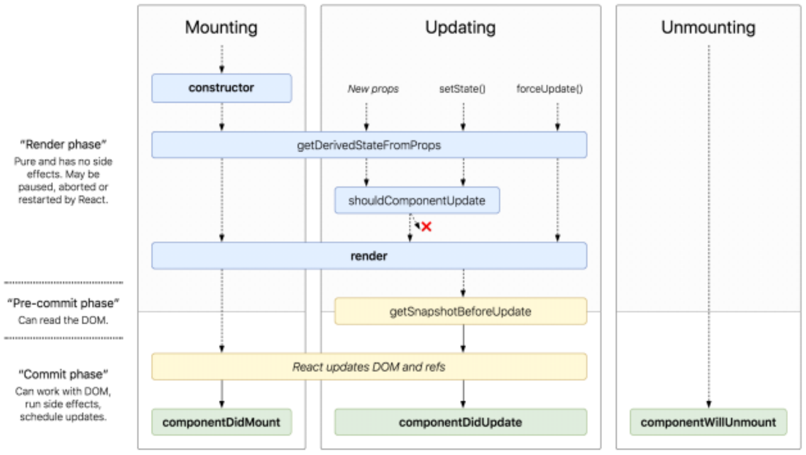
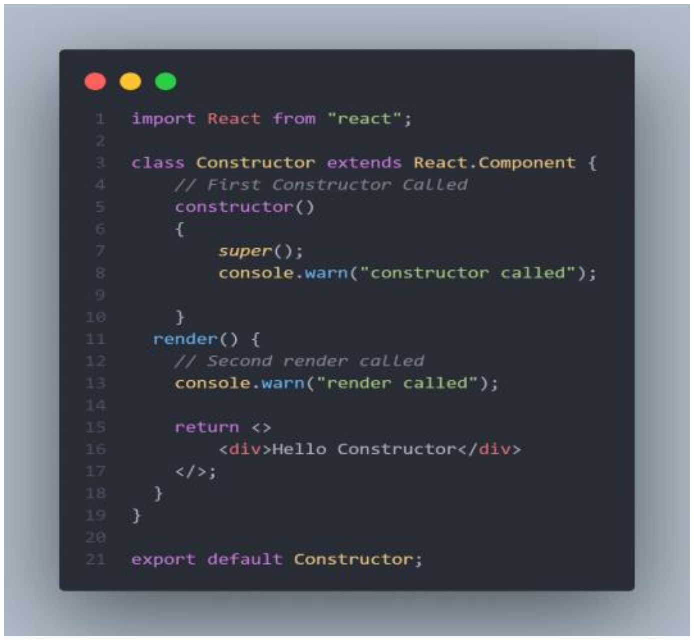
React : Day3

Saturday, November 30, 2024 7:09 PM

React Lifecycle Method :

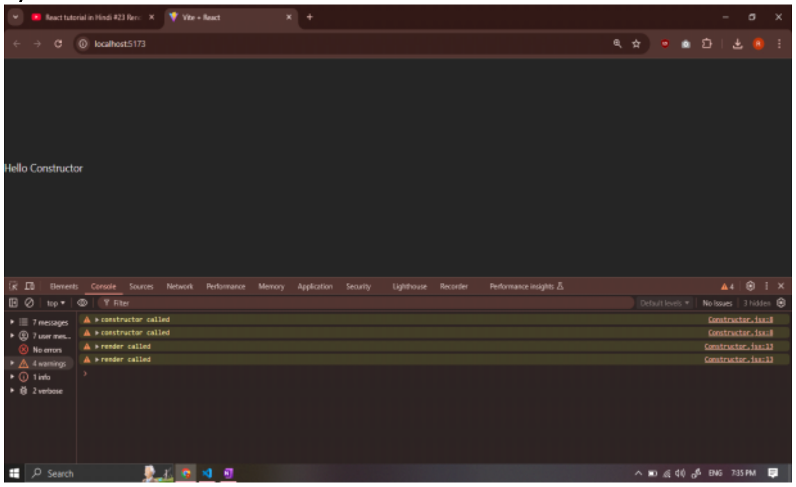


Constructor lifecycle method :



```
import React from "react";

class Constructor extends React.Component {
    // First Constructor Called
    constructor()
    {
        super();
        console.warn("constructor called");

    }
  render() {
    // Second render called
    console.warn("render called");

    return <>
        <div>Hello Constructor</div>
    </>;
  }
}

export default Constructor;
```

