

```
import React from "react";
```

```
class App extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = { name: "John" };  
    console.log("Inside constructor");  
  }  

```

```
  UNSAFE_componentWillMount() {  
    console.log("ComponentWillMount");  
  }  

```

```
  shouldComponentUpdate(nextProps, nextState) {  
    console.log("Inside ShouldComponentUpdate");  
    console.log("Next State =>", nextState);  
    console.log("Current State =>", this.state);  
    return true;  
  }  

```

```
  componentDidUpdate() {  
    console.log("Inside ComponentDidUpdate");  
  }  

```

```
  componentDidMount() {  
    console.log("Inside ComponentDidMount");  
  }  

```

```
  componentWillUnmount() {  
    console.log("Unmounting component");  
  }  

```

```
render() {  
  console.log("Render called for App.js");  
  return (  
    <div>  
      <h1>Hello, {this.state.name}</h1>  
      <button  
        onClick={() => {  
          this.setState({ name: "Jack" });  
        }}  
      >  
        John  
      </button>  
    </div>  
  );  
}  
}  
  
export default App;
```

```
import React from "react";
```

```
class App extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = { show: true, count: 0 };  
    console.log("Inside constructor");  
  }  

```

```
  delHeader = () => {  
    this.setState({ show: false });  
  };  

```

```
  componentDidMount() {  
    console.log("Inside ComponentDidMount");  
  }  

```

```
  componentDidUpdate() {  
    console.log("Inside ComponentDidUpdate");  
  }  

```

```
  render() {  
    console.log("Render called for App.js");  
    let myheader;  
    if (this.state.show) {  
      myheader = <Child />;  
    }  
    return (  
      <div>  
        <center>  
          {myheader}  

```

```

    <button onClick={this.delHeader}>Delete Header</button>

    <hr />

    <h3>count : {this.state.count}</h3>

    <button
      onClick={() => this.setState({ count: this.state.count + 1 })}
    >
      Increment
    </button>
  </center>
</div>
);
}
}

```

```

class Child extends React.Component {
  componentWillUnmount() {
    console.log("Unmounting component");
  }
  render() {
    return <h1>Hello World!</h1>;
  }
}

```

```

export default App;

```