Lifecycle Methods

Mounting	When an instance of a component is being created and inserted into the DOM
Updating	When a component is being re-rendered as a result of changes to either its props or state
Unmounting	When a component is being removed from the DOM
Error Handling	When there is an error during rendering, in a lifecycle method, or in the constructor of any child component

Lifecycle Methods

| Mounting | *constructor, static getDerivedStateFromProps, render* and *componentDidMount* |

| Updating | *static getDerivedStateFromProps, shouldComponentUpdate, render, getSnapshotBeforeUpdate* and *componentDidUpdate* |

| Unmounting | *componentWillUnmount* |

| Error Handling | *static getDerivedStateFromError* and *componentDidCatch* |

Mounting Lifecycle Methods

| constructor(props) | A special function that will get called whenever a new component is created. |

Initializing state
Binding the event handlers

Do not cause side effects. Ex: HTTP requests

super(props)
Directly overwrite this.state

Mounting Lifecycle Methods

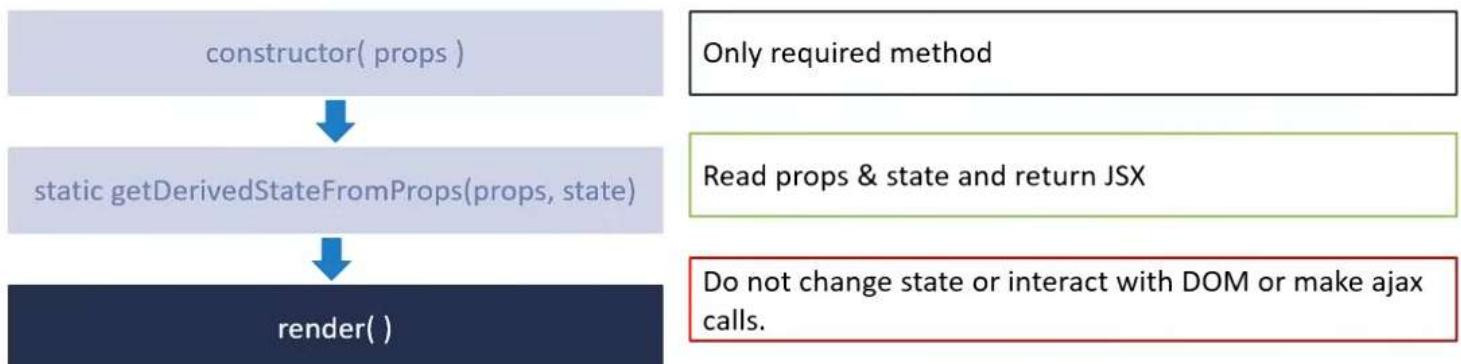|	
constructor(props)	When the state of the component depends on changes in props over time.
static getDerivedStateFromProps(props, state)	Set the state
	Do not cause side effects. Ex: HTTP requests
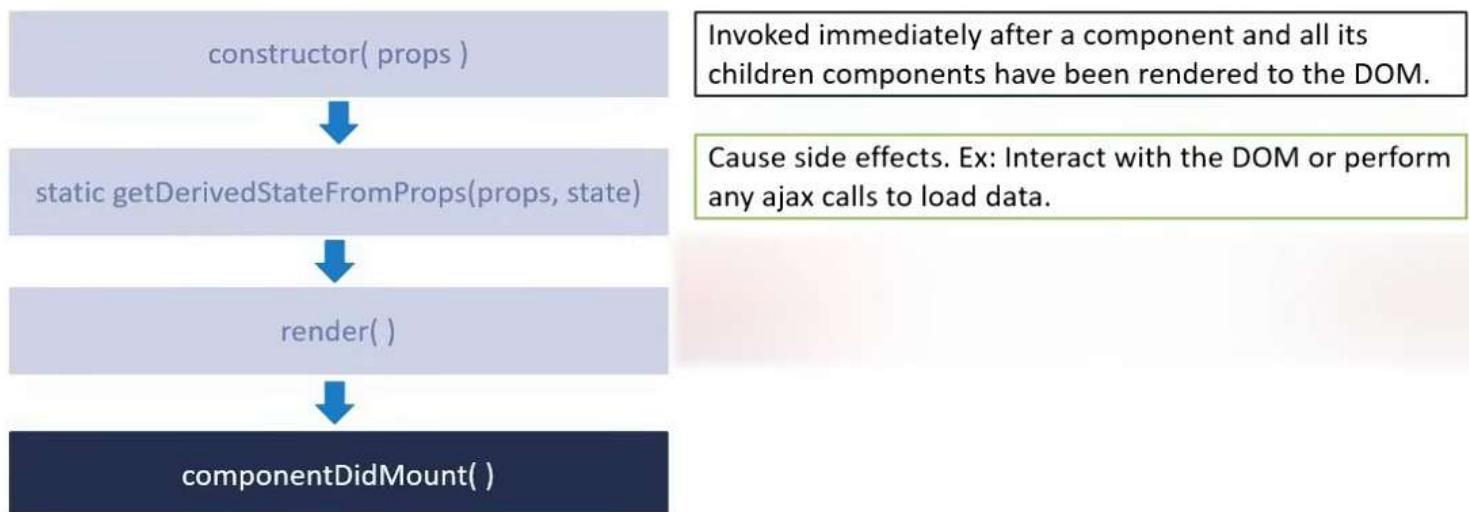
Mounting Lifecycle Methods

