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

To render a list of items in React, you typically use the .map() function to loop through the array and return JSX for each item. Here's a basic

**example:**

const fruits = ['Apple', 'Banana', 'Cherry'];

function FruitList() {

return (

<ul>

{fruits.map((fruit, index) => (

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

))}

</ul>

);

}

**📌 Why is it important to use keys when rendering lists?**

Keys help React identify which items have changed, been added, or removed. This makes the **re-rendering process more efficient**, especially when lists update dynamically.

**🔑 Key Points:**

* **Keys must be unique among siblings.**
* React uses them to optimize performance.
* Without proper keys, React may **reorder elements incorrectly**, leading to **bugs or unexpected UI behaviour**.

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

**🔑 What are keys in React?**

In React, keys are special string attributes that help identify which elements in a list have changed, been added, or removed. They are used when rendering lists using functions like .map().

**Example:**

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

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

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

));

**⚠️ What happens if you don’t provide a unique key?**

1. **React can't track changes efficiently.**
   * Without a unique key, React might re-render or re-use DOM elements incorrectly.
2. **Unexpected bugs or UI behaviour.**
   * For example, if list items have input fields or animations, their state might get mixed up.
3. **Performance issues.**
   * React diffing algorithm becomes less optimized, causing unnecessary renders.