**WEEK-6: 4. REACT JS – HOL**

**Name:** Abhignya Gitti

**Superset ID:** 6374632

* **Explain the need and Benefits of component life cycle**

React components go through a **lifecycle** from the moment they are created to when they are removed. Lifecycle methods help developers perform useful tasks at key moments — like when the component first appears on screen, after it updates, or just before it disappears. These methods make apps more predictable, help manage things like data fetching, cleanup, and updating, and make components easier to reuse.

Why lifecycle methods are important:

* Let you fetch data from an API once the component is visible (componentDidMount)
* Help you run code after state or props change (componentDidUpdate)
* Clean up tasks like stopping timers or removing event listeners before the component goes away (componentWillUnmount)
* Improve performance by letting you skip re-rendering when nothing changed
* **Identify various life cycle hook methods**

React class components have three main phases:

1. **Mounting (when the component appears)**

* **constructor()**: Sets up initial state and props
* **render()**: Returns the JSX that should be displayed
* **componentDidMount()**: Called after the component appears; used for data loading or setup.

1. **Updating (when state or props change)**

* **shouldComponentUpdate()**: Optional check to skip if changes aren’t needed
* **render()**: Rerenders the UI based on updated props or state
* **componentDidUpdate()**: Called after updates; useful for side-effects like API calls or syncing data**.**

1. **Unmounting (when component is removed)**

* **componentWillUnmount()**: Clean up any subscriptions, timers, or listeners before it's removed from the page
* **List the sequence of steps in rendering a component**

1. React calls the **constructor()** to set up props and initial state
2. It runs **render()** to build and display the JSX
3. After the element is rendered to the page, **componentDidMount()** runs for setup tasks
4. When props or state change:
   * React may call **shouldComponentUpdate()** to check if a rerender is needed
   * If yes, React runs **render()** again
   * After UI updates, **componentDidUpdate()** executes for side-effects
5. If the component is removed, **componentWillUnmount()** executes to clean up resources