**13.ReactJS-HOL**

**1. Explain various ways of conditional rendering**  
React provides multiple ways to implement conditional rendering:

**if/else statement**

if (isLoggedIn) {

return <Dashboard />;

} else {

return <Login />;

}

**Ternary operator**

{isLoggedIn ? <Dashboard /> : <Login />}

**Logical AND (&&) operator**

{isAdmin && <AdminPanel />}

**Immediately Invoked Function Expression (IIFE)**

{(() => {

if (status === 'loading') return <Loader />;

else return <Content />;

})()}

**2. Explain how to render multiple components**  
Multiple components can be rendered by including them inside a parent component:

function App() {

return (

<div>

<Header />

<Sidebar />

<Content />

<Footer />

</div>

);

}

React also supports returning an array of components:

return [<Item1 key="1" />, <Item2 key="2" />];

**3. Define list component**  
A list component is a React component that renders a list of elements based on an array of data.

function NumberList(props) {

const numbers = props.numbers;

const listItems = numbers.map((number) => <li key={number}>{number}</li>);

return <ul>{listItems}</ul>;

}

**4. Explain about keys in React applications**  
Keys are unique identifiers used in lists to help React identify which items have changed, been added, or removed. Using keys improves performance and reduces unnecessary re-renders.

const listItems = numbers.map((number) => <li key={number}>{number}</li>);

Keys should be unique among siblings and ideally use stable identifiers (not indexes).

**5. Explain how to extract components with keys**  
Components can be extracted from list rendering logic, and keys should be assigned at the top level of iteration:

function ListItem(props) {

return <li>{props.value}</li>;

}

function NumberList(props) {

const listItems = props.numbers.map((number) => <ListItem key={number} value={number} />);

return <ul>{listItems}</ul>;

}

**6. Explain React Map, map() function**  
The map() function in JavaScript is often used in React to iterate over arrays and render components:

const names = ['Alice', 'Bob', 'Charlie'];

const nameList = names.map((name) => <p key={name}>Hello, {name}</p>);

This method allows creating dynamic lists in the UI based on data.