

1. Explain About Conditional Rendering in React

Conditional rendering in React refers to displaying different UI elements based on specific conditions (usually state or props). It's similar to how conditionals work in JavaScript (if, else, ternary operator, etc.).

React uses JSX to render UI, and since JSX supports JavaScript expressions, we can use conditional logic directly in our render methods or return statements.

Common Techniques:

- a. Using if statement (inside a function)
- b. Using ternary operator (inline)
- c. Using && operator for conditional output (short-circuit evaluation)
- d. Using switch statement

Sometimes multiple conditions are best handled using switch

2. Explain How to Render Multiple Components

Rendering multiple components in React is simple: just include them inside a parent component's return statement.

```
jsx
function App() {
  return (
    <div>
      <Header />
      <Main />
      <Footer />
    </div>
  );
}
```

3. Define List Component

A **list component** is used to render a list of items (like an array of users, products, or messages). Lists in React are often rendered using the `map()` function.

Example:

```
jsx
function NameList(props) {
  const names = ['Alice', 'Bob', 'Charlie'];
  return (
    <ul>
      {names.map((name, index) => <li key={index}>{name}</li>)}
    </ul>
  );
}
```

4. Explain About Keys in React Applications

Keys are special string attributes used in React to identify which items have changed, been added, or removed from lists. They help optimize re-rendering and keep React components in sync with the DOM.

5. Explain How to Extract Components with Keys

Sometimes it's better to extract a list item into its own **reusable component**. When you do this, **pass the key to the top-level element** in the returned component.

6. Explain React Map and map() Function

The `map()` function in JavaScript is commonly used in React to create **lists of elements** from arrays.

Syntax:

```
js
array.map((item, index) => {
  return <Component key={index} />;
});
```

`map()` function:

- Iterates through an array.
- Transforms each item into a React element.
- Returns a new array of elements.

This is essential for rendering lists dynamically and efficiently.