**Cognizant Digital Nurture 4.0**  
**Week-7**  
**Name** – Brishti Das  
**Superset ID** – 6363729  
**React Hands-on Exercise 13**

### 1.Explain various ways of conditional rendering

**If/Else Statement**:

Use traditional if or if/else blocks inside a function or component.

Example:

if (isLoggedIn) {

return <Dashboard />;

} else {

return <Login />;

}

**Ternary Operator**:

Compact form of if/else.

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

**Logical AND (**&&**)**:

Renders right-hand side only if the condition is true.

Example: {isLoggedIn && <LogoutButton />}

**Switch Statement**:

Useful for multiple cases.

**Example**:

switch (view) {

case 'home': return <Home />;

case 'profile': return <Profile />;

default: return <NotFound />;

}

### 2.Explain how to render multiple components

Wrap multiple components inside a parent element like <div>, <> (Fragment), or React.Fragment.

**Example:**

return (

<>

<Header />

<MainContent />

<Footer />

</>

);

React Fragment helps avoid extra nodes in the DOM.

### 3. Define list component

A **List Component** displays multiple items using JSX and the map() function.

**Example:**

function NumberList(props) {

const numbers = props.numbers;

return (

<ul>

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

</ul>

);

}

List components are reusable and display data dynamically.

**4.Explain about keys in React applications**

**Keys** are unique identifiers used by React to keep track of elements in a list.

Help React optimize rendering and reordering.

Should be stable, predictable, and unique.

**Example**:

const items = ['a', 'b', 'c'];const listItems = items.map((item, index) => <li key={index}>{item}</li>);

**5. Explain how to extract components with keys**

You can create a child component and pass the key prop when mapping items.

**Example**:

function ListItem(props) {

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

}

function NumberList(props) {

const numbers = props.numbers;

return (

<ul>

{numbers.map((num) => <ListItem key={num} value={num} />)}

</ul>

);

}

Keys should be passed directly to the component within the array.

**6.Explain React Map, map() function**

map() is a JavaScript array function used in React to iterate over an array and return a new array of JSX elements.

It allows rendering lists dynamically from data.

**Example**:

const fruits = ['Apple', 'Banana', 'Cherry'];const fruitList = fruits.map((fruit, index) => <li key={index}>{fruit}</li>);

It's commonly used inside React components to render arrays of elements.