## **LIFECYCLE METHODS**

### **Introduction**

Now that you’re done with your first project, you are on your way to becoming advanced with React. So let’s talk about some more advanced topics. The first will be lifecycle methods. This is an incredibly important topic, so pay close attention and make sure you understand everything.

Lifecycle methods are special methods built-in to React, used to operate on components throughout their duration in the DOM. For example, when the component mounts, renders, updates, or unmounts. You already know the most important lifecycle method, the render method. Soon you will know a few more, with a good understanding of when to use them and what tasks should be done within them.

One important thing to notice is that lifecycle methods can only be used in class components. In functional components we have other options. But more on that in the next lesson. Let’s begin.

### **Learning Outcomes**

* Understanding the lifecycle of a component
* Knowing the most important lifecycle methods
* Understanding which tasks should be done in the different life cycles

### **Lifecycle**

As stated in the introduction, a component’s lifecycle is the sequence of stages an instance of a component goes through in the DOM (or virtual DOM in React’s case). Let me give you a quick example:

1. Creation of the component (componentDidMount)
2. Render of the component (render)
3. (Optional) Update of the component (componentDidUpdate)
4. Death of the component (componentWillUnmount)

A component can be in different states of its “life”, and in React we can use those states to handle certain tasks.

### **Render**

We are already familiar with the render method in React, each class component needs to contain a render method, and it is fairly simple to understand. The render method contains all the logic your component should display on the screen. It might also contain a null value, if you don’t want to show anything. (Conditional rendering! Look it up!)

### **ComponentDidMount**

As the name already suggests, this method is run when the component is mounted, meaning when it is inserted in the DOM tree.

Common tasks that are done in this method are:

* Connect to web APIs or JavaScript frameworks.
* Set Timers using setTimeout or setInterval.
* Add event listeners.

### **ComponentDidUpdate**

This method is not called for the initial render, but any other time when the component updates. This method is a great place to work and operate on the DOM when the component has updated. For example you could do network requests. Just make sure to compare it to the current props.

### **ComponentWillUnmount**

This is the last lifecycle method, which will be called when the component is removed from the DOM tree. Usually you would do cleanup tasks in this method. This often includes cleaning up what you added in the componentDidMount method.

A few examples are:

* Remove event listeners
* Cancelling network requests
* Other cleanup routines

### **Assignment**

1. Read [this article](https://programmingwithmosh.com/javascript/react-lifecycle-methods/) for a great overview of lifecycle methods in React.
2. The React documentation is always a good source as it’s well structured. In [this article](https://reactjs.org/docs/react-component.html) you can read more about lifecycle methods from the people who made it!
3. Code along with [this video](https://www.youtube.com/watch?v=m_mtV4YaI8c), it gives you practical examples about how those methods work and when React calls them.

### **Additional Resources**

* [Here](https://blog.bitsrc.io/react-16-lifecycle-methods-how-and-when-to-use-them-f4ad31fb2282) is another great resource for understanding React Lifecycle methods