

---

---

---

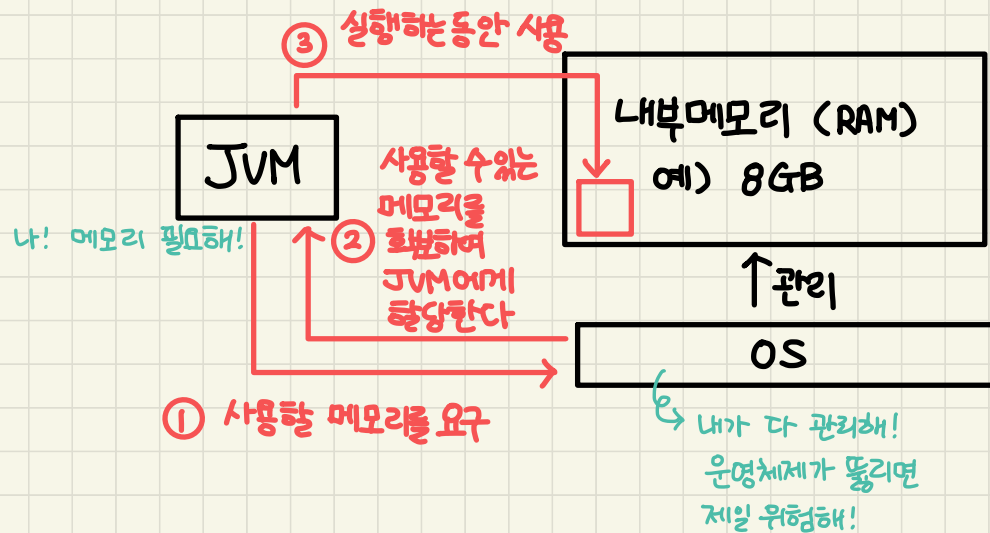
---

---





## \* JVM과 메모리



\* 메모리를 쓰다가 ~ 부족해지면?  
더이상 쓰지 않는 부분이 있는지  
청소하고 확인해야해.  
(garbage collector)

# \* 데이터의 타입

String은 캐릭터배열 / byte 배열

자바에서  
기본으로 만들 수  
있는

Primitive  
Types

정수

값을 담을 수 있는  
메모리

→ { 1 byte : byte  
2 byte : Short  
4 byte : int  
8 byte : long

-128 ~ +127

-32768 ~ +32767

-약 21억 ~ +약 21억

-약 922경 ~ +약 922경

부동소수점

→ { 4 byte : float  
8 byte : double

유효 자릿수 7자리 (단정도) Single-precision

유효 자릿수 16자리 (배정도) Double-precision

float에 비해 배로  
정밀하게 메모리를 담을 수 있기

논리

→ int 타입의 메모리 사용 (4 byte)

배열은 byte 타입의 메모리 사용 (1 byte)

boolean

(정식으 자바에서 boolean이  
몇 byte라 명시하지는 않음)

문자

→

\* 변수 (메모리)에 값 저장

metaverse

```
int age;
```

```
age = 20;
```

↑

assignment operator

(할당 / 배정)

```
int age = 20;
```

~~~~~

||

initializer → 변수 선언과 동시에  
(변수 초기화 문장) 즉시 값을 할당하는 것.

# \* l-value 와 r-value

```
int age;
```

```
age = 20;
```

문제를 실행한후  
결과값을 리턴하는  
명령문

변수 < l-value      r-value : 리터럴, 변수, 표현식 (expression)  
(메모리)

l-value는  
반드시  
메모리여야 한다.  
상수일수 없음.  
~~(20 = 30)~~

```
int age2 = age;
```

```
age2 = age + 11;
```

\* github.io

프로그래밍  
기초

## \* System.in 과 Scanner

Scanner keyboard

System.in

키보드



java

util

LinkedList

1

192002!

## \* Scanner 사용

