

The render() function is a required method in class components. It returns the JSX that should be displayed in the browser.

Example:

render() {

return <h1>Hello, World!</h1>;

}

* It must return a single JSX element; wrap multiple elements in a div or fragment
* It is called automatically when a component is first rendered or when its state or props change

**Output:**