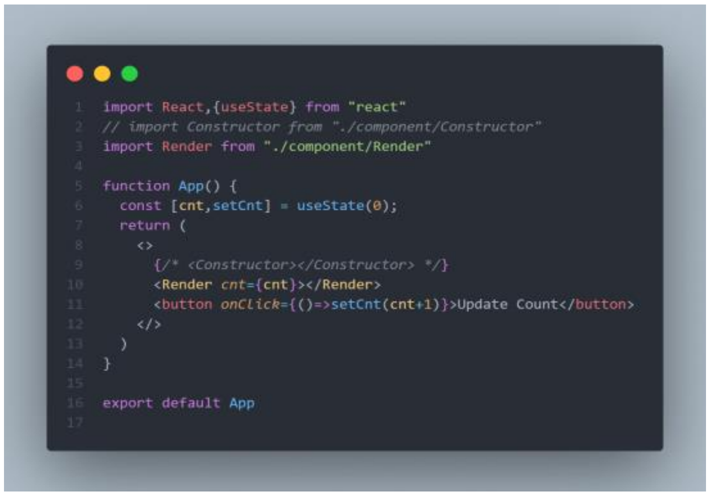
O/P :



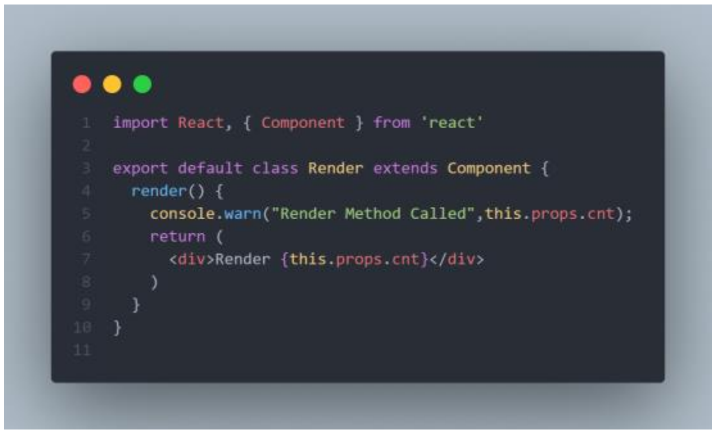
```
Hello Constructor
```

```
▲ ▸ constructor called          Constructor.jsx:8
▲ ▸ constructor called          Constructor.jsx:8
▲ ▸ render called               Constructor.jsx:13
▲ ▸ render called               Constructor.jsx:13
```

Render Lifecycle method :
When we pass props then react render method called

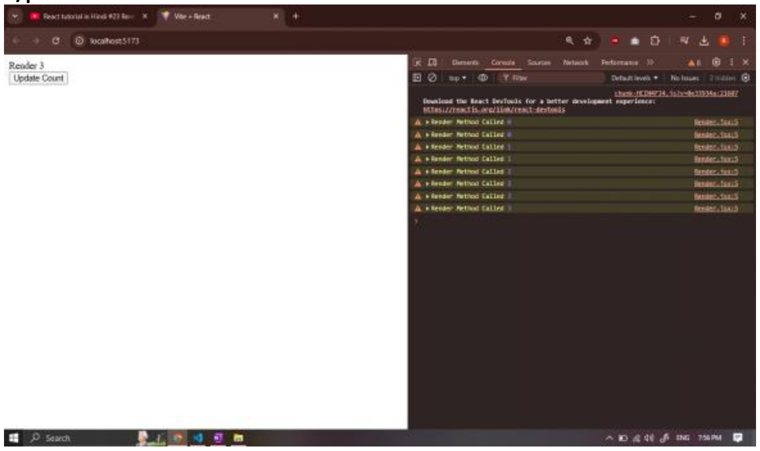


```
import React,{useState} from "react"
// import Constructor from "./component/Constructor"
import Render from "./component/Render"

function App() {
  const [cnt,setCnt] = useState(0);
  return (
    <>
      {/* <Constructor></Constructor> */}
      <Render cnt={cnt}></Render>
      <button onClick={()=>setCnt(cnt+1)}>Update Count</button>
    </>
  )
}

export default App
```



