

# 2020.12.06

## Spread Operator

...

스프레드 오퍼레이터

## Arrow Function

⇒

화살표 함수

1. 표현식에 결과값을 반환하는 표현식 본문에 쓰임
2. 상태 블록 본문에 쓰임

### 1. 표현식에 결과값을 반환하는 표현식 본문에 쓰이는 경우

map함수 : 자바스크립트에서 지원하는 배열객체 내장함수 input이 파라미터로 전달  
output은 새로운 배열

각 요소가 새로운 배열로 리턴 됨

```
let arr = [1, 2, 3, 4, 5];  
  
let twice = arr.map(v=>v*2);  
  
console.log(twice)
```

콘솔

[2, 4, 6, 8, 10] 이 반환 됨

arrow function이 없었다면

```
let twice = arr.map(function(val){  
  return val * 2;  
});
```

```
})
```

로 사용해야 함

## 2. 상태 블록 본문에 쓰이는 경우

```
let arr = [1, 2, 3, 4, 5];

let twice = arr.map(v=>{
  if (v % 2 == 0){
    console.log('even number')
  }else{
    console.log('odd number')
  }
});

console.log(twice)
```

콘솔

```
odd number
even number
odd number
even number
odd number
```

## Class

constructor 클래스 인스턴스를 생성하고 생성한 인스턴스를 초기화하는 역할을 함

인스턴스 : 클래스를 구성하기 위한 껍데기

new 연산자를 이용해 클래스 객체를 생성

## 상속

extends 로 상속 구현

super : super 로 감싸져 있는 부분은 부모 클래스에 있는 초기화률을 따른다는 것

함수는 오버라이드

```
class Person{
    constructor(region_, gender_) {
        this.region = region_;
        this.gender = gender_;
    }

    greetings(val = 'an-nyeong') {
        console.log(val);
    }
}

let korean = new Person('Korea', 'male')
//console.log(korean);

korean.gender = 'female'
//console.log(korean);

// korean.greetings();

class American extends Person {
    constructor(region_, gender_, language_) {
        super(region_, gender_);
        this.language = language_;
    }

    greetings(val = 'hello') {
        console.log(val);
    }
}

let american = new American('USA', 'female', 'English')
//console.log(american);

american.greetings();
```