# Practical No: 4

1. **Mounting and Updating Header.js**

import React, { Component } from 'react';

class Header extends React.Component { constructor(props) {

super(props);

this.state = {favoritecolor: "red"};

}

componentDidMount() { setTimeout(() => {

this.setState({favoritecolor: "yellow"})

}, 5000)

}

getSnapshotBeforeUpdate(prevProps, prevState) { document.getElementById("div1").innerHTML =

"Before the update, the favorite was " + prevState.favoritecolor;

}

componentDidUpdate() { document.getElementById("div2").innerHTML = "The updated favorite is " + this.state.favoritecolor;

}

render() { return (

<div>

<h1>My Favorite Color is {this.state.favoritecolor}</h1>

<div id="div1"></div>

<div id="div2"></div>

</div>

);

}

}

export default Header;

# App.js:

import Header from './header';

//import Unmount from './unmount'; import './App.css';

function App() { return (

<div className='App'>

<Header/>

</div>

);

}

export default App;

# Output:



1. **Unmounting**

Unmount.js:

import { render } from "@testing-library/react"; import React from "react";

class Unmount extends React.Component{ constructor(){

super(); this.state={

delete: false,

};

}

render(){ return(

<div>

<h1>User List</h1>

<button onClick={()=>this.setState({delete:!this.state.delete})}> Delete Users</button>

{this.state.delete? null:<User/>}

</div>

);

}

}

class User extends React.Component{ componentWillUnmount(){

alert('Delete user successfully!');

}

render(){ return(

<div>

<h3>UserName: Rahul</h3>

<h3>Email: [Rahul@gmail.com</h3>](mailto:Rahul@gmail.com)

</div>

)

}

}

export default Unmount;

# App.js:

//import Header from './header'; import Unmount from './unmount'; import './App.css';

function App() { return (

<div className='App'>

<Unmount/>

</div>

);

}

export default App;

# Output:



