

1. 프라미티브 타입 : 기본적인 data 타입 ex) int, char, float, boolean

값의 범위 ↓ 기본값	int	$-2^{31} \sim 2^{31} - 1$	0
	long	$-2^{63} \sim 2^{63} - 1$	0
	float	32비트 부동소수점	0.0
	double	64비트 부동소수점	0.0
	char	16 비트	널문자
	boolean	true or false	false

2. 레퍼런스 타입 : 객체를 참조하는 data 타입 (그 객체의 메모리 주소를 참조)  
ex) 클래스, 배열

3. 리터럴 : 실제 사용되는 값  
ex) int a=4, char ch='A' 이시 각각  
리터럴은 4와 'A'

4. 변수 선언 및 초기화 : 선언과 동일,  
선언과 초기화를 동시에 할 수, 따로 할 수 있음

5. 변수 선언과 라이프 타임 : 변수가 사용될 수 있는 범위와 시간

- a) 블록스코프 : 중괄호로 표현된 범위 내부
- b) 함수스코프 : 해당 함수 내부
- c) 클래스스코프 : 해당 클래스 내부

6. 타입 변환 : 타입을 다른 타입으로 바꾸는 것

- a) 타입 프로모션 : 작은 타입  $\rightarrow$  큰 타입 ex) `int`  $\rightarrow$  `long`
- b) 캐스팅 : 큰 타입  $\rightarrow$  작은 타입 ex) `double`  $\rightarrow$  `int`

7. 배열 선언하기 : `int[ ] array = {1, 2, 3}` (1차원)  
`int[ ][ ] array = { {1, 2, 3, 4}, {5, 6, 7, 8} }`  
 두 행의 길이가 달라도 됨

선언과 초기화를 동시에 하지 않으면 크를 지정  
 ex) `int[ ] array = new int[4]`

8. 'var' 키워드 : 프라미티브 타입을 추론하여 자동으로 지정해주는 키워드

ex) `int a = 1`

`var b = a + 1` (var이 자동으로 b가 int임을 인식)

\* C++의 `auto`와 동일