**[ 리액트 컴포넌트 ]**

- 컴포넌트는 HTML요소를 반환하는 함수다!

**[ 특징 ]**

1. 컴포넌트는 독립적이고 재사용이 가능한 코드 집합

2. JS함수와 비슷하지만 HTML코드 반환이 필수라는 점이 다름!

3. 컴포넌트는 다음 2가지로 생성 가능함

1) 클래스형 컴포넌트

2) 함수형 컴포넌트

-> 클래스형 컴포넌트는 리액트 초 중기에 주로 사용되었으나...

React 16.8버전에서 Hooks와 함께는 더 이상 사용되지 않는다!

Hooks는 함수형 컴포넌트에서만 사용가능하다!

**[ 첫번째 컴포넌트 만들기! ]**

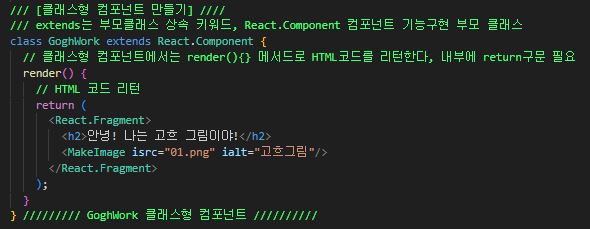
- 리액트 컴포넌트 이름은 반드시 **첫 글자를 대문자**로 만든다! (안 지키면 적용 안됨!!!)

[ 클래스 컴포넌트 ]: 클래스 컴포넌트에서는 extends React.Component 상속문이 포함 돼야함!

-> 컴포넌트에서도 메서드가 필요함, render() 메서드는 HTML을 반환함

(함수형 컴포넌트의 return 키워드를 사용할 수 있는 역할을 함!)

1. **클래스형 컴포넌트**



1. **함수형 컴포넌트**



[ 이미지 생성 공통 컴포넌트] : 변수를 설정하지않아도 비구조화 할당(구조분해할당)을 사용하여 전달 객체를 개별 변수 값으로 처리할 수 있다.

-> **비구조화 할당 방식(=구조분해할당방식)** =>> 쓰는 이유는 객체가 없기 때문

하나의 변수에는 전달된 값이 보낼 때 설정된 객체로 전달된다 {속성:값, 속성:값,...}

순서를 바꿔서 값을 보내도, 이름을 아무거나 써도 노 상관

1. function MakeImage(props){

return (<img src={"./images/"+props.isrc} alt={props.ialt} />) ;}

1. function MakeImage({ialt, isrc}){

return (<img src={"./images/"+isrc} alt={ialt} />) ;}

1. function MakeImage({ialt:ㅋㅋ, isrc:ㅎㅎ}){

return (<img src={"./images/"+ㅎㅎ} alt={ㅋㅋ} />); }

[ Props 사용하기 ]: props는 properties 에서 나온 말. 속성들...

즉, 변수에 값을 할당하여 전달하는 방법. 함수의 전달 값과 같고 속성으로 컴포넌트에 보낸다!

-> props라는 변수명은 마음대로 지을 수 있다

-> 컴포넌트에 {}를 사용하면 개별적인 속성 변수를 전달받을 수 있다(구조 분해 할당 방식)