

# 11강 상수

## 1. 상수

① 프로그램 상에서 의미가 변하지 않는 값

## 2. 문자상수

(1) " " 안에 묶여 있는 문자열을 말함

## 3. 출력식식을 가진 출력문

(1) system.out.printf

ex) system.out.printf("a: %d, b: %s", 100, "안녕하세요");

(2) 사용가능한 서식

%d	정수값
%c	문자
%f	실수형
%s	문자형
%숫자d	"숫자" 만큼의 공간을 확보하고 빈칸은 빈칸으로 출력한다.
%0숫자d	"숫자" 만큼의 공간을 확보하고 빈칸은 '0'으로 채워서 출력.

(3) 숫자가 -2147483648 ~ 2147483647 을 넘어가면 숫자뒤에 L을 붙인다.

# 12강 변수와 자료형

## 1. 변수

① 프로그램에서 각 값의 값의 의미가 서로 변경될 수 있는 것을 의미.

## 2. 자료형

⇒ 기억 공간에 저장 값을 메모리 상의 크기는 개발자가 결정해 주어야 한다!

### (1) 양식

→ 자료형 변수이름

### (2) 자료형

→ 값을 담을 기억공간의 크기, 담겨질 값의 형태를 결정

## 3. 자료형의 종류

### (1) 논리자료형

#### ① boolean

1) 참과 거짓을 담을 수 있는 자료형, 1byte

### (2) 정수자료형

① byte - 1byte의 정수값

② char - 2byte의 문자값

③ short - 2byte의 정수값

④ long - 8byte의 정수값

### (3) 실수자료형

① float - 4byte의 실수값

② double - 8byte의 실수값

### (4) 문자열 자료형 class

① string - " " 묶인 문자의 값을 담을 수 있음.

### (5) 기본자료형

① 정수형 기본자료형 : int (범위가 벗어나면, 숫자뒤에 L 을 붙이면 됨)

② 실수형 기본자료형 : double (범위가 너무 크면, F 를 붙여 메모리 사용을 줄인다.)

# 13강 표준입출력

## 1. 키보드입력

(1) `System.in.read()`

① 자바에서 키보드를 통해 콘솔로 입력을 받는 것을 표준입력이라 함.

② `System.out.println` = 표준 출력

↔ `System.in.read()` = 표준입력

③ `System.in.read()` 를 통해 입력을 받을 때, 글자 한 개를 입력. 여러개 입력을 위해서는 여러번 써줘야함.

④ 엔터를 누를때 

[	CR	→	캐시를 제일 앞으로 이동	]	가 볼음
	LF	→	한 줄 아래로 내리는 것		

## (2) Buffered Reader

① 라인단위의 입력이 가능

② `InputStreamReader ir = new InputStreamReader(System.in);`  
`BufferedReader br = new BufferedReader(ir);`

ex) `String str = br.readLine();`  
`System.out.println(str);`