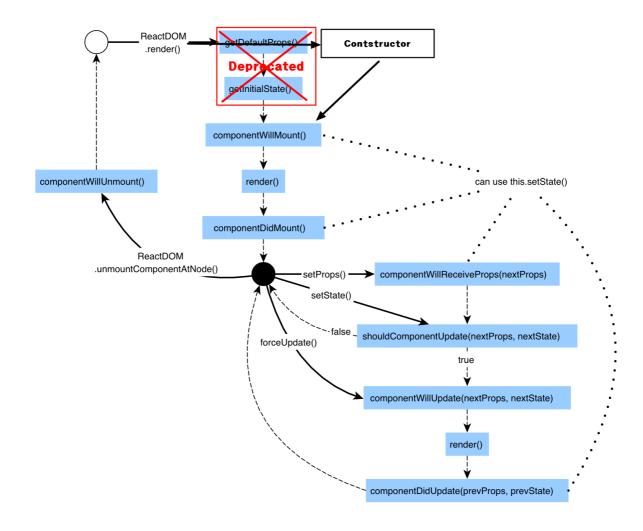
4. React_LifeCycle_Class <중 요!> (March 8th, 21)

A. Class형 컴포넌트에서의 LifeCycle



class형 컴포넌트의 LifeCycle에서, 화면을 새로고침하면 class형 컴포넌트의 render() 함수가 호출되기 전후로 각각componentWillMount(), componentDidMount() 함수 를 호출한다.

```
40 class ClassComp extends React.Component{
 41
      state = {
 42
        number:this.props.initNumber,
 43
        date:(new Date()).toString()
 44
 45
      componentWillMount(){
 46
        console.log('%cclass => componentWillMount', classStyle);
 47
 48
      componentDidMount(){
 49
        console.log('%cclass => componentDidMount', classStyle);
 50
                                  component DidMount
 51
      render(){
                                           TrassStyle);
 52
        console.log('%cclass => render'
 53
        return (
          <div className="container">
 54
{} Line 49, Column 45
    Console
              What's New
                             Request blocking
                                                 Remote devices
                                       Filter
|
    0
          top
                                  0
                                                                 Custor
  numberState \triangleright (2) [2, f]
  class => componentWillMount
  class => render
  class => componentDidMout
>
```

화면을 **새로고침** 했을 때, render되기 전에 돼야하는 일은 **componentWillMount()라는 method를 구현해서 내용을 가져오고, 그 내용을 render한 후에 후처리 해야할 일이 있다 면, componentDidMount()를 통해서 해결**한다.

그 다음, 예를 들어 사용자가 random 버튼을 눌러, state에 변화가 생기면, 해당하는 컴포 넌트의 render함수가 호출되는데, 이 때 컴포넌트가 업데이트 돼야하는지를 확인하는 shouldComponentUpdate가 호출되고, shouldComponentUpdate가 true를 반환 하면 componentWillUpdate가 호출된 후 render가 호출되고, 그 후에 componentDidUpdate가 차례로 호출된다.

<u>App.js:56</u>
<u>App.js:60</u>
<u>App.js:66</u>
<u>App.js:63</u>

이렇듯, class 컴포넌트의 동작 순서가 정해져있으므로, 각 상황 마다 원하는 동작을 구현하고 싶다면, 그 상황에 맞는 method 의 내용을 설정함으로써 해결할 수 있다.