chapter 02.

자바 기본 문법

❍식별자

- 식별(識別)자란? 보고 느낄 수 있는 모든 사물(객체)들을 각각 구별할 수 있는 것을 의미한다.

❍식별자 명명 규칙

- 영문자(A~Z,a~z)와 숫자(0~9)와 '_','$'의 조합
- 첫 글자는 반드시 영문자나 '_'로 시작.숫자로 시작 불허
- 식별자는 대소문자를 철저히 구분
- 자바에서 사용되는 예약어는 식별자로 사용할 수 없다.
- 식별자는 가급적이면 자기 역할에 맞는 이름 부여

❍식별자 정의 규칙

구분	정의 규칙	사용 예

클래스	❑첫 문자는 항상 대문자로 표현
	❑하나 이상의 단어가 합쳐질 때는 각 단어의 첫 문자들만 대문자로 표현
		class JavaTest{
		...;
		}

변수와 메서드	❑첫 문자는 항상 소문자로 표현
	❑하나 이상의 단어가 합쳐질 때는 두 번째부터 오는 단어의 첫 문자들만 대문자로 표현
		String itLand;
		public void getTest(){
		...;
		}

상수	❑모든 문자를 대문자로 표현 ❑하나 이상의 단어가 합쳐질 때 공백 필요시 under score(_)를 사용하여 연결한다.	int JAVATEST = 10; int JAVA_TEST = 20;

❍ 예약어

- 자바 프로그래밍을 하는데 있어 특정한 의미가 부여되어 이미 만들어진 식별자를 말한다.
- 예약어에 등록되어 있는 것을 프로그래밍에서 식별자로 사용할 수 없다.

(const와 goto는 예약어로 등록만 되어 있을 뿐 사용되지 않는 예약어이다.)

[표 2-4] 예약어의 종류

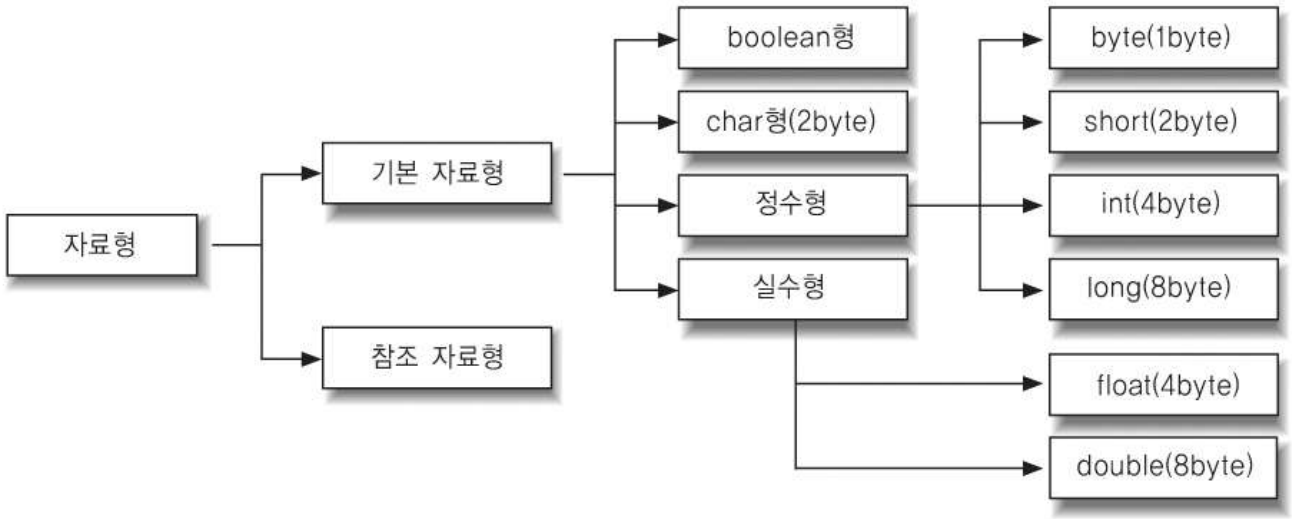
abstract	assert	boolean	break	byte	case	catch
char	class	const	continue	default	do	double
else	enum	extends	false	final	finally	float
for	goto	if	implements	import	instanceof	int
interface	long	native	new	null	package	private
protected	public	return	short	static	strictfp	super
switch	synchronized	this	try	void	while	

❍ 자바의 자료형

- 기본 자료형 (primitive data type)
  - 자바 컴파일러에 의해서 해석되는 자료형

- 참조 자료형 (reference data type)
  - 자바 API에서 제공되거나 프로그래머에 의해서 만들어진 클래스를 자료형으로 선언하는 경우
  - 클래스, 인터페이스, 배열

❍ 자바의 자료형



[그림 2-25] 자료형의 구조

❍ 기본 자료형의 종류

[표 2-5] 기본 자료형의 종류

자료형	키워드	크기	기본값	표현 범위
논리형	boolean	1byte	false	true 또는 false(0과 1이 아니다)
문자형	char	2byte	\u0000	0 ~ 65,535
정수형	byte	1byte	0	-128 ~ 127
	short	2byte	0	-32,768 ~ 32,767
	int	4byte	0	-2,147,483,648 ~ 2,147,483,647
	long	8byte	0	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
실수형	float	4byte	0.0	-3.4E38 ~ +3.4E38
	double	8byte	0.0	-1.7E308 ~ +1.7E308

❍형 변환

[표 2-7] 형 변환의 종류

종류	설명	코딩 예

프로모션	• 더 큰 자료형으로의 변환(자동)
	• 정보의 손실 없음
		short a, b;
		a = b = 10;
		int c = a + b; // 형 변환

		short s = 10; float f = 10 + 3.5f; // 형 변환

디모션	• 더 작은 자료형으로의 변환(명시)
	• 정보의 손실이 발생할 가능성 있음
		short a, b, c;
		a = b = 10;
		c = (short)(a + b); // 형 변환

		int c = 0;
		short s = 10;
		c = (int)(10 + 3.5f); // 형 변환

❑ 3) 연산자1 (73p)

- 연산자란 자료의 가공을 위해 정해진 방식에 따라 계산하고 결과를 얻기 위한 행위를 의미하는 기호들의 총칭이다. 그리고 각 연산자들은 연산을 하기 위해 인식하는 자료형들이 정해져 있다.