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

export default class Render extends Component {
  render() {
    console.warn("Render Method Called",this.props.cnt);
    return (
      <div>Render {this.props.cnt}</div>
    )
  }
}
```

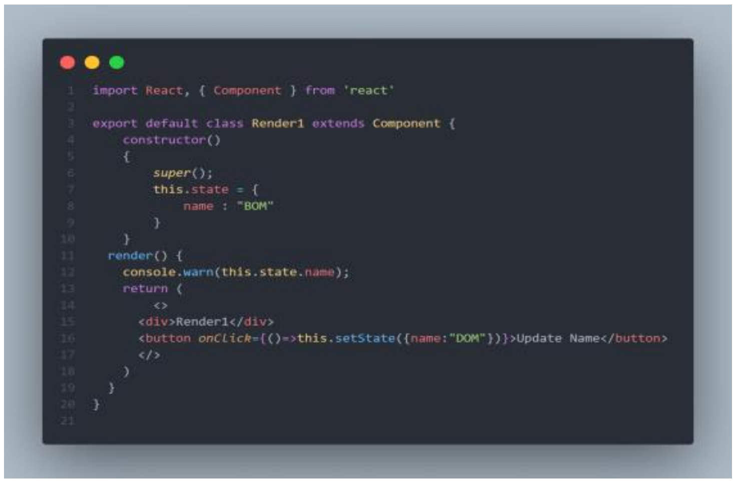
o/p :



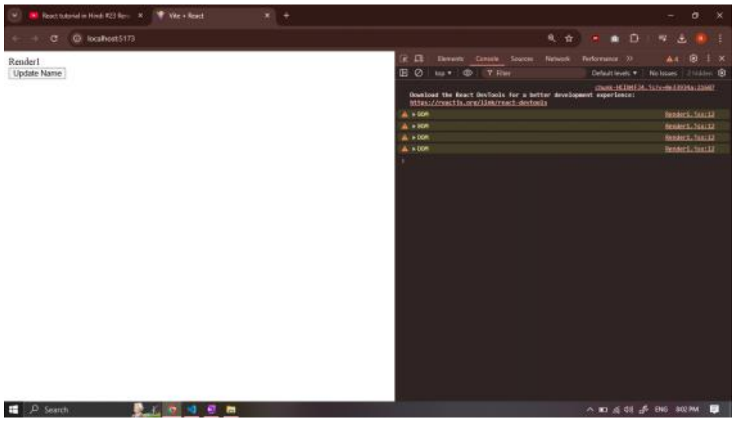
```
Render 3
Update Count
```

```
Download the React DevTools for a better development experience: https://reactjs.org/link/react-devtools
▸ Render Method Called 0          Render.jsx:5
▸ Render Method Called 0          Render.jsx:5
▸ Render Method Called 1          Render.jsx:5
▸ Render Method Called 1          Render.jsx:5
▸ Render Method Called 2          Render.jsx:5
▸ Render Method Called 2          Render.jsx:5
▸ Render Method Called 3          Render.jsx:5
▸ Render Method Called 3          Render.jsx:5
```

When we change state then also componet re-render



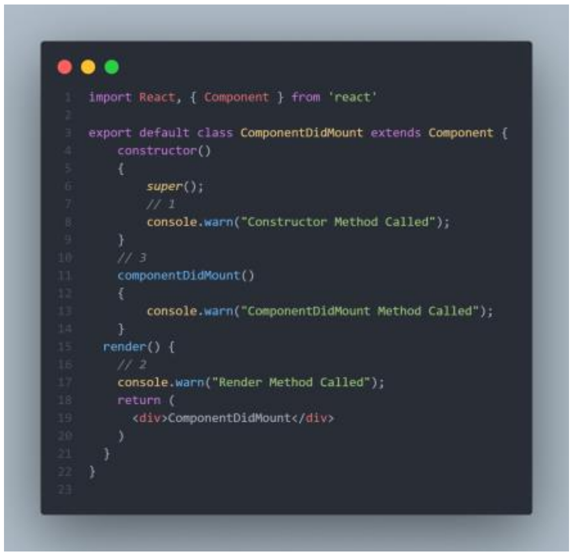
```jsx
import React, { Component } from 'react'

export default class Render1 extends Component {
    constructor()
    {
        super();
        this.state = {
            name : "BOM"
        }
    }
  render() {
    console.warn(this.state.name);
    return (
        <>
      <div>Render1</div>
      <button onClick={()=>this.setState({name:"DOM"})}>Update Name</button>
      </>
    )
  }
}
```



```
Render1
Update Name
```

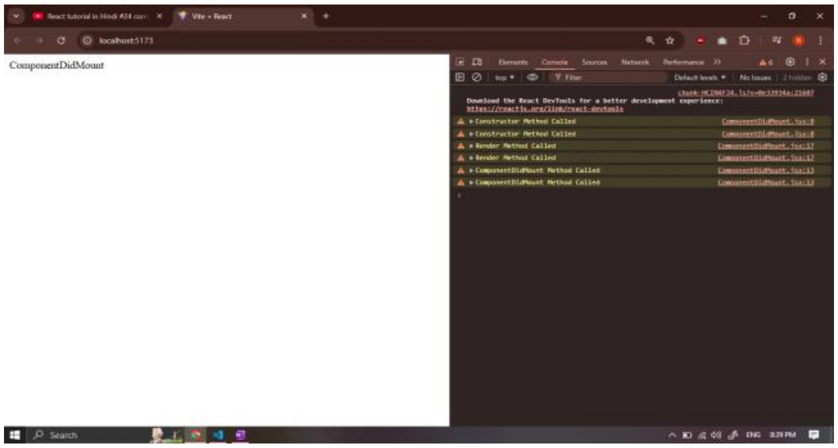
```
Download the React DevTools for a better development experience:
https://reactjs.org/link/react-devtools
⚠ ▸ DOM                                   Render1.jsx:12
⚠ ▸ DOM                                   Render1.jsx:12
⚠ ▸ DOM                                   Render1.jsx:12
⚠ ▸ DOM                                   Render1.jsx:12
```

ComponentDidMount method Called :-



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

export default class ComponentDidMount extends Component {
    constructor()
    {
        super();
        // 1
        console.warn("Constructor Method Called");
    }
    // 3
    componentDidMount()
    {
        console.warn("ComponentDidMount Method Called");
    }
  render() {
    // 2
    console.warn("Render Method Called");
    return (
      <div>ComponentDidMount</div>
    )
  }
}
```

