2. ReactJS-HOL

**1. Explain React Components**

React components are **building blocks of a React application**. They are reusable, self-contained pieces of UI that return HTML via **JSX**. Each component can manage its own state and props.

**2. Difference Between Components and JavaScript Functions**

| **Feature** | **React Component** | **JavaScript Function** |
| --- | --- | --- |
| Purpose | Used to build UI | Used for logic or computation |
| Returns | JSX (UI elements) | Any value |
| Lifecycle Methods | Yes (in class components) | No |
| State Management | Yes (with useState or class state) | No |

**3. Types of Components**

* **Class Components** – Use ES6 classes, support lifecycle methods and state.
* **Function Components** – Simpler, use hooks (like useState, useEffect) to manage state/lifecycle.

**4. Class Component**

A class component is a **JavaScript class** that extends React.Component. It has a render() method that returns JSX.

jsx

class Welcome extends React.Component {

render() {

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

}

}

**5. Function Component**

A function component is a **plain JavaScript function** that returns JSX. It’s the modern way to write components.

jsx

function Welcome(props) {

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

}

**6. Component Constructor**

In class components, the constructor is used to **initialize state** and bind methods.

jsx

class Welcome extends React.Component {

constructor(props) {

super(props);

this.state = { name: "Lilly" };

}

}

**7. render() Function**

The render() method is **required** in class components. It returns JSX to be rendered in the UI.

jsx

render() {

return <div>Welcome to React!</div>;

}