**4.ReactJS-HOL**

**• Explain the Need and Benefits of Component Lifecycle**

**🔹 Need:**

* React components go through stages: **creation**, **update**, and **removal**.
* Lifecycle methods allow running code at each of these stages.
* Useful for managing **side effects** like data fetching, animations, or cleanup.

**🔹 Benefits:**

* Gives **precise control** over a component’s behavior.
* Helps **manage resources** efficiently (e.g., starting/stopping timers).
* Ensures **data is fetched** at the correct time (after mounting).
* **Improves performance** by avoiding unnecessary re-renders.
* Helps **track changes** in props or state and react accordingly.

**• Identify Various Lifecycle Hook Methods**

**🔹 1. Mounting Phase (when component is created and added to DOM):**

* constructor() → Initializes state and binds methods.
* getDerivedStateFromProps() → Syncs state with props before render.
* render() → Returns the JSX to be displayed.
* componentDidMount() → Runs once after the component mounts (ideal for API calls, timers, subscriptions).

**🔹 2. Updating Phase (when props or state change):**

* getDerivedStateFromProps() → Runs again before rendering with updated props/state.
* shouldComponentUpdate() → Decides whether the component should re-render.
* render() → Renders the updated JSX.
* getSnapshotBeforeUpdate() → Captures snapshot before DOM update (e.g., scroll position).
* componentDidUpdate() → Runs after updates; useful for handling post-update logic or API calls.

**🔹 3. Unmounting Phase (when component is removed from DOM):**

* componentWillUnmount() → Used for cleanup (e.g., removing listeners, canceling requests, clearing timers).

**🔸 Note:**

* In **function components**, the same behavior is handled using the useEffect() hook.

**• List the Sequence of Steps in Rendering a Component**

**🔹 1. Initialization:**

* constructor() is called.
* State is initialized.
* Event handlers are bound.
* Happens **once** when the component is created.

**🔹 2. Mounting Phase:**

* getDerivedStateFromProps() is optionally called.
* render() generates the JSX.
* componentDidMount() is called **after** the component is inserted into the DOM.
* Ideal place for:
  + **API calls**
  + **Starting animations**
  + **Setting up event listeners or subscriptions**

**🔹 3. Updating Phase:**

* Triggered when props or state change.
* getDerivedStateFromProps() runs (if used).
* shouldComponentUpdate() decides whether to re-render.
* render() is called again with updated data.
* getSnapshotBeforeUpdate() captures pre-update info (optional).
* componentDidUpdate() runs **after** the update is complete.
* Ideal for:
  + **Reacting to changes**
  + **Calling APIs again if needed**

**🔹 4. Unmounting Phase:**

* componentWillUnmount() is called.
* Used for:
  + **Clearing timers**
  + **Removing event listeners**
  + **Canceling API/network requests**

**Output:**