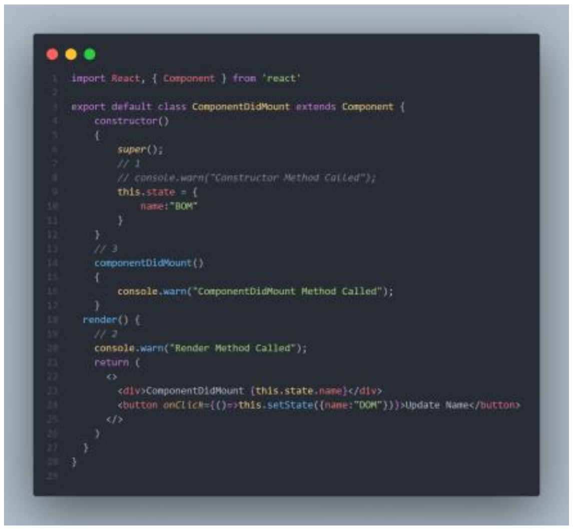
O/P :-



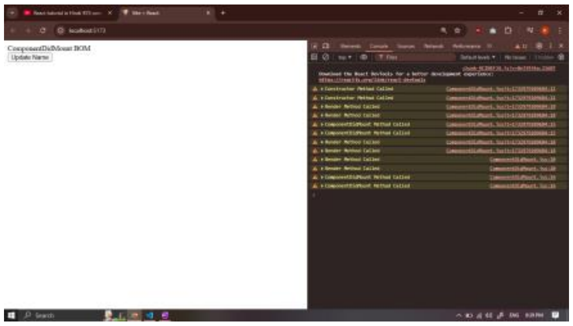
ComponentDidMount

```
chunk-HCINHF24.js?v=0e33934a:21607
Download the React DevTools for a better development experience:
https://reactjs.org/link/react-devtools
▸ Constructor Method Called                 ComponentDidMount.jsx:8
▸ Constructor Method Called                 ComponentDidMount.jsx:8
▸ Render Method Called                      ComponentDidMount.jsx:17
▸ Render Method Called                      ComponentDidMount.jsx:17
▸ ComponentDidMount Method Called           ComponentDidMount.jsx:13
▸ ComponentDidMount Method Called           ComponentDidMount.jsx:13
```

Note : When we update state and props then componentDIdMount() not called
ComponentDidMount is used for API Call



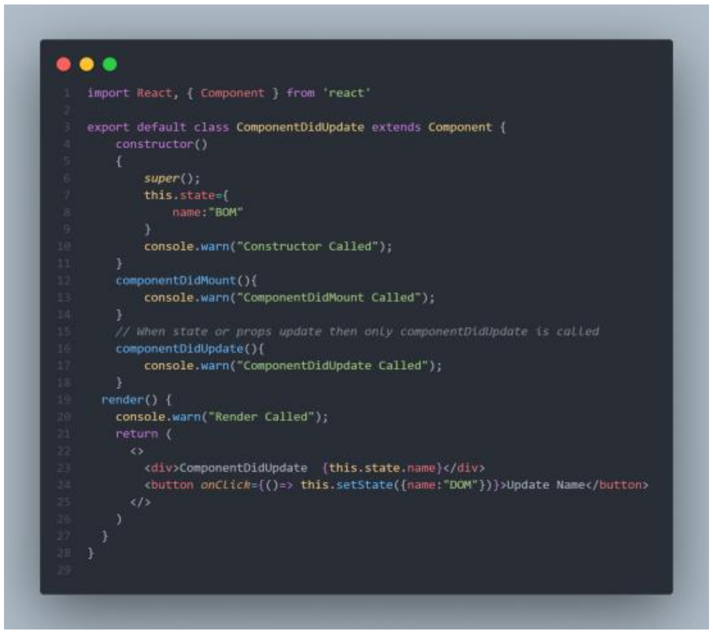
```jsx
import React, { Component } from 'react'

export default class ComponentDidMount extends Component {
    constructor()
    {
        super();
        // 1
        // console.warn("Constructor Method Called");
        this.state = {
            name:"BOM"
        }
    }
    // 3
    componentDidMount()
    {
        console.warn("ComponentDidMount Method Called");
    }
  render() {
    // 2
    console.warn("Render Method Called");
    return (
      <>
        <div>ComponentDidMount {this.state.name}</div>
        <button onClick={()=>this.setState({name:"DOM"})}>Update Name</button>
      </>
    )
  }
}
```



