변수  
- 표현하려는 수에 맞는 데이터 타입을 이용하여 변수 선언  
- 영문자를 사용, 대소문자 구분함  
- 특문은 $와 \_만 사용  
- 첫 글자는 숫자로 할 수 없음  
- 예약어는 사용 불가  
- 표현 방법 : 카멜 표기법, 스네이크 표기법

- 변수를 선언하면 해당되는 자료형의 크기만큼 메모리가 할당됨

- 추론 가능한 변수 자료형을 선언하지 않음

- 한 번 선언하여 추론된 변수는 다른 타입의 값을 대입할 수 없음

- Local variable type

자료형(Data Type)  
1) 정수형  
- byte(1바이트) : 동영상, 음악 파일, 실행 파일의 자료 처리에 사용  
- short(2바이트)  
- int(4바이트) : 자바에서 사용하는 정수 기본 자료형  
- long(8바이트) : 숫자 뒤에 대소문자L을 써서 long형임을 표시

-

int num = 10; 10진(Deciamal)

int bNum = 01bB1010: 2진(Binary)

int oNum = 012; 8진수(Octal)

int xNum - 0XA; 16진수 (Hexadecimal)

2) 부동소수점 방식 : 소수점의 위치를 고정하지 않고 따로 적음  
- 가수부와 지수부로 표현 ex)1.00010101\*2^4  
- float(4바이트) : 뒤에 대소문자 F를 붙여 표현  
- double(8바이트) : 실수 자료형의 기본 자료형

- 실수 연산에서 오차가 생기는 이유 : 2진수로 컴파일되기 때문 10=>2=>10 과정에서 오차 생김

3) 문자

- 문자도 정수로 표현됨 문자-(encording)- 정수-(decording)-문자

- 문자세트(Character set) : 각 문자를 얼마로 표현할 것인지 코드 값을 보여주는 것

- ASCII : 알파벳과 숫자 특수 문자등 1바이트에 표현하는데 사용하는 문자세트

- UNICOE : 전세계 표준으로 만든 문자 세트 ex) UTF-8 1바이트부터 4바이트까지 다양하게 문자 표현

- 자바는 UNICODE 사용

- char name = ‘a’; : 자료형이 char인 것을 확인해서 유니코드값 기준대로 인코딩 및 디코딩

4) 논리형   
- true, false 두 가지만 나타냄

- 1바이트 사용  
- 값이 존재하는지, 배열이 비었는지, 결과가 참인지 거짓인지등을 표현

상수(constant)

- 변하지 않는 수 : 재할당X ex) 원주율, integrel MIXMAX값

- final 예약어를 사용하여 선언

리터럴(literal)  
- 프로그램에서 사용하는 숫자, 문자, 논리값을 뜻함  
- 정수 리터럴은 int, 실수 리터럴은 double로 저장됨

- 상수와 마찬가지로 상수 풀(constant pool)에 있음

형변환(type conversion)  
- 서로 다른 자료형 간에 연산등의 수행을 위해 하나의 자료형으로 통일

- 묵시적 형변환(자동, explicit), 명시적 형변환(강제, implicit)  
- 작은 바이트 => 큰 바이트 자료형으로 형 변환은 자동으로 이루어짐

- ex)byte=>short=>int=>long, float => double

- 큰 바이트 =>작은 바이트 : 데이터 손실 발생(ex. float=>int: 소수점 사라짐)

이상한 값 나옴(ex int=>byte)

Scanner 클래스 : 입력을 받기 위한 클래스

- Scanner 객체명 = new Scanner(System.in); :스캐너 객체 생성

- System.out.print(); : 입력값 받음

next() : 공백을 기준으로 문자열을 입력 =>단위 입력시 사용

nextline() : enter 키를 입력할 때까지

Integer.parseInt : 문자열 리터럴을 정수 형태로 다시 입력