

# 23.01.06

## 변수란 무엇일까요?

프로그램에서 이름을 부여 받아, 데이터를 저장할 수 있는 메모리 저장 공간을 말합니다.

## 자바 변수의 종류

자바에서 변수는 **선언된 위치**를 기반으로 4 종류로 나눌 수 있습니다.

먼저 클래스 중괄호를 제외한 중괄호(메소드 중괄호, if, for 중괄호 등)에 선언된 변수를 **지역 변수(local variables)**라고 합니다. 중괄호에서 선언된 부분 부터, 중괄호가 끝나는 부분까지의 생명 주기를 가집니다.

메소드 호출 시 넘겨주는 변수인 **매개 변수(parameters)**도 있습니다. 매개 변수는 메소드가 호출되고 완료되기 까지의 생명 주기를 갖습니다.

클래스 내, 그리고 메소드 밖이자 static을 포함한 변수는 **클래스 변수(class variables)**라고 합니다. 클래스가 초기화된 이후로부터 자바 애플리케이션이 종료되기 까지의 생명 주기를 갖습니다.

클래스 내, 그리고 메소드 밖이자 static을 포함하지 않은 변수를 **인스턴스 변수(instance variables)**라고 합니다. 인스턴스가 생성된 시점부터, 참조하는 변수가 없어 소멸할 때까지를 생명 주기로 가집니다.

## 변수의 자료형

변수의 저장 공간은 변수의 데이터 타입에 따라 달라집니다. 자바의 자료형은 크게 기본형과 참조형으로 나누어볼 수 있습니다.

**기본 자료형**은 자바 언어 자체에 의해 이미 정의된 자료형이며, 자료형에 따라 크기가 달라지고, 항상 null이 아닌 값을 가지고 있습니다.

반면 **참조 자료형**은 클래스를 통해 타입을 정의하며, 참조 변수는 데이터의 값 자체가 아닌 메모리 상에 생성된 데이터의 메모리 주소를 저장합니다.

## 기본 자료형

기본 자료형은 8가지입니다.

논리 자료형인 boolean, 문자 자료형인 char, 정수 자료형인 byte, short, int, long, 실수 자료형인 float, double이 있습니다.

정수 자료형인 byte는 1byte, short는 2byte, int는 4byte, long은 8byte의 크기를 가집니다.

실수 자료형인 float와 double은 각각 4, 8byte의 크기를 가지며, 부동 소수점 방식으로 표현됩니다. 이는 자료형의 비트에서 부호, 가수, 지수를 표현하는 영역을 각각 나누어 수를 표현하는 것을 말합니다. 그러나 각 자료형에 주어진 메모리의 영역이 한정적이므로 완전히 정확한 표현이기는 어렵습니다. 또한 같은 의미에서  $0.1d == 0.1f$  는 false라고 할 수 있습니다.

## 부동소수점으로 표현하기

1. 부호를 부호 비트에 반영
2. 절댓값을 이진법으로 변환 ex)  $118.625 \Rightarrow 1110110.101(2)$
3. 소수점을 이동시켜 정수부가 한 자리가 되도록 변환해줍니다.

ex)  $1110110.101 \rightarrow 1.110110101 \times 2^6$

- 소수점 이동시키는 것을 정규화(Normalization)이라고도 부르며, 여기서 말하는 정규화는 2진수를  $1.xxxx \times 2^n$  꼴로 변환하는 것을 말합니다.

4. 가수부 비트에 실수값 그대로를 넣습니다(점 뒤의 실수 값만. 1은 단순히 1이므로)
5. 지수에 바이어스 값(bias, float의 경우 127)을 더하고 이를 이진수로 표현하여 지수부에 넣습니다.

### \*바이어스 값

지수 비트에서 양수와 음수를 구분하기 위해 더해주는 값입니다. 지수 비트로 표현될 수 있는 값을 절반 나누어서(0-127 / 128-255) 앞은 음수, 뒤는 양수로 표현합니다. 지수에 바이어스 값을 더하여 나온 수를 통해 지수의 부호를 판단할 수 있습니다. 최상위 비트의 값이 음수이면 0, 양수이면 1이 나오기 때문입니다. 이후 7개의 비트로 지수를 계산할 수 있습니다.

\*고정소수점 방식  $\rightarrow$  이진법으로 정수와 소수를 나눈 뒤, 이를 고정된 정수 표현 비트와 소수 표현 비트를 통해 표현하는 것을 말합니다.

## 참고

<https://velog.io/@13circle/자바-스터디-1주차-참조-자료형>

<https://inpa.tistory.com/entry/JAVA-☕-실수-표현부동-소수점-원리-한눈에-이해하기>

<https://m.blog.naver.com/PostView.naver?>

[isHttpsRedirect=true&blogId=aka\\_handa&logNo=10081412293](#)