**What is Life Cycle Method**

* Each component in React has a lifecycle which you can monitor and manipulate during its three main phases.
* The three phases are: Mounting, Updating, and Unmounting.

**Mounting:**

* Mounting means putting elements into the DOM.
* React has four built-in methods that gets called, in a order, when mounting a component:
* The constructor() method is called before anything else, when the component is initiated, and it is used to set up the initial state and other initial values. It will take props as argument. We also add super() inside it to initiate the parent's constructor method and inherit the parent class’s property and methods.
* The getDerivedStateFromProps() method is called right before rendering the element(s) in the DOM. This is the natural place to set the state object based on the initial props. It takes state as an argument, and returns an object with changes to the state.
* The render() method is required, and is the method that actually outputs the HTML to the DOM.
* The componentDidMount() method is called after the component is rendered . Here we run those statements that tell us that the component is already placed in the DOM

**Updating:**

* Updating means Updating Components into the DOM.
* A component is updated whenever there is a change in the component's state or props.
* React has five built-in methods that gets called, in a order, when Updating a component:
* getDerivedStateFromProps is the first method that is called when a component gets updated. This is the natural place to set the state object based on the initial props.
* In the shouldComponentUpdate() method you can return a Boolean value that specifies whether React should continue with the rendering or not. The default value is true.
* The render() method is of course called when a component gets updated, it has to re-render the HTML to the DOM, with the new changes.
* In the getSnapshotBeforeUpdate() method you have access to the props and state *before* the update, meaning that even after the update, you can check what the values were *before* the update.
* If the getSnapshotBeforeUpdate() method is present, you should also include the componentDidUpdate() method, otherwise you will get an error. The componentDidUpdate method is called after the component is updated in the DOM.

**Unmounting:**

* Unmounting means Removing Components from the DOM. It calls only one method named componentWillUnmount()