# React Lifecycle Assignment

## Explain the Need and Benefits of Component Lifecycle

The component lifecycle in React defines a series of methods that are invoked at different stages of a component's existence. These stages include mounting, updating, and unmounting. Understanding and utilizing the lifecycle methods allows developers to hook into key points in a component's life, enabling tasks such as:  
  
- Fetching data when the component mounts  
- Optimizing performance during updates  
- Cleaning up resources before a component unmounts  
- Handling errors gracefully within a component tree  
  
Benefits:  
- Efficient resource management  
- Enhanced control over UI rendering behavior  
- Better debugging and error handling  
- Smooth integration with APIs or subscriptions

## Identify Various Lifecycle Hook Methods

Lifecycle methods in React class components can be categorized into the following phases:  
  
Mounting (component is being inserted into the DOM):  
- constructor()  
- static getDerivedStateFromProps()  
- render()  
- componentDidMount()  
  
Updating (component is re-rendered due to state or prop changes):  
- static getDerivedStateFromProps()  
- shouldComponentUpdate()  
- render()  
- getSnapshotBeforeUpdate()  
- componentDidUpdate()  
  
Unmounting (component is removed from the DOM):  
- componentWillUnmount()  
  
Error Handling:  
- componentDidCatch()  
- static getDerivedStateFromError()

## List the Sequence of Steps in Rendering a Component

When a component is mounted (initial rendering):  
1. constructor()  
2. static getDerivedStateFromProps()  
3. render()  
4. componentDidMount()  
  
When a component is updated (re-rendering):  
1. static getDerivedStateFromProps()  
2. shouldComponentUpdate()  
3. render()  
4. getSnapshotBeforeUpdate()  
5. componentDidUpdate()  
  
When a component is unmounted (removed from the DOM):  
1. componentWillUnmount()  
  
In case of an error in a child component:  
1. static getDerivedStateFromError()  
2. componentDidCatch()