**OBJECTIVES**

1. **Explain the Need and Benefits of Component Lifecycle**

**Need:**

* React components go through various stages (mounting, updating, unmounting).
* Developers need control over what happens at each stage (e.g., fetch data when component loads, clean up before unmount).

**Benefits:**

* **Better control over component behaviour:**  
  e.g., fetch data only once when the component mounts.
* **Optimize performance:**  
  Prevent unnecessary renders, update only when needed.
* **Resource management:**  
  Clean up memory, listeners, or subscriptions to avoid memory leaks.
* **Debugging support:**  
  Understand and trace the flow of a component's rendering and updates.

1. **Identify Various Lifecycle Hook Methods**

| **Phase** | **Hook Method** | **Description** |
| --- | --- | --- |
| **Mounting** | constructor() | Initialize state/props. |
|  | static getDerivedStateFromProps() | Sync props to state (rare). |
|  | render() | Required. Returns JSX. |
|  | componentDidMount() | Runs after the component is mounted. Ideal for data fetching. |
| **Updating** | shouldComponentUpdate() | Controls re-rendering. |
|  | getSnapshotBeforeUpdate() | Capture DOM info before update. |
|  | componentDidUpdate() | Runs after the update is done. |
| **Unmounting** | componentWillUnmount() | Cleanup tasks (e.g., event listeners, API calls). |

1. **List the Sequence of Steps in Rendering a Component**

* Here’s the lifecycle sequence for class components:
* **Mounting Phase:**

1. constructor()
2. static getDerivedStateFromProps()
3. render()
4. componentDidMount()

* **Updating Phase (on props/state change):**

1. static getDerivedStateFromProps()
2. shouldComponentUpdate()
3. render()
4. getSnapshotBeforeUpdate()
5. componentDidUpdate()

* **Unmounting Phase:**

1. componentWillUnmount()