

React Hook

그래서 뭐??

React Hook이 정확히 뭐하는거?

Class형 말고 Hook을 사용하는 이유

그럼 Hook은 어떻게 동작할까?

React Hook이 정확히 뭐하는거?

기존에는 클래스를 통해 컴포넌트의 state를 관리하였는데 Hook을 통해 함수형 컴포넌트에서도 state를 관리할 수 있게 해주는 것

이러면 무엇을 할 수 있게 해주는데?

React Hook 생기고 난 후 함수형에서 라이프 사이클을 사용할 수 있는 것

그럼 클래스 형이랑 다르게 뭐임?

함수형이 클래스형 처럼 스테이트 관리하고 라이프 사이클을 사용한다는것을 클래스 형처럼 변한거잖아
응 맞음, 근데 다르고 더 좋으니깐 사람들이 쓰겠지

```

class Example extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      count: 0
    };
  }

  render() {
    return (
      <div>
        <p>You clicked {this.state.count} times</p>
        <button onClick={() => this.setState({ count: this.state.count + 1 })}>
          Click me
        </button>
      </div>
    );
  }
}

```

<= 클래스형

클래스형 보다 함수형은 코드가 짧고 가독성이 높다

함수형 =>

```

import React, { useState } from 'react';

function Example() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>
        Click me
      </button>
    </div>
  );
}

```

React Hook 동작 방법

Hook은 Closure 기반

그럼 closure가 뭔데??

일단 Closure은 자바스크립트 개념 X

함수를 일급 객체로 취급하는 함수형 프로그래밍 언어에서 사용되는 중요한 특성

Closure 정의

클로저란 어떤 함수 A에서 선언한 변수 a를 참조하는 내부함수 B를 외부로 전달할 경우, A의 실행 컨텍스트가 끝난 뒤에도 변수 a가 사라지지 않는 현상

Closure 예시

```
let func1 = function(){  
    let a = 1;  
  
    let func2 = function (){  
        return ++a;  
    };  
    return func2; //func2 함수 자체를 반환  
};  
  
let func3 = func1();  
console.log(func3()); // 2  
console.log(func3()); // 3
```

func2는 a를 참조해서 ++a를 반환한다.
func1은 func2의 결과가 아닌 func2의 전체를 반환.
그럼 func3은 func2를 참조

func1의 실행 컨텍스트는 이미 종료된 상태인
데 어떻게 func2가 func1의 변수 a를 참조해?

가비지 컬렉터 때문 <= 이건 뭔데?

가비지 컬렉터는 어떤 값을 참조하는 변수가 하나라도 있다면 그 값은 수집 대상에 포함하지 않는다.

다음에 이어서..