[표 2-8] 연산자의 종류와 우선순위

종류	연산자	우선순위
증감 연산자	++, --	1순위
산술 연산자	+, -, *, /, %	2순위
시프트 연산자	>>, <<, >>>	3순위
비교 연산자	>, <, >=, <=, ==, !=	4순위
비트 연산자	&, |, ^, ~	~만 1순위, 나머지는 5순위
논리 연산자	&&, ||, !	!만 1순위, 나머지는 6순위
조건(삼항) 연산자	?, :	7순위
대입 연산자	=, *=, /=, %=, +=, -=	8순위

❍ 산술 연산자

- 4칙 연산(+, −, *, /)과 나머지 값을 구하는 연산자(%)를 말한다.

[표 2-9] 산술 연산자의 종류

구분	연산자	의미
산술 연산자	+	더하기
	−	빼기
	*	곱하기
	/	나누기
	%	나머지 값 구하기

❍ 대입 연산자

- 특정한 상수 값이나 변수 값 또는 객체를 변수에 전달하여 기억시킬 때 사용하는 연산자이다.

[표 2-10] 대입 연산자의 종류

구분	연산자	의미
대입 연산자	=	연산자를 중심으로 오른쪽 변수값을 왼쪽 변수에 대입한다.
	+=	왼쪽 변수에 더하면서 대입한다.
	-=	왼쪽 변수값에서 빼면서 대입한다.
	*=	왼쪽 변수에 곱하면서 대입한다.
	/=	왼쪽 변수에 나누면서 대입한다.
	%=	왼쪽 변수에 나머지 값을 구하면서 대입한다.

❍ 비교 연산자(관계 연산자)

- 변수나 상수의 값을 비교할 때 쓰이는 연산자로서 결과가 항상 true 또는 false인 논리값(boolean)이어야 한다.

[표 2-11] 비교 연산자의 종류

구분	연산자	의미
비교 연산자	>	크다.
	<	작다.
	>=	크거나 같다.
	<=	작거나 같다.
	==	피연산자들의 값이 같다.
	!=	피연산자들의 값이 같지 않다.

❍ 논리 연산자

- true나 false인 논리 값을 가지고 다시 한번 조건 연산하는 연산자이다. 하나 이상의 처리 조건이 있어야 하며 먼저 처리되는 조건에 따라 다음의 처리 조건을 처리할지 안 할지를 결정하는 말 그대로 논리적인 연산자이다.

[표 2-12] 논리 연산자의 종류 (1)

구분	연산자	의미	설명
논리 연산자	&&	and(논리곱)	주어진 조건들이 모두 true일 때만 true를 나타낸다.
	||	or(논리합)	주어진 조건들 중 하나라도 true이면 true를 나타낸다.
	!	not(부정)	true는 false로 false는 true로 나타낸다.

[표 2-13] 논리 연산자의 종류 (2)

연산자	설명
&&	선조건이 true일 때만 후조건을 실행하며 선조건이 false이면 후조건을 실행하지 않는다.
||	선조건이 true이면 후조건을 실행하지 않으며 선조건이 false일 때만 후조건을 실행한다.

❍ 비트 연산자

- 피연산자 즉 연산의 대상이 되는 값들을 내부적으로 bit단위로 변경한 후 연산을 수행하는 연산자이다.

[표 2-14] 비트 연산자의 종류

구분	연산자	의미
비트 연산자	&	비트 단위의 AND
	|	비트 단위의 OR
	^	XOR(배타적 OR)

❍ 시프트 연산자

- bit단위의 연산처리를 하며 자료의 가공을 위해 오른쪽 또는 왼쪽으로 이동하여 값에 대한 변화를 일으키는 연산자이다.

[표 2-16] 시프트 연산자의 종류

구분	연산자	의미
시프트 연산자	〉〉	bit값을 오른쪽으로 이동(빈 자리는 부호값으로 대입)한다.
	〈〈	bit값을 왼쪽으로 이동(빈 자리는 0으로 대입)한다.
	〉〉〉	bit값을 오른쪽으로 이동(빈 자리는 0으로 대입)한다.

❍ 증감 연산자

- 1씩 증가 또는 감소시키는 연산자이다. 무엇보다 중요한 것은 ++ 또는 --와 같은 연산자가 변수 앞에 위치하느냐? 아니면 변수 뒤에 위치하느냐?가 더 중요한 연산자이다.

[표 2-17] 증감 연산자의 종류

구분	연산자	의미
증감 연산자	++	1씩 증가시킨다.
	--	1씩 감소시킨다.

❍ 조건 연산자(삼항 연산자)

- 하나의 조건을 정의하여 만족 시에는 '참값'을 반환하고 만족하지 못할 시에는 '거짓값'을 반환하여 단순 비교에 의해 변화를 유도하는 연산자이다.

[표 2-18] 조건 연산자의 종류

구분	연산자	의미	구성
조건 연산자	? :	제어문의 단일 비교문과 유사하다.	조건식 ? 참값 : 거짓값