

21-1학기 COMCODE

# 웹프로그래밍 심화

## #2 2주차 내용 복습



# 변수 (Variable)



## 변수의 선언

ECMAScript6

var



let  
const

function level scope

block level scope

## function level scope

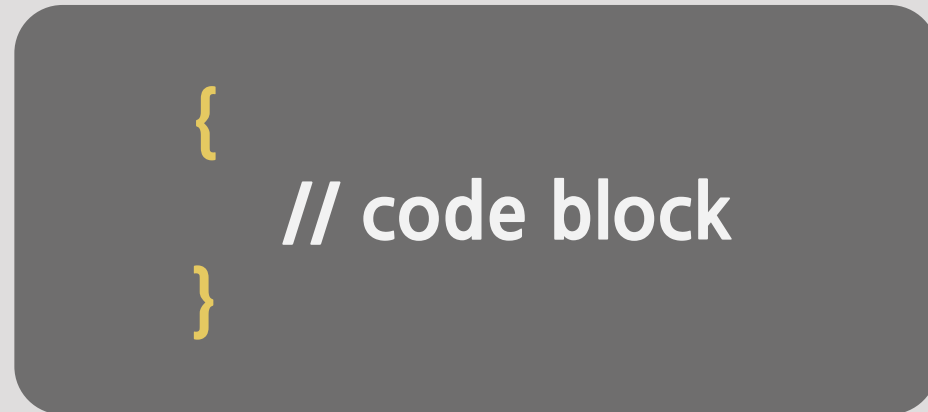
: 함수 **안**에 있는 변수만 **지역** 변수



```
function () {  
  
}
```

## block level scope

: 코드 블록 **안**에 있는 변수만 **지역** 변수



## let vs const

let

재할당 가능

const

재할당 불가능



**Data Types**



**Dynamic Typing**

**object**

**function**

**number**

**string**

**boolean**

**undefined**

**null**

**symbol**

**array**

# 연산자 (Operator)





 == VS ===

==

Loose equality

value

===

Strict equality

value & **type**

# 함수 (Function)



## 함수의 선언

```
function name(param) {  
  // body code  
}
```

기본 함수

```
const nfunc = function () {  
  // body code  
}
```

함수 표현식

Anonymous function

## 함수의 선언

```
function name(param) {  
  // body code  
}
```

기본 함수

```
const nfunc = function name() {  
  // body code  
}
```

함수 표현식  
Named function

## 함수의 선언

```
const name = (param) => code(1 line);
```

Arrow function

```
(function name(param) {  
    // body code  
})();
```

IIFE(즉시 실행 함수)

## 함수의 선언

```
const name = (param) => {  
  // body code  
  return;  
}
```

Arrow function

```
(function name(param) {  
  // body code  
})();
```

IIFE(즉시 실행 함수)



# Parameters

primitive type

passed by  
**value**

object type

passed by  
**reference**

## Default Parameter

: 값이 전달되지 않았을 경우 default 값으로 대체

```
function name(param = 'default value') {  
    // body code  
}
```



## Rest Parameter

: **배열** 형태로 인자 전달

```
function name(...param) {  
    // body code  
}
```

[value1, value2, ...]

name(value1, value2, ...)

## Callback 함수

: 함수를 **인자**로 사용

```
function name(func) {  
    // body code  
}
```

```
const func = function () {}
```