**Unit 4**

**Q.1) Create an application in ReactJs to implement component lifecycle.**

**LifecycleClassComponent.js**

import React, { Component } from 'react';

class LifecycleClassComponent extends Component {

constructor(props) {

super(props);

console.log('Constructor: Component is being created');

this.state = {

message: 'Hello, React Lifecycle!',

};

}

static getDerivedStateFromProps(nextProps, nextState) {

console.log('getDerivedStateFromProps: Called before every render');

return null;

}

shouldComponentUpdate(nextProps, nextState) {

console.log('shouldComponentUpdate: Deciding if re-render is necessary');

return true;

}

getSnapshotBeforeUpdate(prevProps, prevState) {

console.log('getSnapshotBeforeUpdate: Capture some data before DOM update');

return null;

}

componentDidUpdate(prevProps, prevState, snapshot) {

console.log('componentDidUpdate: Component updated successfully');

}

componentDidMount() {

console.log('componentDidMount: Component mounted (initial render completed)');

}

componentWillUnmount() {

console.log('componentWillUnmount: Component is being removed');

}

render() {

console.log('render: Component rendering');

return (

<div>

<h1>{this.state.message}</h1>

<button onClick={() => this.setState({ message: 'Updated Message!' })}>

Update Message

</button>

</div>

);

}

}

export default LifecycleClassComponent;

**LifecyleFunctionComponent.js**

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

const LifecycleFunctionComponent = () => {

  const [message, setMessage] = useState('Hello, React Hooks!');

  useEffect(() => {

    console.log('useEffect: Component mounted or updated');

    return () => {

      console.log('useEffect Cleanup: Component will unmount');

    };

  }, [message]);

  const updateMessage = () => {

    setMessage('Updated Message using Hooks!');

  };

  return (

    <div>

      <h1>{message}</h1>

      <button onClick={updateMessage}>Update Message</button>

    </div>

  );

};

export default LifecycleFunctionComponent;

**App.js**

import React from 'react';

import './App.css';

import LifecycleClassComponent from './LifecycleClassComponent';

import LifecycleFunctionComponent from './LifecycleFunctionComponent';

function App() {

  return (

    <div className="App">

      <h1>React Component Lifecycle Demo</h1>

      <h2>Class Component with Lifecycle Methods</h2>

      <LifecycleClassComponent />

      <h2>Functional Component with Hooks</h2>

      <LifecycleFunctionComponent />

    </div>

  );

}

export default App;

**index.js**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App';

import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

reportWebVitals();

Output :



