**ReactJs**

**Module – 10 Lists and Hooks**

**• Explain Life cycle in Class Component and functional component with Hooks.**

-> Brief about life cycle in Class component and functional component is shown below.

**- Life cycle in Class Component and functional component includes:**

**Mounting:**

- The mounting phase refers to the initial setup and rendering of a component. During the mounting phase, specific lifecycle methods are called for both Class Components and Functional Components with Hooks.

**componentWillMount():-** This method is called before the component mounts on DOM or we can say before render method is called.  
**componentDidMount():-** This method is called after the component mounted on DOM or we can say after render method is called.

**Updating:**

- The updating phase refers to the period during which a component is re-rendered due to changes in its state or props. This phase occurs after the initial rendering and involves updating the component to reflect the new state or prop values.

**componentShouldUpdate():-** This method is called when you want to re-render the component on some condition. By default, it returns true.

**componentWillUpdate():-** This method is called before rerendering has done and when you want to do some functionalities just before re-rendering take place.

**componentDidUpdate():-**This method is called after the rerendering has done or we can say after component mounted on DOM.

**Unmounting:**

- The unmounting phase in React occurs when a component is being removed from the DOM. During this phase, cleanup tasks and actions can be performed before the component is completely taken out of the application.

**componentWillUnmount():-** This method is called before the component Unmounting on DOM. Denoting that it will Unmounted soon or we can say end of the component life cycle.

**Example:- Functional components.**

import React, { useEffect } from "react";

**Mounting:**

useEffect(() => {

  console.log("DidMount");

}, []);

**Updating:**

useEffect(() => {

  return () => console.log("DidUpdate");

}, [data.number]);

**Unmounting:**

useEffect(() => {

  console.log("DidMount");

  return () => console.log("WillUnmount");

}, []);

**Example:- Class components:**

**Mounting:**

componentDidMount(){

    console.log("DidMount")

    }

**Updating:**

componentDidUpdate(){

    console.log("DidUpdate")

    }

**Unmounting:**

componentWillUnmount(){

    console.log("WillUnmount")

    }