

왕초보 코틀린 #2 변수와 문자열

✔ 주석comment; 1줄

```
// 오늘은 자료형과 변수를 배워볼게요
```

2줄 이상

```
/* 안녕  
 * 하세요 */
```

✔ 클래스 이름은 파스칼 표기법: 모든 단어를 대문자로 시작

```
ClassName
```

함수나 변수 이름은 카멜 표기법: 첫 단어만 소문자로 시작

```
functionName
```

✔ 변수의 선언

- ① **var** : 일반적인 변수, 언제든지 읽기 쓰기가 가능
- ② **val** : 선언시에만 초기화 가능, 중간에 값을 변경할 수 없음

변수는 선언 위치에 따라 두 가지 이름으로 불린다

- ① 클래스에 선언된 변수 : Property (속성)
- ② 이 외의 Scope 내에 선언된 변수 : Local Variable (로컬변수)

✔ println 기능

- ① 문자열을 출력
- ② 정수형 변수의 값도 바로 출력

```
println(변수명)
```

✔ 변수에 값이 할당되지 않았다는 것을 하나의 정보로 사용하는 경우

```
var a:Int? = null
```

자료형 뒤에 물음표를 붙이면 null을 허용하는 nullable 변수로 선언해줄 수 있다

nullable변수

null인 상태로 연산할 시 null pointer exception이 발동할 수 있으므로 꼭 필요한 경우에만 사용할 것

✔ 코틀린이 제공하는 기본 자료형(자바와 거의 동일)

숫자형	정수형	Byte	8bits
		Short	16bits
		Int	32bits
		Long	64bits
	실수형	Float	32bits
		Double	64bits

✔ 정수형의 리터럴

*리터럴: 코드 내에 값을 표기하는 것

10진수	Int(32bits)	*기본형, 숫자만 표시 var intValue:Int = 1234
	Long(64bits)	숫자 뒤에 L 붙이기, 더 큰 메모리를 사용하는 정수 var longValue:Long = 1234L
16진수	‘0x’를 앞에 붙이면 됨 *x는 hexadecimal의 약어 var intValueByHex:Int = 0x1af	
2진수	‘0b’를 앞에 붙이면 됨 *b는 binary의 약어 var intValueBybin:Int = 0b10110110	

✔ 실수형의 리터럴

Float	뒤에 소문자 또는 대문자 f를 붙여주면 16비트의 float 형으로 취급 var floatValue:Float = 123.5f *f는 float의 약어
Double	기본형 var doubleValue:Double = 123.5 필요시 지수 표기법을 추가하면 됨 var doubleLeValueWithExp:Double = 123.5e10 *e는 exponential의 약어

✔ Char의 리터럴

문자열을 유니코드 방식 중 하나인 UTF-16BE로 관리함

글자 하나하나가 2bytes의 메모리 공간을 사용함(16bits)

\t	탭	\b	백스페이스
\r	첫 열로 커서 옮김	\n	개행
\'	작은 따옴표	\"	큰 따옴표
\\	역 슬래시	\\$	\$ 문자
\uxxxx	유니코드 문자		

✔ 논리형

Boolean : 참 또는 거짓

true	var booleanValue:Boolean = true
false	var booleanValue:Boolean = false

✔ 문자열

```
val 변수명 = "문자열"

val 변수명 = """여러줄의
문자열"""
```