Name:- kinjal savaliya

MODULE: 4 (List and Hooks)

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

The Component Lifecycle

Each component in React has a lifecycle which you can monitor and manipulate during its three main phases. The three phases are: Mounting, Updating, and Unmounting.

Mounting

The component rendered to the DOM for the first time. This is called mounting. These methods are called in the following order when an instance of a component is being created and inserted into the DOM.

- constructor()
- static getDerivedStateFromProps() rarely case use
- render()
- componentDidMount()

Updating

An update can be caused by changes to props or state. These methods are called in the following order when a component is being re-rendered

- static getDerivedStateFromProps() rarely case use
- shouldComponentUpdate() rarely case use
- render()
- getSnapshotBeforeUpdate() rarely case use
- componentDidUpdate()

Unmounting

When component removed from DOM. This is called unmounting. Below method is called in this phase.

- componentWillUnmount()

Lifecycle Methods

constructor()

The constructor for a React component is called before it is mounted. The constructor call only once in whole lifecycle. You and set initial value for this component.

Constructors are only used for two purposes 1. Initializing local state by assigning an object to this.state 2. Binding event handler methods to an instance.

static getDerivedStateFromProps()

It is invoked before calling render method. This method is used when state depends on props, whenever props is changed, then state automatically will be changed. getDerivedStateFromProps method use for both phase mounting and updating of the component. If null is returned then nothing is changed in the state.

render()

This method is the only required method in a class component. The render() function should be pure, meaning that it does not modify component state, which means it returns the same output each time it's invoked.

componentDidMount()

This is invoked immediately after the component is mounted. If you load data using api then it right place for request data using api.

shouldComponentUpdate()

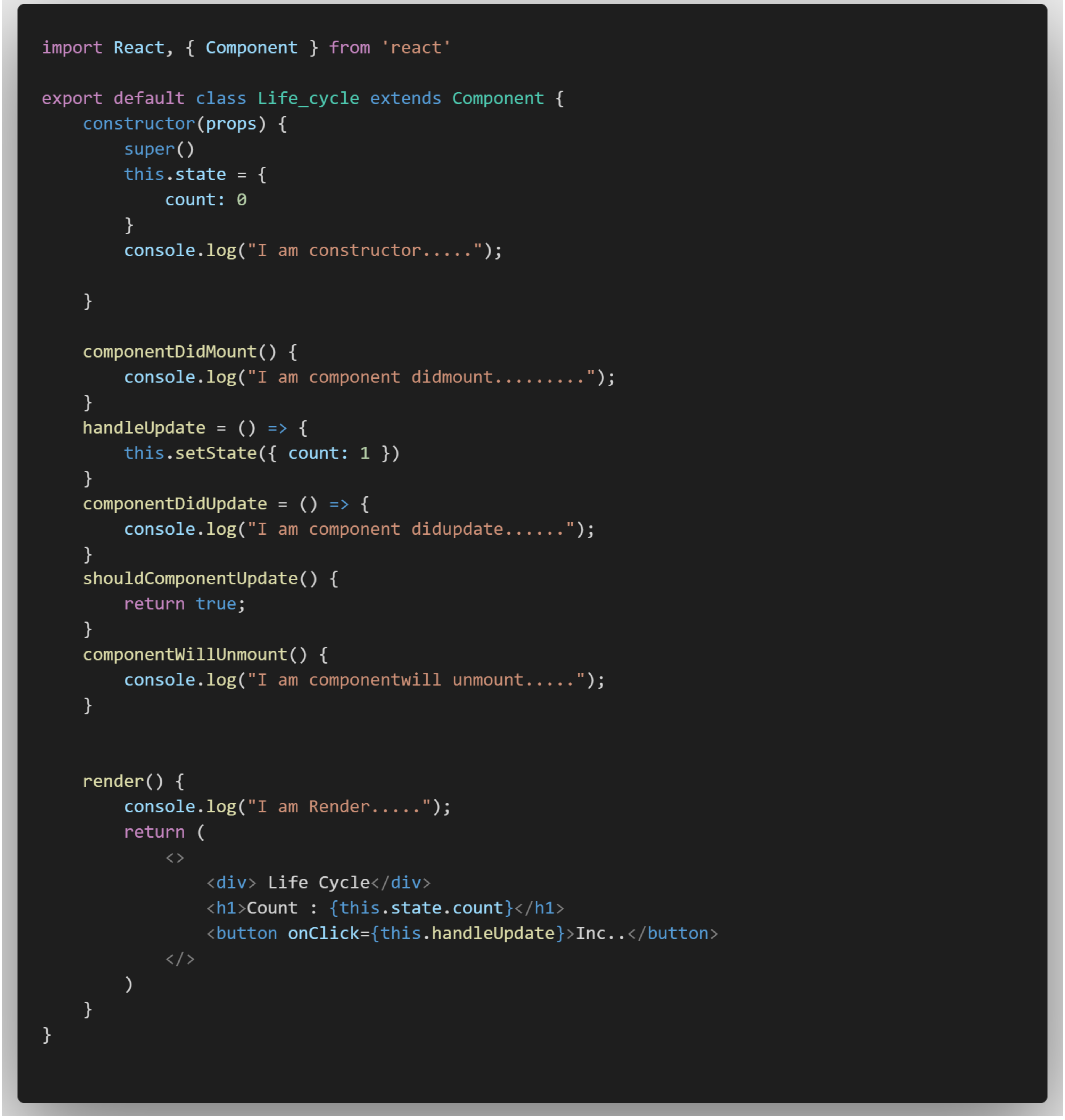
This is invoked before rendering when new props or state are being received. This methods return true or false according the component re-render or skipped. By default, this method returns true. This method is not called for the initial render.

