React (Day-8)

Class based components

Almost similar to function based components, but are traditional and old.

```
=) import React from 'react';
   class UserClass extends React.Component {
      constructor(props) {
          super(props)
      }
      render() {
          return (
              <div>
                  <h1>{this.props.name}</h1>
              </div>
          );
      }
   }
```

↳ to let react know that this is a component

↳ to load all props inside 'this' keyword

{ basically our component inherits from React.Component }

{ render() is the method used to return JSX in CBC }

How to use & update state?

→ State is created when component is created, so what better way than creating state in constructor.

```
constructor(props) {
    super(props)
    this.state = {
        count: 0,
        count2: 1
    }
render() {
    return (
        <div>
            <h3> count: {this.state.count} </h3>
            <button onClick={() => { this.setState({
                            count: this.state.count+1
            })}> </button> </div>); }
```

← { All the states of component are declared in 'this.state' object }

→ this.state can be used to access any state of the component, where 'state' is a keyword.

Ex:. this.state.count (or) this.state.count2.

→ this.setState() can be used to update the state where setState() is a function (keyword) used to update one or all states of the component.

Ex:1
```
this.setState({
        count: this.state.count+1
});
```

(or)

Ex:2
```
this.setState({
        count: this.state.count+1
        count2: this.state.count2+1
});
```

Note: while updating state like Ex:1, only count is updated, while other states remain unaffected.

Note: There might be hardly any case where we would update multiple states at same time like Ex:2, but yeah, you can do that too.

Might be handy while handling forms where you need to update all the input field states on click of submit button.

But we won't be using class based components anyway, this is just for knowledge & interview purpose.

componentDidMount() :

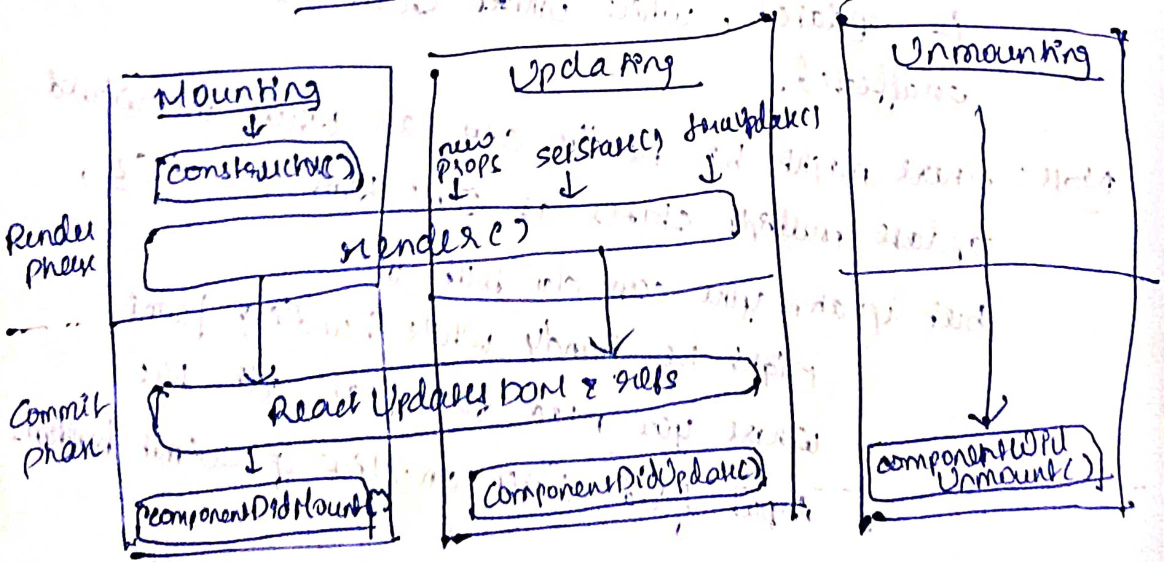
It is a function/method in CBC which is called after the instantiation and rendering phase of the class component.

- Basically it is used to make API calls, where we call the API just after the component is mounted (rendered) on to the page.

- It is like useEffect(()=>{
 },[]);

- But the catch here is, if there are multiple components, then the render phase of all the sibling components is batched together for better efficiency, and after that, componentDidMount() is called.

React Class Based Component Lifecycle



Mounting :

- constructor()
- Render (dummy)
 <HTML Dummy>
- ComponentDidMount()
 <API call>
 <setState>

Updating (No constructor call in updates)

- render(api data)
- <HTML (new API data)>
- ComponentDidUpdate()