ComponentDidMount BOM
Update Name

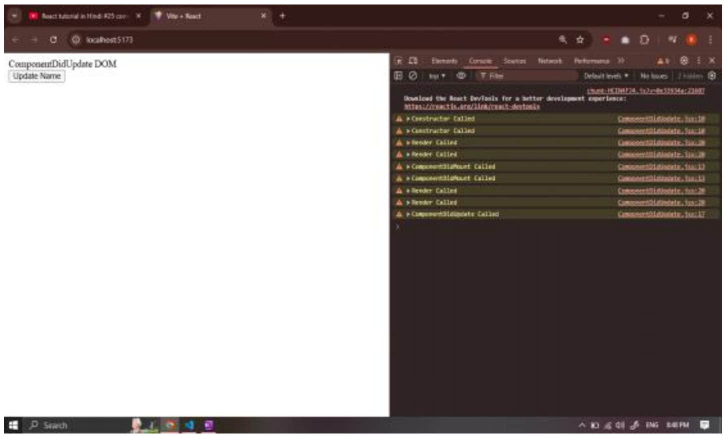
```
Download the React DevTools for a better development experience:
https://reactjs.org/link/react-devtools
▸ Constructor Method Called            [illegible]
▸ Constructor Method Called            [illegible]
▸ Render Method Called                 [illegible]
▸ Render Method Called                 [illegible]
▸ ComponentDidMount Method Called      [illegible]
▸ ComponentDidMount Method Called      [illegible]
▸ Render Method Called                 [illegible]
▸ Render Method Called                 [illegible]
▸ Render Method Called                 ComponentDidMount.jsx:18
▸ Render Method Called                 ComponentDidMount.jsx:18
▸ ComponentDidMount Method Called      ComponentDidMount.jsx:16
▸ ComponentDidMount Method Called      ComponentDidMount.jsx:16
```

ComponentDIdUpdate Method Called :



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

export default class ComponentDidUpdate extends Component {
    constructor()
    {
        super();
        this.state={
            name:"BOM"
        }
        console.warn("Constructor Called");
    }
    componentDidMount(){
        console.warn("ComponentDidMount Called");
    }
    // When state or props update then only componentDidUpdate is called
    componentDidUpdate(){
        console.warn("ComponentDidUpdate Called");
    }
  render() {
    console.warn("Render Called");
    return (
      <>
        <div>ComponentDidUpdate  {this.state.name}</div>
        <button onClick={()=> this.setState({name:"DOM"})}>Update Name</button>
      </>
    )
  }
}
```

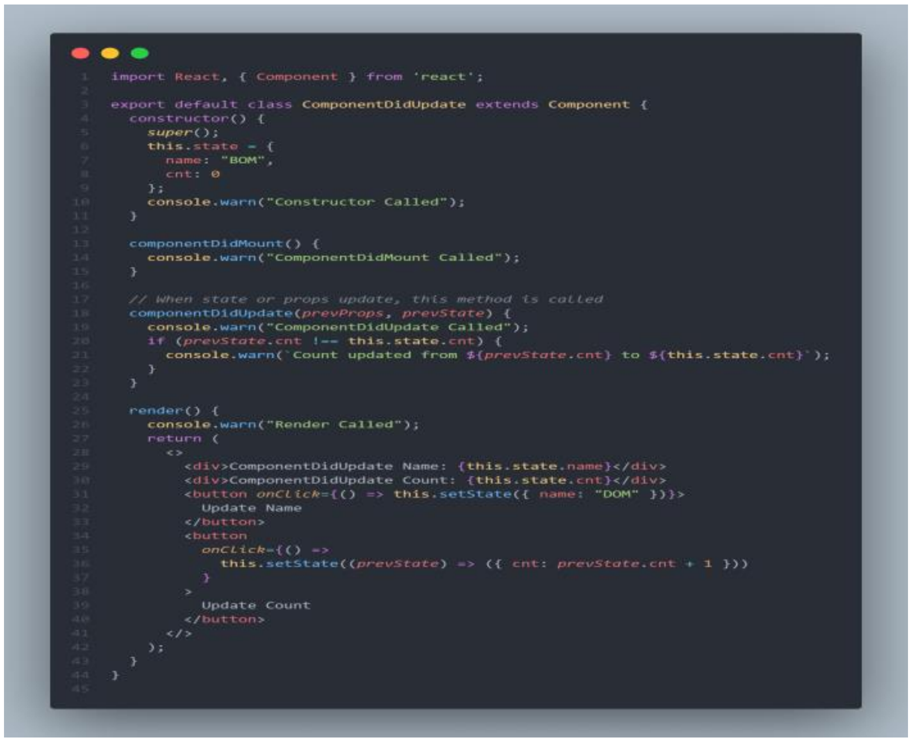
O/P :-