componentDidUpdate()

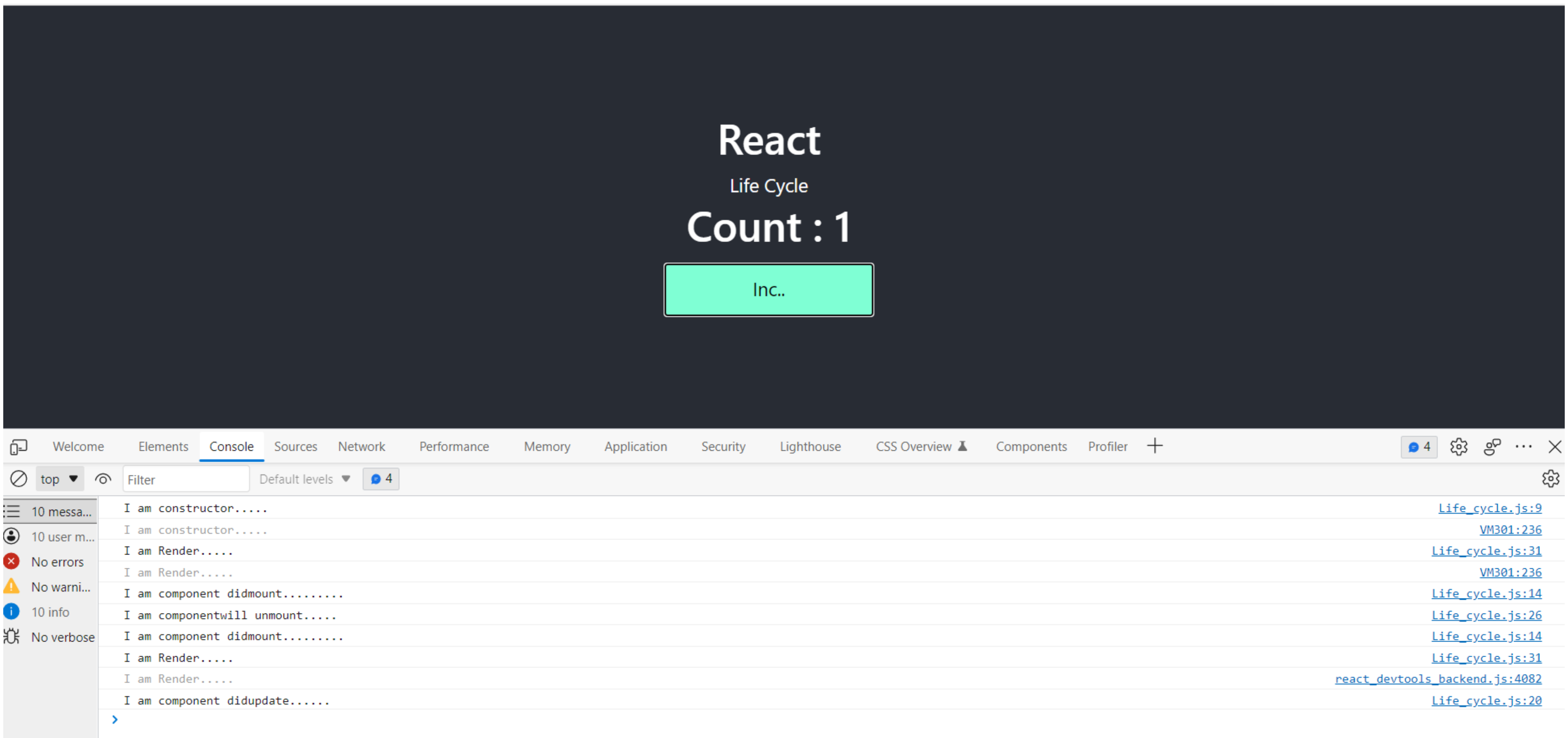
This method immediately executed on the DOM when the component has been updated. Update occurs by changing state and props. This method is not called for the initial render. This is a good place for compare the current props to previous props.

componentWillUnmount()

