

1. Intro (March 7th, 21)

리액트가 컴포넌트를 만드는 두 가지 방법 : class vs function

function으로 컴포넌트를 만든다면 컴포넌트 내부에 state를 만들어 사용할 수 없고,

컴포넌트의 생성, 변경, 소멸의 이벤트인 life cycle API도 사용할 수 없다.

그래서 function은 상위 컴포넌트가 시키는 일만 처리하는 컴포넌트만 사용할 수 있다.

하지만, function에 hook 기능을 사용하면, function에서도 내부적으로 state을 다룰 수 있고,

컴포넌트의 life cycle에 따라서 해야할 작업을 정의할 수 있게 된다.

[https://www.youtube.com/watch?](https://www.youtube.com/watch?v=MZjT_P5HElw&list=PLuHgQVnccGMCEfBwnNGsJCQDlqSWI-edj&index=3)

[v=MZjT_P5HElw&list=PLuHgQVnccGMCEfBwnNGsJCQDlqSWI-edj&index=3](https://www.youtube.com/watch?v=MZjT_P5HElw&list=PLuHgQVnccGMCEfBwnNGsJCQDlqSWI-edj&index=3)

```
import logo from './logo.svg';
import './App.css';
import React from 'react';

function App() {
  return (
    <div className="App">
      Hello World
      <FuncComp></FuncComp>
      <ClassComp></ClassComp>
    </div>
  );
}

function FuncComp(){
  return(
    <div className="container">
      <h1>function style component</h1>
    </div>
  );
}

class ClassComp extends React.Component{
```

```
render(){                                #class 형식 기본 끝
  return (
    <div className="container">
      <h2>class style component</h2>
    </div>
  )
}

export default App;
```

페이지 내용물은 App.js에, 페이지 디자인은 App.css

```
App.css

.container{
  border:5px solid red;
  margin:5px;
  padding:5px;
}
```