

Booltest

객체에서도 참 거짓 판별이 된다.

js 는 인식되는 내용물이 있으면 true를 리턴함

```
function Vehicle () {  
    this.vendor = "Volvo"  
    this.mcu = "tms570"  
    this.price = 200000000  
}
```

```
Vehicle.prototype.getVendor = function () {  
    return this.vendor  
}
```

왜 이런식으로 하지?

자바처럼

vendor값이 필요한거면 그냥 Vehicle.vendor로 하면 안 되는건가?

꼭 funtion으로 묶어야 하는건가?

```
Vehicle String getVendor(){  
return vendor}이런건?
```

비교연산자 ==와 ===

==는 내용물만 비교함

===는 데이터 타입 까지 비교해줌

Ex)

1=="1" >true

1==="1" >false

true==1 >true //값이 있냐없냐

true===1 >false

null==undefined >true

null===undefined >false

문자열도 ==이 통합 .equal을 안 써도 됨

```
function EqualTest () {
  let name = "test"

  if (name) {
    console.log("true")
  } else {
    console.log("false")
  }

  if (name == "test") {
    console.log("equal")
  } else {
    console.log("not equal")
  }

  if (name == "go") {
    console.log("equal")
  } else {
    console.log("not equal")
  }

  if (name == null) {
    console.log("equal")
  } else {
    console.log("not equal")
  }
}
```

```
true
equal
not equal
not equal
not equal
not equal
>
```

Spread

1. let arrStringA=[..."123"] >> ["1","2","3"] 잘라줌

2. let arrA=[1,2,3]

let arrB=[...arrA,4,5]

console.log(arrB)

->[1,2,3,4,5]

java랑은 달리 array를 바로 인자로 받을 수 있음

java였다면

for(int i=0; i<=listA.size; i++){

listB.add(listA.get(i))

3. spread를 활용하면 배열에서 요소들을 효과적으로 빼낼 수 있다.

1) function arrprint(a,b,c){

console.log(a)

console.log(b)

console.log(c)

}

• let arrA=[1,2,3]

• arrprint(...arrA)

```
>> 1
    2
    3
```

```
2)function arrSpreadingPrint (a, b, c, ...remain) {
    console.log(a)
    console.log(b)
    console.log(c)
    console.log(remain)}
```

• let arrSpread = [1, 2, 3, 4, 5, 6, 7, 8, 9]

```
console.log(arrSpread)
arrPrint(...arrSpread)
arrSpreadingPrint(...arrSpread)
```

JS에서 class만들기

js내부에서는 어디서든 class의 선언이 가능하다



```
function ClassTest () {
  class Obj {
    constructor () {
      this.name = "test"
      this.major = "electronics"
    }
  }

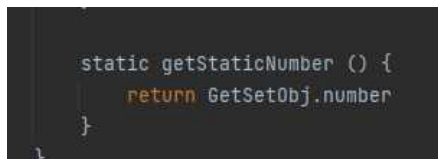
  let obj = new Obj()

  console.log(obj.name)
  console.log(obj.major)
}
```

constructor의 의미?

public ClassTest와 같은 건가?

전역변수



```
static getStaticNumber () {
  return GetSetObj.number
}
```