**Lists & Hooks**

**Question 1:** How do you render a list of items in React? Why is it important to use keys when rendering lists?

**Ans:** To render a list of items in React, you typically use the JavaScript map () method to iterate over the array of items and return a JSX element for each item.

**Example:**

const items = ["Apple", "Banana", "Cherry"];

function ItemList() {

return (

<ul>

{items.map((item, index) => (

<li key={index}>{item}</li>

))}

</ul>

); }

**Why is it important to use keys?**

Keys are a crucial attribute when rendering lists because they help React efficiently identify which elements have changed, been added, or removed. This optimization helps React update only the necessary parts of the DOM, improving performance and ensuring a smooth user experience.

**When keys are used:**

1. **Improved Performance**: React uses keys to track changes in the DOM, enabling it to efficiently re-render only the updated components.
2. **Avoid Bugs:** Using keys prevents React from incorrectly associating components with outdated state or properties when the list order changes.

**Question 2:** What are keys in React, and what happens if you do not provide a unique key?

**Ans:** Keys in React are unique identifiers that React uses to track and differentiate between components in a list. They are passed as a key prop when rendering list elements.

Example

const items = ["Nilu", "Nitu", "Niyu"];

function ItemList() {

return (

<ul>

{items.map((item) => (

<li key={item}>{item}</li>

))}

</ul>

);

}

**What happens if you do not provide a unique key?**

* **Performance Degradation**: Without unique keys, React falls back to using the index of the array as the default key. This can cause inefficiencies in rendering, especially when the list changes dynamically (e.g., items added, removed, or reordered).
* **State Mismatch**: React may retain the wrong component state when reordering elements, leading to unexpected behaviors in the UI.
* **Warning Messages**: React will display a warning in the console (Each child in a list should have a unique "key" prop) to highlight the absence of keys or their non-uniqueness.