

### 지난주 추가 내용

`printf` : 변수의 값을 **여러가지 형식으로 변환**해서 출력하는 함수이다.  
(d:정수 f:실수 s:문자열 ....)

## 1.타입변환

[byte < short < int < long < float < double

(float가 long보다 큰 이유는 표현할 수 있는 값의 범위가 더 크기 때문이다)

### 자동타입변환

자바 컴파일러에서 **자동으로 타입을 변경하여 적용**하는 방법이다.

작은 데이터 타입을 큰 데이터 타입으로 변환 후 연산이나 대입을 수행한다.

**ex)** double d1 = 1; // 정수 1을 실수 1.0으로 자동 타입 변환  
(int가 double보다 작아서 변환이 가능하다.)

### 강제타입변환

개발자가 직접 타입을 골라 적용하는 방법이다.

**ex)** byte b = (byte)300 **오류!**  
(큰 → 작음으로 바꿀 때 데이터가 손상이 되어 값이 바뀜)  
double d= (double)3.14f; **정상작동!**  
(작은 → 큰으로 바꿀 때 데이터가 손상이 안된다 그러므로 값 그대로 나옴)

## 2.객체 생성 방법

1. **import** ~을 **package**와 **public class** 사이에 작성한다.
2. 스캐너 객체 생성한다. (Scanner(클래스 이름) 객체 이름 = new 클래스 이름)
3. Scanner 클래스가 제공하는 다양한 메소드를 이용하여 키보드로 데이터 입력 받음

### 3-1. Scanner 클래스 메소드

`next()` 공백 이전의 문자열(String)을 입력 받음  
`nextLine()` 라인 전체(공백을 포함) 문자열(String)을 입력 받음  
`nextInt()` 정수를 입력 받음  
`nextDouble()` 실수를 입력받음  
`next` 자료형() 형태로 자료형에 따라 다르게 받아들일 수 있음  
(ex. `nextByte()`, `nextLong()`, `nextFloat()` 등)

### 3.연산자

연산자(사칙연산) : 값을 연산하기 위해 사용되는 부호를 말한다.

(3 + 5 수식에서 +는 연산자)

비연산자 : 연산자를 이용한 수식에서 사용되는 값들을 말한다.

(3 + 5 수식에서 3과 5는 피연산자)

**산술연산자(+, -, \*, /,% ) => 문자, 숫자 가능**

```
ex) int a = 10;
    int b = 20;
    system.out.println(a+b); =>30
```

\*산술연산할 때 데이터 타입이 일치되어야 함

\*불일치하면 타입변환해서 출력해야함

\*작은 것이 큰 것으로 변환되어 출력됨

```
ex) int a = 2;
    double b = 0.5;
    system.out.println(a+b); =>2.5
```

\*연고 싶은 값이 있다면 타입을 변환하여 출력함

**비교연산자 (==, !=, <=, >=, <, >)**

**\*==는 대입, ==는 같다, !=는 다르다\***

```
ex) int a = 10;
    double b = 20;
    system.out.println(a==b); => false
    system.out.println(a!=b); => true
```

**논리연산자**

피연산자의 조건을 결합해서 true 나 false 를 조사하는 연산자이고 논리 타입에만 사용한다.

```
ex) int a = 10;
    int b = 20;

    • system.out.println(a>0 && b<100); => true
    system.out.println(a<0 && b<100); => false
    system.out.println(a<0 && b>100); => false
=> &&(and) 둘 다 true여야지 true나옴

    • system.out.println(a>0 || b<100); => true
    system.out.println(a<0 || b<100); => true
    system.out.println(a<0 || b>100); => false
=> ||(or) 둘중 하나가 true여야지 true나옴

    • system.out.println(a>0 ^ b<100); => false
    system.out.println(a<0 ^ b<100); => true
    system.out.println(a<0 ^ b>100); => false
=> ^(xor) 둘 다 틀리거나 같으면 false나옴
```

## 07.06 노트정리

- `system.out.println(!(a>100));` => `true`  
=> `!a`(부정) 피연산자의 논리 값을 반대로 반환

### 대입연산자

```
int weight = 50;
weight = weight + 50;
weight +=50; =>복합 대입 연산자
weight -=50;
```

### 증가 감소 연산자

`++a` : 전위 => `a`를 수식에 대입하기 전 1을 먼저 **더해주고** 대입한다.  
`--a` : 전위 => `a`를 수식에 대입하기 전 1을 먼저 **빼주고** 대입한다.  
`a++` : 후위 => `a`를 먼저 수식에 대입한 뒤 1을 **더해준다**.  
`a--` : 후위 => `a`를 먼저 수식에 대입한 뒤 1을 **빼준다**.

ex) `int a =10;`  
`int b =20;`  
`++a;`  
`system.out.println(a);` =>결과 : 11  
`system.out.println(b++);` =>결과 : 20 (**이 문장이 출력하고 나서 적용이 되기 때문에 20으로 나온다.**)  
`system.out.println(b);` =>결과 : 21

### 조건연산자(=삼항연산자)

조건식이 맞으면 연산자1 실행하고 틀리면 연산자2를 실행한다.

ex) `int x =20;`  
`String line = x>10? "Big":"Small";`  
`system.out.println(line);` => 결과 : Big

### 연산자우선순위

- \* `a=b=c=d`=> **오->왼**으로 대입
- \* `++a --a`=> **오->왼**으로 대입

위에서 제외한 다른 연산자들은 모두 **왼 -> 오** 순서로 연산한다.

## 3주차 과제

## 첫번째 과제



```

Problems Javadoc Declaration Console
<terminated> ksi_0720 [Java Application] C:\Users\Wsu
김수임
20
부천대학교
컴퓨터소프트웨어학과
이름 : 김수임
나이 : 20
학교 : 부천대학교
학과 : 컴퓨터소프트웨어학과

```

## 두번째 과제

```

3 public class Main {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int a = 5;
8         int b = 6;
9
10        System.out.println("a의 값: " + a++);
11        System.out.println("b의 값: " + ++b);
12        System.out.println("a+b의 값: " + (a + b));
13
14    }
15
16 }

```

a의 값 : 5

b의 값 : 7

a+b의 값 : 12