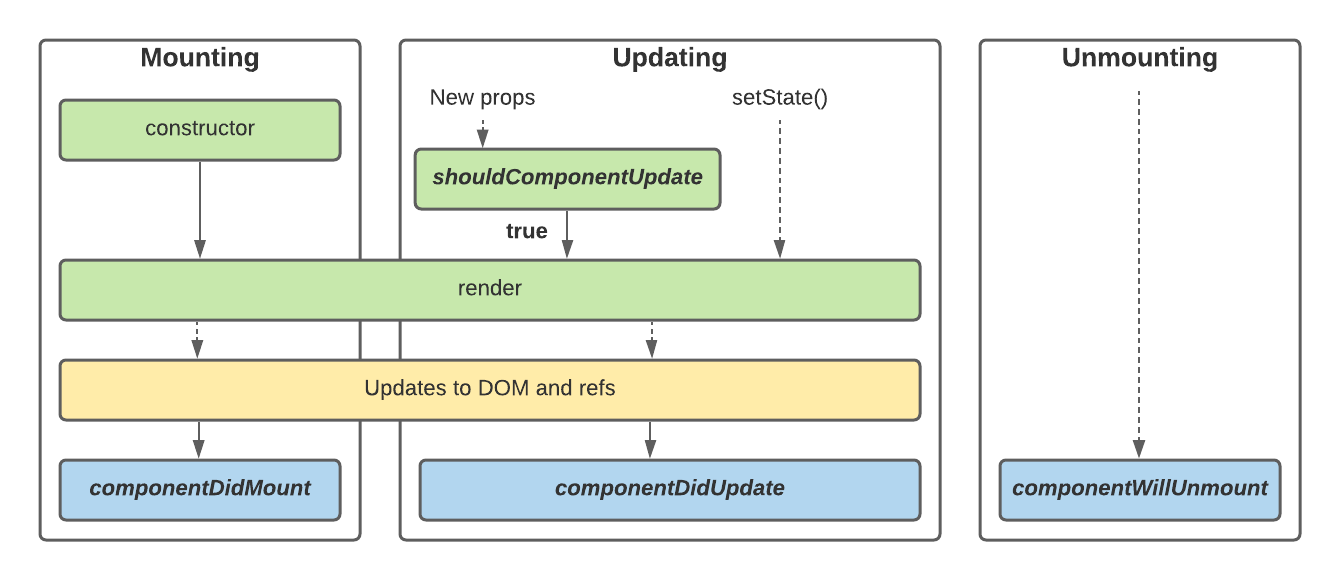
MODULE: 10 List and Hooks

**Que : 1 Explain Life cycle in Class Component and functional component with Hooks.**

**Ans.**

**Life cycle in Class Component and functional component with Hooks:**



**Lifecycle of a React component:**

1. Initial Render or Mount
2. Update (When the states used in the component or props added to the component is changed)
3. Unmounting

**1.Function Components:**

Function components are some of the more common components that will come across while working in the React. These are simply JavaScript functions. We can create a function component to React by writing a JavaScript function.

**Example of functional component:**

import React, { useState } from "react";

export default function Lifecycle() {

  const [count, setCount] = useState(0);

  const Increase = () => {

    setCount(count + 1);

  };

  return (

    <div style={{ margin: "50px" }}>

      <h3>Couter App Using Functional Components :</h3>

      <h2>{count}</h2>

      <button onClick={Increase}>Add</button>

    </div>

  );

}

**2.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:**

import React, { Component } from "react";

class ClassComponent extends React.Component {

  constructor() {

    super();

    this.state = {

      count: 0,

    };

    this.increase = this.increase.bind(this);

  }

  increase() {

    this.setState({ count: this.state.count + 1 });

  }

  render() {

    return (

      <div style={{ margin: "50px" }}>

        <h3>Couter App Using Class Components :</h3>

        <h2>{this.state.count}</h2>

        <button onClick={this.increase}>Add</button>

      </div>

    );

  }

}

export default ClassComponent;