Java Start

기본형 타입 Primitive Data Types

자료형	키워드	크기	표현 범위	사용 예
논리형	boolean	1byte	true or false (0과 1이 아니다)	boolean isFun = true;
문자형	char	2byte	0~65,535	char c = 'f';
정수형	byte	1byte	-128 ~ 127	byte b = 89;
	short	2byte	-32,768 ~ 32,767	short s = 32760;
	char	2byte	0 ~ 65,535	char c = 64;
	int	4byte	-2147483648 : 2147483647	int x = 59; int z = x;
	long	8byte	...	long big = 3456789L;
실수형	float	4byte	-3.4E038 ~ 3.4E038	float f = 32.5f
	double	8byte	-1.7E308 ~ 1.7E308	double d = 23.34

상수 선언

ex)

final int count = 5;

final double num = 3.14;

→ 원주율 같이 계속 사용되는 실수하기 쉬운 값들을 상수 선언 해놓고 사용

변수 저장

long bing = 3333333333L;

float f = 32.4F;

→ 숫자 뒤에 대문자or소문자 붙여주기

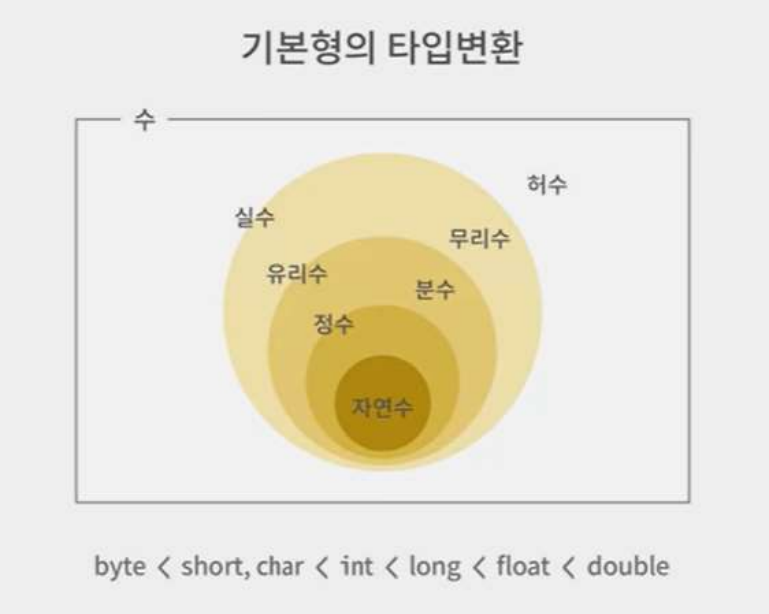
but, 모든 경우를 포함하는 것이 아님

예를 들면, integer 타입인 5를 long 타입에 대입할 때에는

long num = 5;

이렇게 써주면 됨 → 자동으로 형 변환을 해주기 때문

기본형의 타입 변환



실수는 소수점 뒤에 있는 숫자들을 담을 공간이 필요함

*long 타입 변수를 int 타입으로 변환할 때(큰거→작은거)

```
long longValue = 5;
int intValue = (int) longValue;  //형변환 해주기
```

*int 타입 변수를 long 타입으로 변환할 때(작은거→큰거)

```
int intValue = 5;
long longValue = intValue;  //자동 형변환
```

비교 연산자

- == 같다
- != 다르다
- < 크다
- > 작다
- <= 크거나 같다
- >= 작거나 같다

→ 비교 연산자의 결과는 boolean

연산자 우선순위

- 최우선연산자 (., [], ())
- 단항연산자 (++,--,!,~,+/- : 부정, bit변환>부호>증감)
- 산술연산자 (*,/,%,+,-,shift) < 시프트연산자 (>>,<<,>>>) >
- 비교연산자 (>,<,>=,<=,==,!=)
- 비트연산자 (&,|,^~)
- 논리연산자 (&& , || , !)
- 삼항연산자 (조건식) ? :
- 대입연산자 =,*=,/=,%=,+=,-=

논리 연산자

A	B	A && B	A ll B	!A	A ^ B
TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE	FALSE	TRUE
FALSE	TRUE	FALSE	TRUE	TRUE	TRUE
FALSE	FALSE	FALSE	FALSE	TRUE	FALSE

A ^ B → 둘의 결과가 다르면 TRUE, 둘의 결과가 같으면 FALSE

삼항 연산자

(조건식) ? 피연산자1 : 피연산자2 → 조건식이 참(true)이면 피연산자1, 거짓(false)이면 피연산자2 반환

ex)

int b1 = (5 > 4) ? 50 : 40;

→ (5 >4)가 참이라면 50을, 거짓이라면 40을 b1에 저장

switch문

switch는 만난 case부터 모든 case를 수행하는 특징이 있으므로 꼭 break문 활용하기

ex)

(틀린 코드)

```
int value = 2;
switch(value) {
    case 1:
        System.out.println(”1”);
    case 2:
        System.out.println(”2”);
    case 3:
        System.out.println(”3”);
    default:
        System.out.println(”그 외 다른 숫자”);
}
```

(올바른 코드)

int value = 2;

```
switch(value) {
    case 1:
        System.out.println(”1”);
        break;
    case 2:
        System.out.println(”2”);
        break;
    case 3:
        System.out.println(”3”);
        break;
    default:
        System.out.println(”그 외 다른 숫자”);
}
```

변수의 스코프

변수가 적용되는 해당 블럭 → c언어에서 지역 변수?

배열

int[] array1 = new int[100];

int[] array2 = new int[]{1,2,3,4};

int[] array3 = {1,2,3,4};

<배열명>.length : 해당 배열의 크기를 알려줌

for each

for(타입 값을 받아줄 변수명 : 출력하고 싶은 자료구조)

ex)