**Cognizant Digital Nurture 4.0**

**Week-6**

**Name -Brishti Das**

**Superset id. -6363729**

**React Handson Exercise 4**

React Component Lifecycle

# Explain the Need and Benefits of Component Lifecycle in React

React Component Lifecycle refers to the series of methods that are invoked at different stages of a component's existence—mounting, updating, and unmounting.

* Need for Lifecycle Methods:
* • To initialize data before the component renders.
* • To perform side effects like fetching data from APIs.
* • To optimize performance through control over updates.
* • To clean up resources (like timers or subscriptions) before the component is destroyed.
* Benefits:
* • Better control over how a component behaves at different stages.
* • Cleaner and more predictable code.
* • Efficient resource management.
* • Helps in debugging and monitoring component changes.
* • Enables reusability and modular design.

# Identify Various Lifecycle Hook Methods

React provides lifecycle methods (class components) or hooks (functional components) based on the component's phase:

1. 1. Mounting Phase:

* • constructor()
* • static getDerivedStateFromProps()
* • render()
* • componentDidMount()

1. 2. Updating Phase:

* • static getDerivedStateFromProps()
* • shouldComponentUpdate()
* • render()
* • getSnapshotBeforeUpdate()
* • componentDidUpdate()

1. 3. Unmounting Phase:

* • componentWillUnmount()
* Additional Methods:
* • componentDidCatch() – for error boundaries.
* • getDerivedStateFromError() – catch errors in child components.

# List the Sequence of Steps in Rendering a Component

The lifecycle of a class component in React can be visualized in a sequence:

* Mounting Sequence:
* • constructor()
* • getDerivedStateFromProps()
* • render()
* • componentDidMount()
* Updating Sequence (on props/state change):
* • getDerivedStateFromProps()
* • shouldComponentUpdate()
* • render()
* • getSnapshotBeforeUpdate()
* • componentDidUpdate()
* Unmounting Sequence:
* • componentWillUnmount()