

## 1. What is `map()` in JavaScript?

`map()` is an array method that creates a new array by applying a function to each item.

### Basic Syntax:

javascript

 Copy

 Edit

```
const numbers = [1, 2, 3]; const doubled = numbers.map(n => n * 2); console.log(doubled); // [2, 4, 6]
```

## 2. Using `map()` in React to Render Lists

In React, we use `map()` to turn an array into a list of elements (usually JSX).

### Example:

jsx

 Copy

 Edit

```
const fruits = ["Apple", "Banana", "Orange"]; function FruitList() { return ( <ul> {fruits.map(fruit => <li>{fruit}</li>)} </ul> ); }
```

◆ This will show:

mathematica

 Copy

 Edit

- Apple
- Banana
- Orange

## 3. What is the `key` Prop in React?

In lists, each element must have a `key` prop — a unique identifier used by React to efficiently update the UI.

### Example with `key` :

jsx

```
{fruits.map((fruit, index) => ( <li key={index}>{fruit}</li> ))}
```

The key should be:

- **Unique**
- **Consistent** between renders
- Not recommended to use `index` if the list might change

✓ Better:

jsx

```
const fruits = [ { id: 1, name: "Apple" }, { id: 2, name: "Banana" } ]; {fruits.map(fruit => ( <li key={fruit.id}>{fruit.name}</li> ))}
```

## ! Why is key important?

React uses `key` to:

- Know which item changed
- Avoid unnecessary re-renders
- Keep things efficient and bug-free in dynamic UIs

## Summary

Concept	Usage	Example
map()	Loop over array to return new JSX	<code>array.map(item =&gt; &lt;li&gt;{item}&lt;/li&gt;)</code>
key	Unique ID for each JSX item in a list	<code>&lt;li key={id}&gt;Item&lt;/li&gt;</code>