```
ComponentDidUpdate DOM
Update Name
```

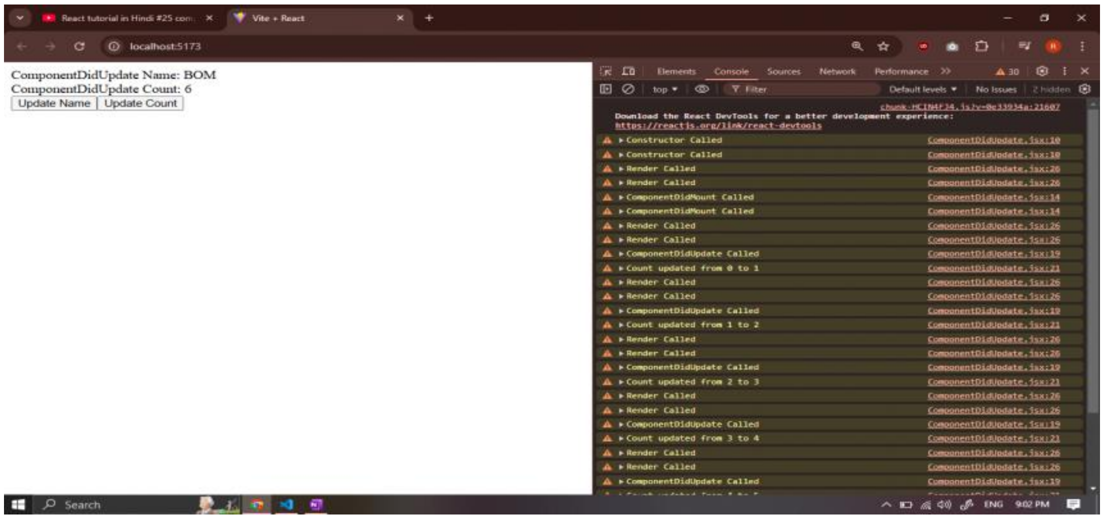
```
Download the React DevTools for a better development experience:
https://reactjs.org/link/react-devtools
▶ Constructor Called                    ComponentDidUpdate.jsx:10
▶ Constructor Called                    ComponentDidUpdate.jsx:10
▶ Render Called                         ComponentDidUpdate.jsx:20
▶ Render Called                         ComponentDidUpdate.jsx:20
▶ ComponentDidMount Called              ComponentDidUpdate.jsx:13
▶ ComponentDidMount Called              ComponentDidUpdate.jsx:13
▶ Render Called                         ComponentDidUpdate.jsx:20
▶ Render Called                         ComponentDidUpdate.jsx:20
▶ ComponentDidUpdate Called             ComponentDidUpdate.jsx:17
```

State :



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

export default class ComponentDidUpdate extends Component {
  constructor() {
    super();
    this.state = {
      name: "BOM",
      cnt: 0
    };
    console.warn("Constructor Called");
  }

  componentDidMount() {
    console.warn("ComponentDidMount Called");
  }

  // When state or props update, this method is called
  componentDidUpdate(prevProps, prevState) {
    console.warn("ComponentDidUpdate Called");
    if (prevState.cnt !== this.state.cnt) {
      console.warn(`Count updated from ${prevState.cnt} to ${this.state.cnt}`);
    }
  }

  render() {
    console.warn("Render Called");
    return (
      <>
        <div>ComponentDidUpdate Name: {this.state.name}</div>
        <div>ComponentDidUpdate Count: {this.state.cnt}</div>
        <button onClick={() => this.setState({ name: "DOM" })}>
          Update Name
        </button>
        <button
          onClick={() =>
            this.setState((prevState) => ({ cnt: prevState.cnt + 1 }))
          }
        >
          Update Count
        </button>
      </>
    );
  }
}
```

