
JS

index.html

느낌표 탭으로 자동완성

main.js

<script src="./main.js"></script>

http://stevesouders.com/examples/rule-js-bottom.php

스크립트 태그를 어디에 넣을까?

```
alert('Welcome to JS');
```

정적이던 문서에 움직임을!

```
/*
    This
    is
    Javascript
*/

// alert('Welcome to JS');
```

```
document.write('<h1>Hello world!</h1>');
```

문서에 써줘 내용을

```
document.querySelector('h1');
```

크롬개발자도구 => 콘솔

```
document.querySelector('h1').innerText;
```

```
document.querySelector('h1').innerText = 'GoodBye world..';
```

```
var name = 'ssafy';
```

~~쑥과마늘을 먹기전~~

```
var a = 30

for (var a = 0 ; a < 10 ; a++){
    console.log(a)
}
console.log(a)
```

for문 아직 안했지만
예상먼저! => 콘솔창에서 확인

var는 function-scoped

```
function counter () {
    for(var i=0; i<10; i++) {
        console.log('i', i)
    }
}
counter()
console.log('after loop i is', i)
```

Uncaught ReferenceError: i is not defined

```
let name = 'ssafy';
```

```
const name = 'ssafy';
```

채신 문법!!!

```
let word = '외안되';
document.write(word);

word = ‘왜안돼';
document.write(word);
```

Variables(변수)는 이후에 재 할당될 때

```
const word = '외안되';
document.write(word);

word = '왜안돼';
document.write(word);
```

Constant(상수)는 바뀌지 않을 때
ex) 당구공의 반지름

```
const firstName = 'happy';
const lastName = 'hacking';
const fullName = firstName + lastName;
document.write('<h1>' + fullName + '!!' + '</h1>');
```

파이썬 처럼 문자열을 이어서 작성가능

```
const firstName = 'happy';
const lastName = 'hacking';
const fullName = firstName + lastName;

document.write(`<h1>${fullName}!!</h1>`);
```

템플릿 문자열 - Backtic 사용!

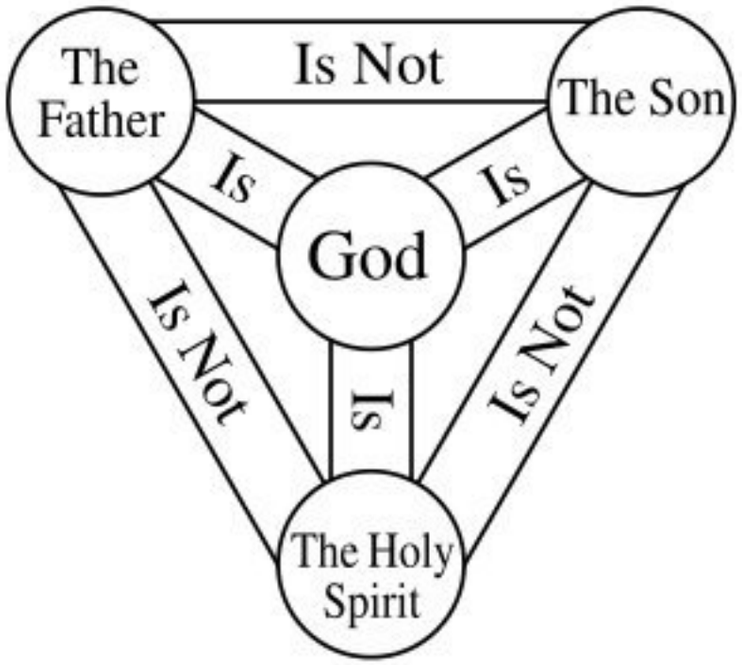
```
const firstName = 'happy';
const lastName = 'hacking';
const fullName = firstName + lastName;
document.write(`Document ${fullName}`);
console.log(`Console ${fullName}`);
```

print()

