**MODULE: 10 List and Hooks**

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

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

Mounting means putting elements into the DOM.

React has four built-in methods that gets called, in this order, when mounting a component:

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

The render() method is required and will always be called, the others are optional and will be called if you define them.

Updating

The next phase in the lifecycle is when a component is *updated*.

A component is updated whenever there is a change in the component's state or props.

React has five built-in methods that gets called, in this order, when a component is updated:

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

The render() method is required and will always be called, the others are optional and will be called if you define them.

Unmounting

The next phase in the lifecycle is when a component is removed from the DOM, or *unmounting* as React likes to call it.

React has only one built-in method that gets called when a component is unmounted:

* componentWillUnmount()

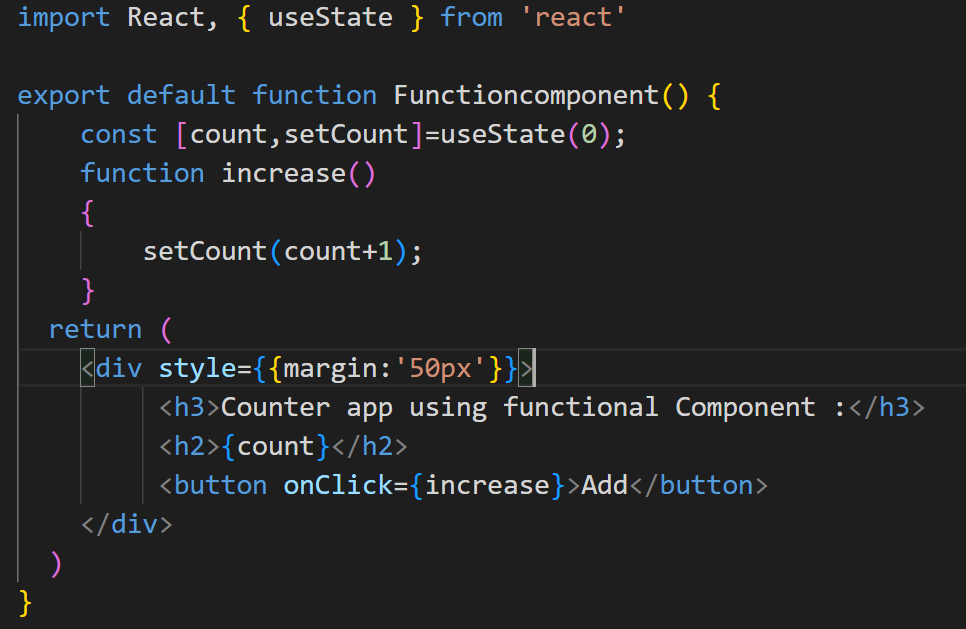
Example



- **functional component**

A functional component in React is a simpler and more concise way to define a component using a JavaScript function. Before the introduction of Hooks in React 16.8, functional components were stateless and didn't have the ability to manage local state or lifecycle methods. With the introduction of Hooks, functional components can now have state and use lifecycle methods, making them a powerful alternative to class components.

Example of functional component



* **Class Component**

This is the bread and butter of most modern web apps built in ReactJS.These components are simple classes(made up of multiple functions that add functionality to the application).

Example of Class component