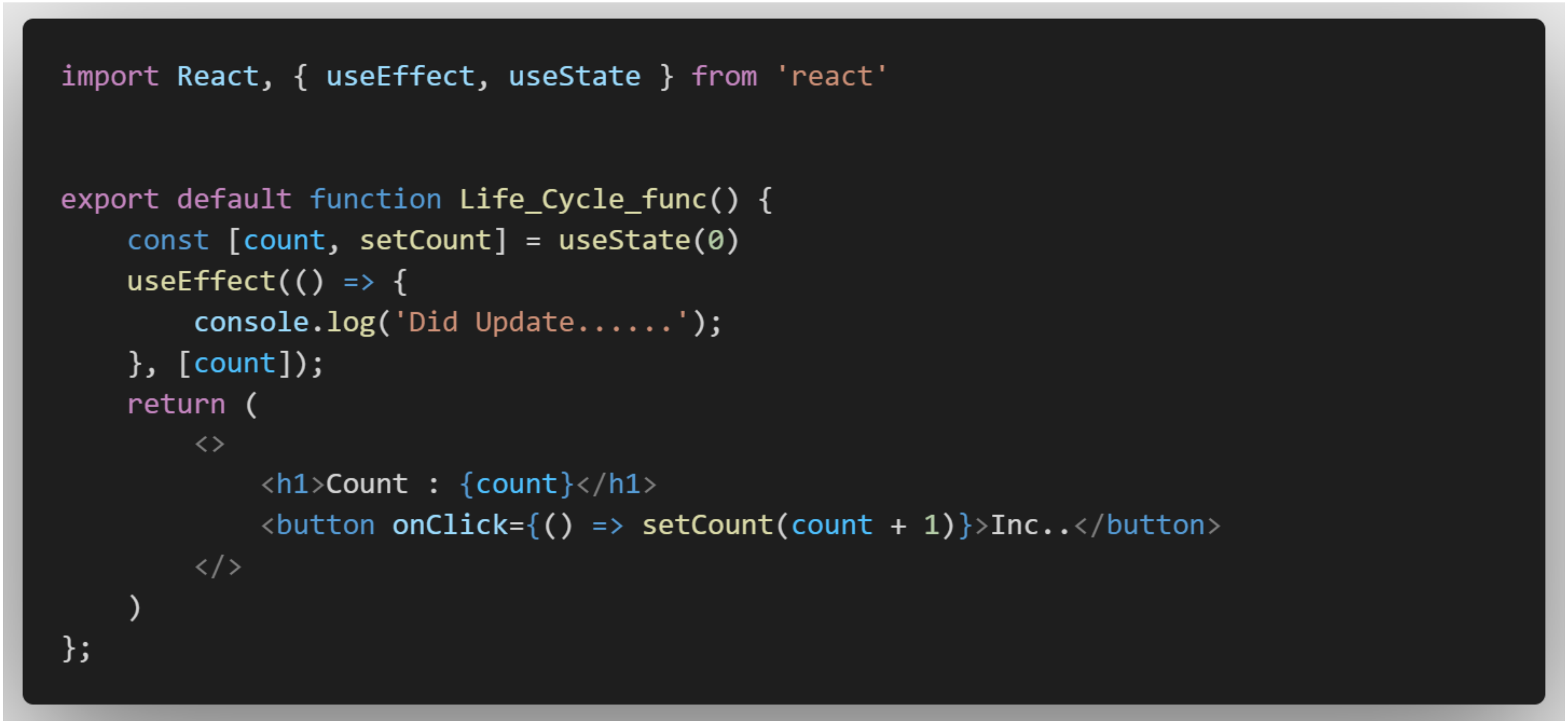
This method immediately executed when the component is unmounted and destroy from the DOM. Means this method is called when a component is being removed from the DOM.



```
import React, { Component } from 'react'

export default class Life_cycle extends Component {
    constructor(props) {
        super()
        this.state = {
            count: 0
        }
        console.log("I am constructor.....");

    }

    componentDidMount() {
        console.log("I am component didmount.........");
    }
    handleUpdate = () => {
        this.setState({ count: 1 })
    }
    componentDidUpdate = () => {
        console.log("I am component didupdate......");
    }
    shouldComponentUpdate() {
        return true;
    }
    componentWillUnmount() {
        console.log("I am componentwill unmount.....");
    }


    render() {
        console.log("I am Render.....");
        return (
            <>
                <div> Life Cycle</div>
                <h1>Count : {this.state.count}</h1>
                <button onClick={this.handleUpdate}>Inc..</button>
            </>
        )
    }
}
```



```
I am constructor.....                          Life_cycle.js:9
I am constructor.....                          VM301:236
I am Render.....                               Life_cycle.js:31
I am Render.....                               VM301:236
I am component didmount.........               Life_cycle.js:14
I am componentwill unmount.....                Life_cycle.js:26
I am component didmount.........               Life_cycle.js:14
I am Render.....                               Life_cycle.js:31
I am Render.....                               react_devtools_backend.js:4082
I am component didupdate......                 Life_cycle.js:20
```



```
import React, { useEffect, useState } from 'react'


export default function Life_Cycle_func() {
    const [count, setCount] = useState(0)
    useEffect(() => {
        console.log('Did Update......');
    }, [count]);
    return (
        <>
            <h1>Count : {count}</h1>
            <button onClick={() => setCount(count + 1)}>Inc..</button>
        </>
    )
};
```