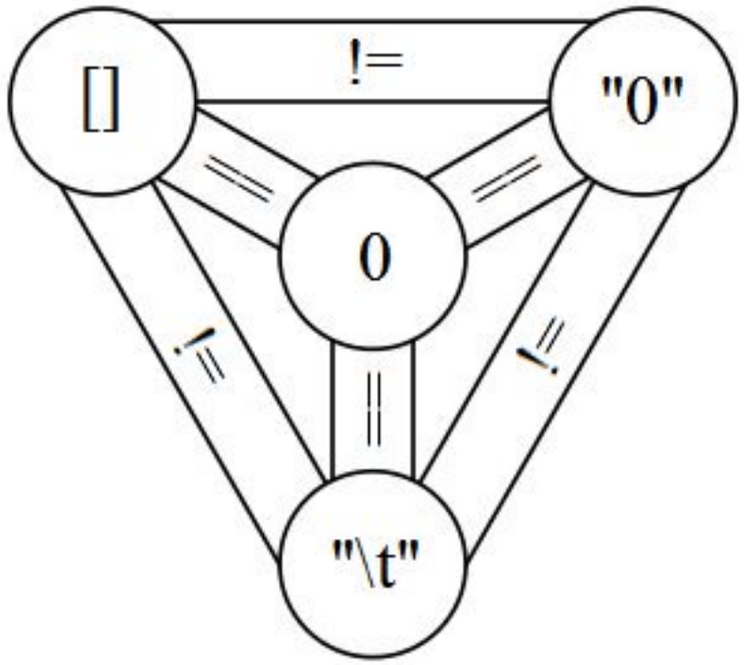
```
const userName = prompt('Hello! Who are you?');
let message = `<h1>Hello ${userName}</h1>`;

document.write(message);
```

input()

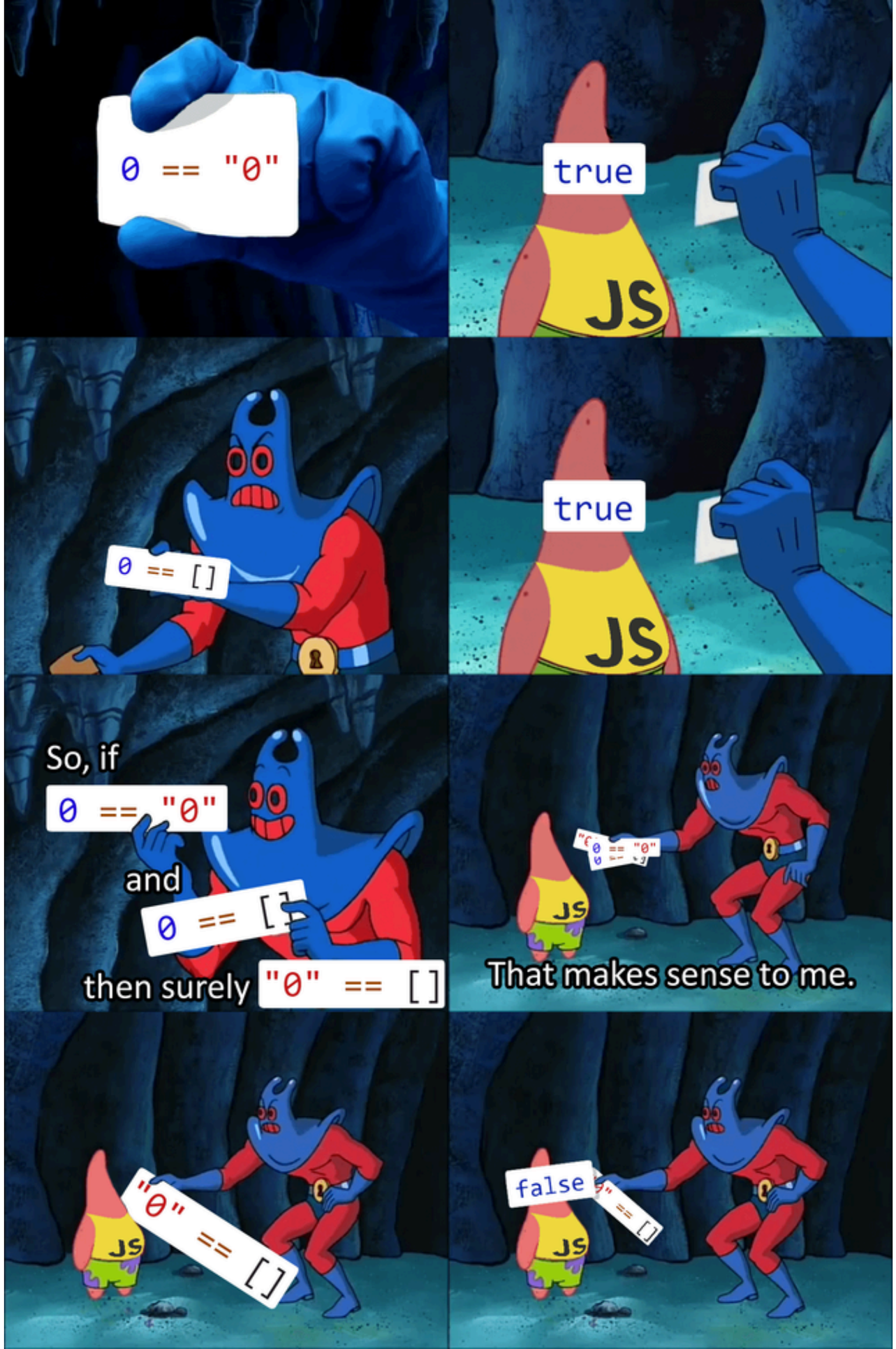
```
const userName = prompt('Hello! Who are you?');
let message = ''

if (userName === '1q2w3e4r') {
    message = '<h1>This is secret Admin page</h1>';
} else if (userName === 'happy') {
    message = '<h1>Hacking!</h1>';
} else {
    message = `<h1>Hello ${userName}</h1>`
}

document.write(message);
```