```
int i = 123;
```

### ① nextInt()

## ② nextFloat()

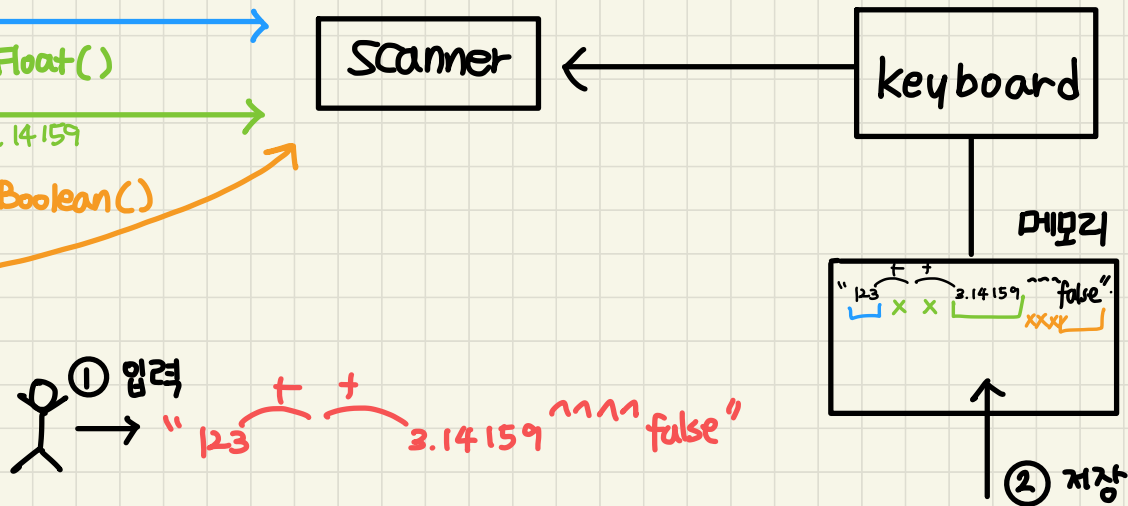
float f = 3.14159

### ③ nextBoolean()

F false

1개의 토큰(token)을 읽을 때까지 기다린다.

↳ 한 단어, 양쪽에 공백으로 (white space, space, tab, newline) 분리된 문자열 (단어)를 가리킨다.

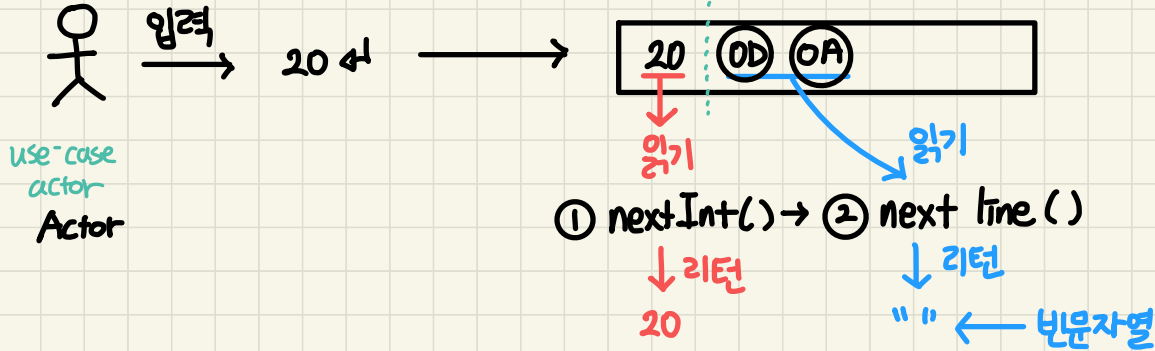


## \* nextXXX()를 사용할 때 주의할 점

↳ 한 개의 토큰을 읽는다

↓  
공백을 만날 때까지 읽는다  
단어 앞의 공백은 제거된다

문자열 토큰  
next



\* nextXxx() 사용법