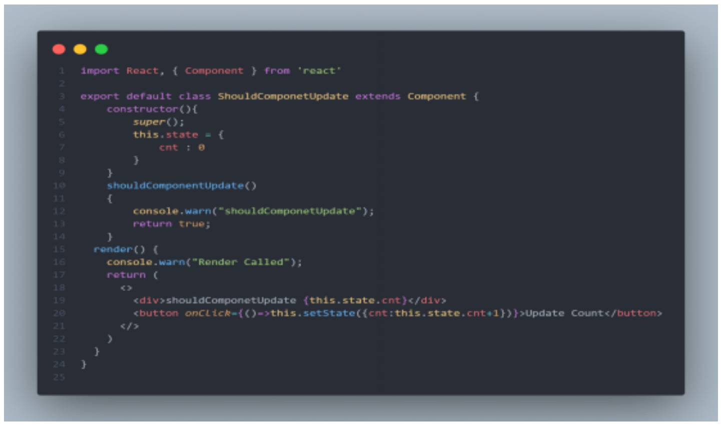
o/P : -



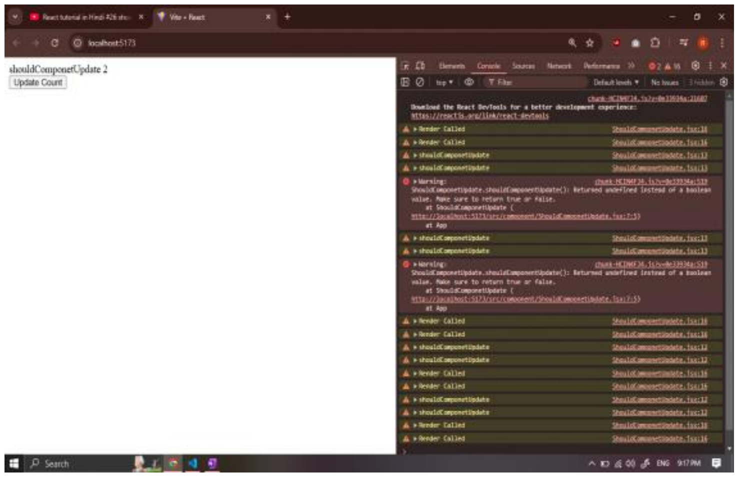
```
ComponentDidUpdate Name: BOM
ComponentDidUpdate Count: 6
Update Name  Update Count
```

```
Download the React DevTools for a better development experience:
https://reactjs.org/link/react-devtools
Constructor Called                ComponentDidUpdate.jsx:10
Constructor Called                ComponentDidUpdate.jsx:10
Render Called                     ComponentDidUpdate.jsx:26
Render Called                     ComponentDidUpdate.jsx:26
ComponentDidMount Called          ComponentDidUpdate.jsx:14
ComponentDidMount Called          ComponentDidUpdate.jsx:14
Render Called                     ComponentDidUpdate.jsx:26
Render Called                     ComponentDidUpdate.jsx:26
ComponentDidUpdate Called         ComponentDidUpdate.jsx:19
Count updated from 0 to 1         ComponentDidUpdate.jsx:21
Render Called                     ComponentDidUpdate.jsx:26
Render Called                     ComponentDidUpdate.jsx:26
ComponentDidUpdate Called         ComponentDidUpdate.jsx:19
Count updated from 1 to 2         ComponentDidUpdate.jsx:21
Render Called                     ComponentDidUpdate.jsx:26
Render Called                     ComponentDidUpdate.jsx:26
ComponentDidUpdate Called         ComponentDidUpdate.jsx:19
Count updated from 2 to 3         ComponentDidUpdate.jsx:21
Render Called                     ComponentDidUpdate.jsx:26
Render Called                     ComponentDidUpdate.jsx:26
ComponentDidUpdate Called         ComponentDidUpdate.jsx:19
Count updated from 3 to 4         ComponentDidUpdate.jsx:21
Render Called                     ComponentDidUpdate.jsx:26
Render Called                     ComponentDidUpdate.jsx:26
ComponentDidUpdate Called         ComponentDidUpdate.jsx:19
```

shouldComponentUpdate :
Use for stop re-rendring or we can say stop execution of componentDidUpdate()



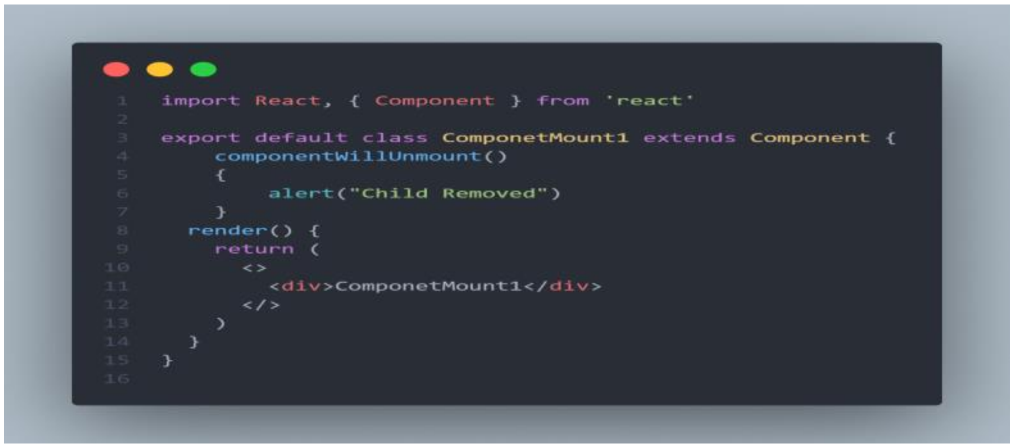
```jsx
import React, { Component } from 'react'

export default class ShouldComponetUpdate extends Component {
    constructor(){
        super();
        this.state = {
            cnt : 0
        }
    }
    shouldComponentUpdate()
    {
        console.warn("shouldComponetUpdate");
        return true;
    }
  render() {
    console.warn("Render Called");
    return (
      <>
        <div>shouldComponetUpdate {this.state.cnt}</div>
        <button onClick={()=>this.setState({cnt:this.state.cnt+1})}>Update Count</button>
      </>
    )
  }
}
```