Christianity



JavaScript


0 == "0"
true
JS
0 == []
true
JS
So, if
0 == "0"
and
0 == []
then surely "0" == []
That makes sense to me.
"0" == []
false
JS

=== : 엄격한 같음 (형을비교)

== : 느슨한 같음 (값을비교)

https://eqeq.js.org/

or

```
let a = 1;

let b = 2;

let c = a + b;

c = c + 10;

c -= 3;

c *= 10;

c++;

--c;
```

```
let i = 0;
while (i < 10) {
    console.log(i);
    i++;
}
```

```
for (let j=0; j < 10; j++) {
    console.log(j);
}
```

```
for (let number of [1, 2, 3, 4, 5]) {
    console.log(number);
}

for (const number of [1, 2, 3, 4, 5]) {
    console.log(number);
}
```

```
const numbers = [1, 2, 3, 4];
```

```
numbers.reverse();
numbers.push('a');
numbers.pop();
numbers.unshift('a');
numbers.shift();
numbers.includes(1);
numbers.push('a');
numbers.indexOf('b');
numbers.join('-');
```

```
const me = {
    name: 'ssafy',
    'phone number': '01012345678',
    languageLevel: {
        python: 'master',
        django: 'pro',
        javascript: 'junior',
    }
};
```

만들고 접근해보자

```
const dessert = {
    coffee: 'Americano',
    iceCream: 'Cookie and cream',
}
const jsonData = JSON.stringify(dessert);
const parseData = JSON.parse(jsonData);
```

내용출력 & 타입확인

```
function add(num1, num2) {
    return num1 + num2;
}
```

함수

```
const sub = function(num1, num2) {
    return num1 - num2;
};
```

함수2

```
const mul = (num1, num2) => {
    return num1 * num2
};
```

함수3

```
let square = (num) => {
    return num ** 2
};
```

함수3

```
square = (num) => num ** 2;
```

함수3`

```
square = num => num ** 2;
```

함수3``

```
square = num => num ** 2;
```

함수3``

```
let noArgs = () => 'No args';
noArgs = _ => 'No args';
```

인자없는함수

```
const sayHello = (name='noName') => `hi ${name}`

sayHello('john');
sayHello();
```

기본인자함수

```
function (num) { return num ** 3 }
(num) => { return num ** 3 }
```

익명함수

1회용으로 사용할 함수는 이름을 짓지 않을 수 있다.