

React Lifecycle sample program

```
import './App.css';
import React from 'react';
//To hide warnings in console
console.warn = () => {}

class App extends React.Component {
  constructor(props) {
    super(props);
    this.state = {hello: "React"};
    this.changeState = this.changeState.bind(this);
  }
  render() {
    return (
      <div>
        <h1>React component's Lifecycle</h1>
        <h3>Hello {this.state.hello}</h3>
        <button onClick = {this.changeState}>Click Here!</button>
      </div>
    );
  }

  //We can use these builtin functions to manage the flow of components

  //This method allows us to execute the React code when
  //the component gets loaded or mounted in the DOM
  componentWillMount () {
    console.log('Component will mount')
  }

  //This method allows us to execute the React code when
  //the component is already placed in the DOM
  componentDidMount() {
    console.log('Component did mount')
  }

  changeState(){
    //setState() will updates the component state and instruct to rerender the page.
    this.setState({hello:"Welcome to React LIfecycle"});
  }

  //The method invoked before our mounted React component receives new prop
  componentWillReceiveProps(newProps) {
    console.log('Component will recieve props')
  }

  //Makes the component re-render only when there is a change in state or props
  //and that change will affect the output
  shouldComponentUpdate(newProps, newState) {
    return true;
  }
}
```

```

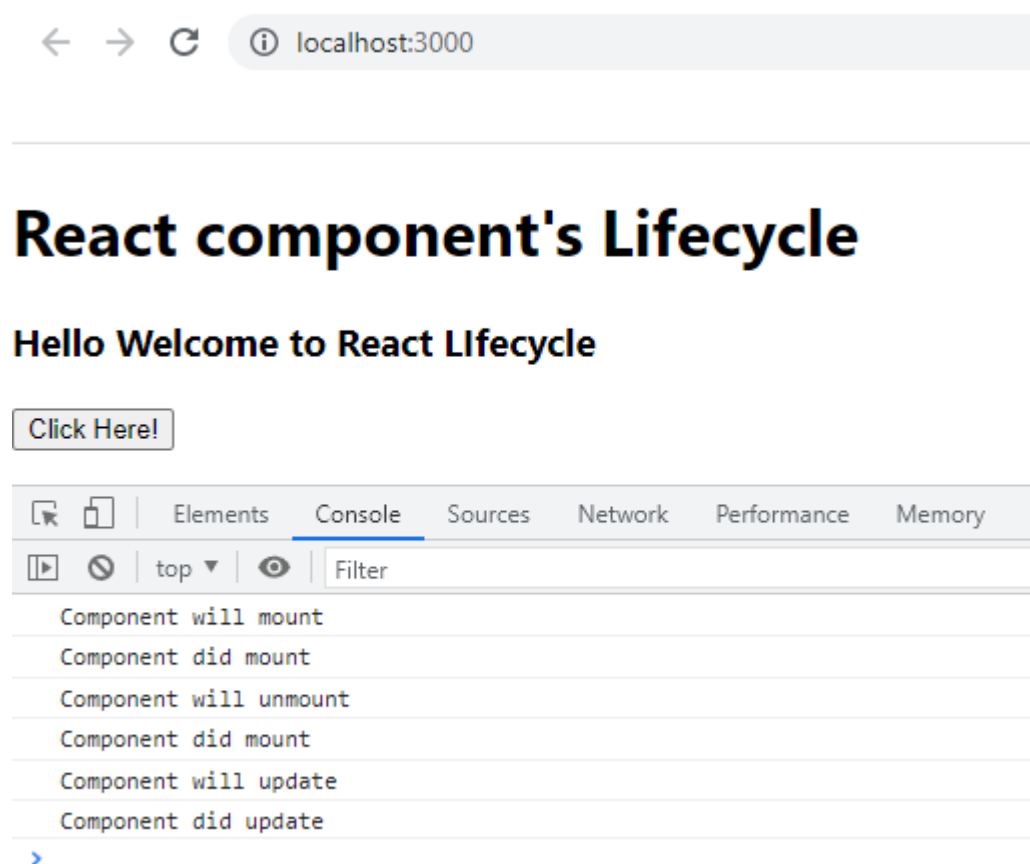
    //called before the component is updated or when the state or props passed to the
    component changes
    componentWillUpdate(nextProps, nextState) {
        console.log('Component will update');
    }

    //This method invoked immediately after updating occurs
    componentDidUpdate(prevProps, prevState) {
        console.log('Component did update')
    }

    //This method allows us to execute the React code when the component gets
    destroyed or
    //unmounted from the DOM
    componentWillUnmount() {
        console.log('Component will unmount')
    }
}
export default App;

```

Output



← → ↻ ⓘ localhost:3000

React component's Lifecycle

Hello Welcome to React Lifecycle

Click Here!

Elements	Console	Sources	Network	Performance	Memory
<div> <div>▶</div> <div>⌵</div> <div>top ▼</div> <div>👁</div> <div>Filter</div> </div>					
Component will mount					
Component did mount					
Component will unmount					
Component did mount					
Component will update					
Component did update					

>