

MODULE:10 List and Hooks

- Explain Life cycle in Class Component and functional component with Hooks MODULE: 11 Styling & Advance React

Ans:- The introduction of Hooks in React (introduced in version 16.8) allows functional components to have state and lifecycle features, making them more powerful and flexible. Let's explore the lifecycle in both types of components.

- Class component:-

1.Mounting Phase:-

- **constructor**: Called when an instance of the component is being created.
- **render**: Responsible for rendering UI elements.
- **componentDidMount**: Invoked after the component is rendered to the DOM. Ideal for fetching data.

2. Updating Phase:-

- **shouldComponentUpdate**: Determines if the component should update.

- **render:** Re-renders the component with updated data.
- **componentDidUpdate:** Invoked after the component is updated in the DOM.

3.Unmounting Phase:-

- **componentWillUnmount:** Called just before the component is removed from the DOM. Useful for cleanup.

Functional component with **Hooks:**

1. Mounting Phase:

- **useState:** Allows you to add state to your functional components.
- **useEffect** (with an empty dependency array): Similar to **componentDidMount**. Used for side effects (e.g., data fetching).

2. Updating Phase:

- **useState:** Update state using the **set** function returned by **useState**.
- **useEffect** (with dependencies): Similar to **componentDidUpdate**. Runs when specific dependencies change.

3. Unmounting Phase:

- **useEffect** (returning a cleanup function): Similar to **componentWillUnmount**. Cleans up side effects before the component is unmounted.