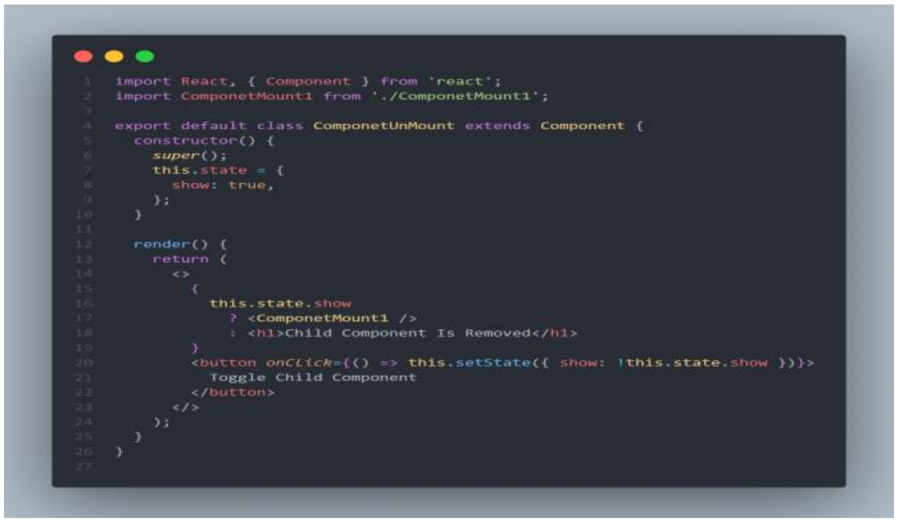
```
shouldComponetUpdate 2
Update Count
```

```
Download the React DevTools for a better development experience:
https://reactjs.org/link/react-devtools
Render Called                     ShouldComponetUpdate.jsx:16
Render Called                     ShouldComponetUpdate.jsx:16
shouldComponetUpdate              ShouldComponetUpdate.jsx:13
shouldComponetUpdate              ShouldComponetUpdate.jsx:13
Warning:
ShouldComponetUpdate.shouldComponentUpdate(): Returned undefined instead of a boolean
value. Make sure to return true or false.
    at ShouldComponetUpdate (
http://localhost:5173/src/component/ShouldComponetUpdate.jsx:7:5)
    at App
shouldComponetUpdate              ShouldComponetUpdate.jsx:13
shouldComponetUpdate              ShouldComponetUpdate.jsx:13
Warning:
ShouldComponetUpdate.shouldComponentUpdate(): Returned undefined instead of a boolean
value. Make sure to return true or false.
    at ShouldComponetUpdate (
http://localhost:5173/src/component/ShouldComponetUpdate.jsx:7:5)
    at App
Render Called                     ShouldComponetUpdate.jsx:16
Render Called                     ShouldComponetUpdate.jsx:16
shouldComponetUpdate              ShouldComponetUpdate.jsx:12
shouldComponetUpdate              ShouldComponetUpdate.jsx:12
Render Called                     ShouldComponetUpdate.jsx:16
Render Called                     ShouldComponetUpdate.jsx:16
shouldComponetUpdate              ShouldComponetUpdate.jsx:12
shouldComponetUpdate              ShouldComponetUpdate.jsx:12
Render Called                     ShouldComponetUpdate.jsx:16
Render Called                     ShouldComponetUpdate.jsx:16
```

componetWillUnmount :



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

export default class ComponetMount1 extends Component {
    componentWillUnmount()
    {
        alert("Child Removed")
    }
  render() {
    return (
      <>
        <div>ComponetMount1</div>
      </>
    )
  }
}
```



```
import React, { Component } from 'react';
import ComponetMount1 from './ComponetMount1';

export default class ComponetUnMount extends Component {
  constructor() {
    super();
    this.state = {
      show: true,
    };
  }

  render() {
    return (
      <>
        {
          this.state.show
            ? <ComponetMount1 />
            : <h1>Child Component Is Removed</h1>
        }
        <button onClick={() => this.setState({ show: !this.state.show })}>
          Toggle Child Component
        </button>
      </>
    );
  }
}
```