

# Day 2-3. Props & State (Feb 24, 21)

Q. State란 무엇인가?

A. Component 내부적으로 사용되는 것들, 혹은 Props에 따라서 Component를 실제로 구현하는 내부의 정보. (Props와 구별된다.)

예시를 보면서 이해해보자.

```
class App extends Component{
  render() {
    return(
      <div className="App">
        <Subject title="WEB" sub="world wide web!"></Subject>npx
        <TOC></TOC>
        <Content></Content>
      </div>
    );
  }
}
```

위 코드의 "WEB", "world wide web!" 을 state로 만들고, 그 state값을 Subject component에 props로 전달한다.

Component가 실행될 때, constructor라는 함수가 있다면, 그 함수가 가장 먼저 실행되면서, 해당 Component의 초기화를 수행한다.

예시. State를 적용하기 전의 Component 코드

```
App.js

class App extends Component{
  render() {
    return(
      <div className="App">
        <Subject title="WEB" sub="world wide web!"></Subject>npx
        <Subject title="React" sub="For UI"></Subject>
        <TOC></TOC>
        <Content></Content>
      </div>
    );
  }
}
```

```
}  
}
```

## State를 적용한 후의 Component 코드

App.js

```
class App extends Component {  
  constructor(props) {  
    super(props);  
    this.state = {  
      subject: {title: 'WEB', sub: 'World Wide Web!'}  
    }  
  }  
  render() {  
    return(  
      <div className="App">  
        <Subject title={this.state.subject.title} sub={this.state.subject.sub}></Subject>npx  
        <Subject title="React" sub="For UI"></Subject>  
        <TOC></TOC>  
        <Content></Content>  
      </div>  
    );  
  }  
}
```