constructor(props)	Only required method
static getDerivedStateFromProps(props, state)	Read props & state and return JSX
render( )	Do not change state or interact with DOM or make ajax calls.

Mounting Lifecycle Methods

constructor(props)

static getDerivedStateFromProps(props, state)

render( )

componentDidMount( )

Invoked immediately after a component and all its children components have been rendered to the DOM.

Cause side effects. Ex: Interact with the DOM or perform any ajax calls to load data.

Updating Lifecycle Methods

static getDerivedStateFromProps(props, state)

Method is called every time a component is re-rendered

Set the state

Do not cause side effects. Ex: HTTP requests

Updating Lifecycle Methods

static getDerivedStateFromProps(props, state)		Dictates if the component should re-render or not
shouldComponentUpdate(nextProps, nextState)		Performance optimization
		Do not cause side effects. Ex: HTTP requests Calling the setState method

Updating Lifecycle Methods

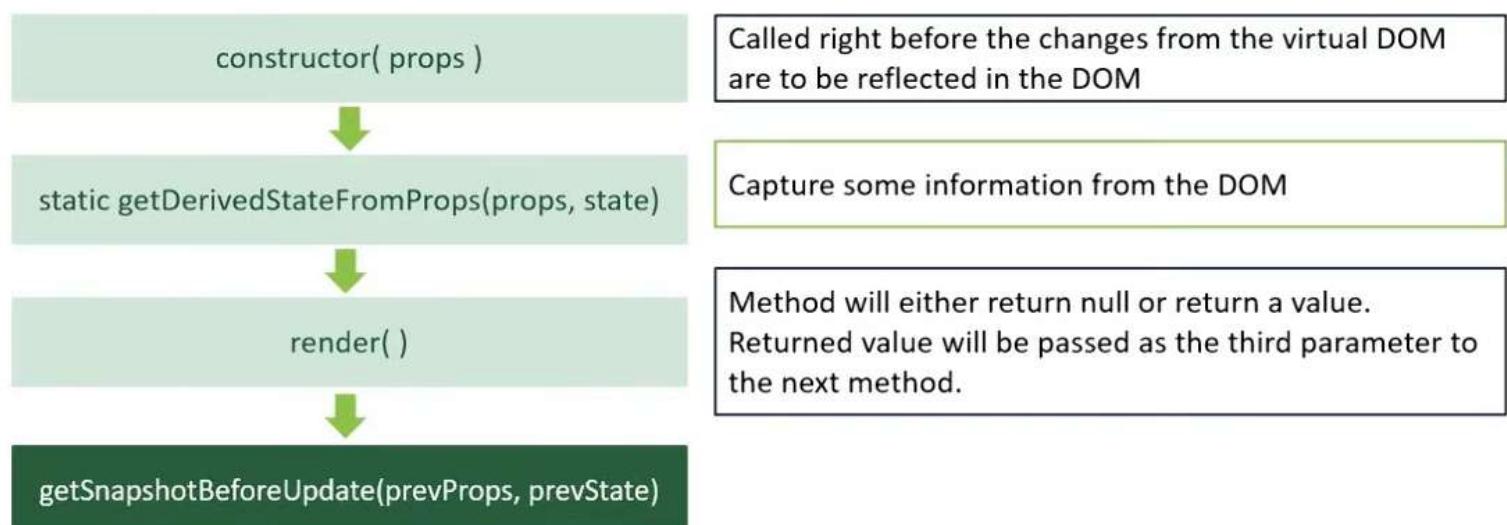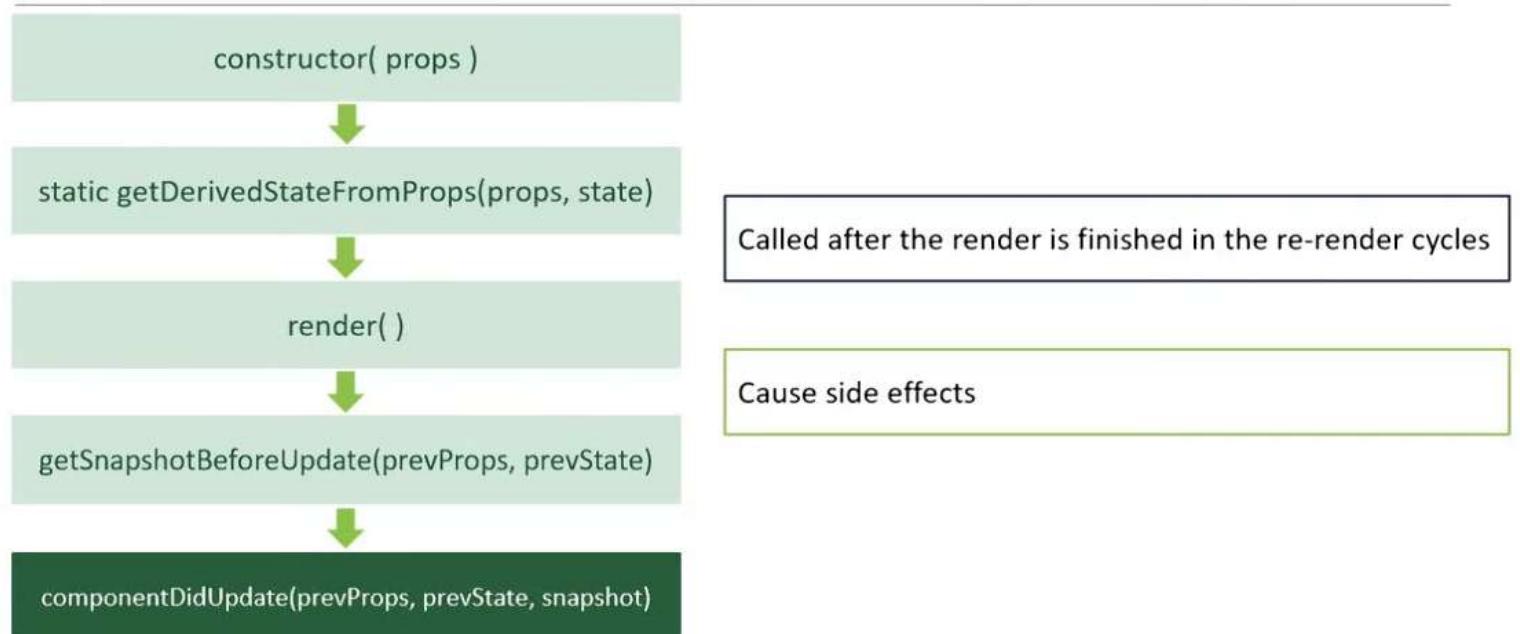| static getDerivedStateFromProps(props, state)	Only required method
shouldComponentUpdate(nextProps, nextState)	Read props & state and return JSX
render( )	Do not change state or interact with DOM or make ajax calls.

Updating Lifecycle Methods

constructor(props)	Called right before the changes from the virtual DOM are to be reflected in the DOM
↓	
static getDerivedStateFromProps(props, state)	Capture some information from the DOM
↓	
render( )	Method will either return null or return a value. Returned value will be passed as the third parameter to the next method.
↓	
getSnapshotBeforeUpdate(prevProps, prevState)	

Updating Lifecycle Methods

```
constructor( props )
        |
        v
static getDerivedStateFromProps(props, state)
        |
        v
render( )
        |
        v
getSnapshotBeforeUpdate(prevProps, prevState)
        |
        v
componentDidUpdate(prevProps, prevState, snapshot)
```

Called after the render is finished in the re-render cycles

Cause side effects

Unmounting Phase Method

| componentWillUnmount( ) | Method is invoked immediately before a component is unmounted and destroyed. |

Cancelling any network requests, removing event handlers, cancelling any subscriptions and also invalidating timers.

Do not call the setState method.

Error Handling Phase Methods

static getDerivedStateFromError(error)

componentDidCatch(error, info)

When there is an error either during rendering, in a lifecycle method, or in the constructor of any child component.