

# 2부 자바 기본 다루기

## － 4장 데이터를 조작하기 위한 연산자

최문환



# 데이터를 조작하기 위한 연산자

1. 산술연산자
2. 관계(비교) 연산자
3. 조건연산자
4. 논리연산자
5. 증감연산자
6. 비트단위연산자

# 1. 수학적 연산을 위한 산술연산자

연 산 자	의 미	사 용 예	결 과
+	더하기	$6 + 2$	8
-	빼기	$5 - 3$	2
*	곱하기	$2 * 3$	6
/	나누기(몫)	$8 / 2$	4
%	나누기(나머지)	$8 \% 3$	2

# <예제> 수치 데이터로 사칙연산하기

```
01: class Op01 //산술연산자
02: {
03:     public static void main(String[] args)
04:     {
05:         int a=10, b=4, c;
06:         c=a+b;
07:         System.out.println(a + " + " + b + " = " + c)
08:         c=a-b;
09:         System.out.println(a + " - " + b + " = " + c)
10:         c=a*b;
11:         System.out.println(a + " * " + b + " = " + c)
12:         c=a/b;
13:         System.out.println(a + " / " + b + " = " + c)
14:         c=a%b;
15:         System.out.println(a + " % " + b + " = " + c)
16:     }
17: }
```

No.4

# <예제> 문자열 데이터에 + 연산하기

```
01: public class Opr01_02
02: {
03:     public static void main(String[] args)
04:     {
05:         String a="Apple";
06:         String b="Banana";
07:         String c=a+b;
08:         System.out.println(c);
09:         String str="결과값 : ";
10:         int n=10;
11:         System.out.println(str+n);
12:         System.out.println("결과값 : "+n);
13:     }
14: }
```

## 2. 두 수치 데이터의 대소관계 비교하는 관계 연산자

연 산 자	의 미	사 용 예	결 과
>	크다	$6 > 2$	true
>=	크거나 같다	$2 >= 5$	false
<	작다	$2 < 3$	true
<=	작거나 같다	$8 <= 2$	false
==	같다	$8 == 3$	false
!=	같지 않다	$8 != 6$	true

# <예제> 수치 데이터의 대소 관계 살피기

```
01: class Opr02 //관계연산자
02: {
03:     public static void main(String[] args)
04:     {
05:         int a=10, b=4, c;
06:         boolean test;
07:         test=a>b;
08:         System.out.println(a + " > " + b + " = " + test);
09:         test=a<b;
10:         System.out.println(a + " < " + b + " = " + test);
11:
12:         //산술연산결과에 대한 대소관계를 물어볼 수도 있다.
13:         System.out.println(a+b > a-b);
14:     }
15: }
```

### 3. 둘중에 한개를 선택하도록하는 조건연산자

#### ▶ 형식

```
조건식 ? 수식-1 : 수식-2;
```

수식-1:조건식의 결과가 참(true)일 때 수행되는 식

수식-2:조건식의 결과가 거짓(false)일 때 수행되는 식



# <예제>조건연산자 이용해서 최대값 구하기

```
01: class Opr03
02: {
03:     public static void main(String[] args)
04:     {
05:         int a=5, b=10;
06:         int max=a;        //최대값을 저장할 변수 선언
07:         max = a>b ? a : b;
08:         System.out.println(" max = "+ max);
09:     }
10: }
```

# 4.논리값을 비교 판단하도록 하는 논리 연산자

논리값1	논리값2	OR연산	AND연산
TRUE	TRUE	TRUE	TRUE
TRUE	FALSE	TRUE	FALSE
FALSE	TRUE	TRUE	FALSE
FALSE	FALSE	FALSE	FALSE

논리값3	NOT연산
TRUE	FALSE
FALSE	TRUE

논리 NOT > 논리 AND > 논리 OR

# <예제> 10대인지 아닌지 알아보기

```
01: class Opr04 //논리 연산자와 조건 연산자 활용하기
02: {
03:     public static void main(String[] args)
04:     {
05:         int a=29;
06:         String s="몰라";
07:         s = (a>=10 && a <=19) ? "10대" : "10대 아님";
08:         System.out.println(a + " => " + s);
09:     }
10: }
```

# <예제>대문자인지 아닌지를 판단하기

```
01: class Opr06
02: {
03:     public static void main(String[] args)
04:     {
05:         char ch='b'; //문자 변수 선언
06:         String s; //대문자인지 판단한 결과를 저장할 문자열 변수 선언
07:
08:         s = (ch>='A' && ch<='Z') ? "대문자임" : "대문자가 아님";
09:
10:         System.out.println(ch + "=>" + s);
11:     }
12: }
```

## 5. 확장된 대입연산자

```
int a=10;
```

```
System.out.println(a + 10); //①
```

```
System.out.println(a);      //②
```

```
a = a + 10;
```

혹은

```
a += 10;
```

## 6. 정수형 변수값을 1 증가, 감소하는 증감 연산자

연산자	의미	사용 예
++	1씩 증가.	++a (선행 처리) a++ (후행 처리)
--	1씩 감소.	--a (선행 처리) a-- (후행 처리)

# <예제> 변수의 값을 1 증가시키는 증가 연산자

```
01: class Opr07
02: {
03:     public static void main(String[] args)
04:     {
05:         int a=10, b=10;
06:         ++a;           //선행처리
07:         System.out.println(a);
08:         b++;           //후행처리
09:         System.out.println(b);
10:     }
11: }
```

# <예제> 증감 연산자의 선행처리와 후행처리

```
01: class Opr08 //대입 연산자와 증감 연산자
02: {
03:     public static void main(String[] args)
04:     {
05:         int a=10, b=10;
06:         System.out.println(++a); //선행처리
07:         System.out.println(a);
08:
09:         System.out.println(b++); //후행처리
10:         System.out.println(b);
```



# <예제> 증감 연산사의 선행처리와 후행처리

```
12:    a=b=10;
13:    int c;
14:
15:    c=++a;    //선행처리
16:    System.out.println(c + " => " + a);
17:
18:    c=b++;    //후행처리
19:    System.out.println(c + " => " + b);
20: }
021: }
```

# 7.1 비트 단위 논리 연산자

A	B	$A \& B$	$A   B$	$A \wedge B$	$\sim A$
0	0	0	0	0	1
1	0	0	1	1	0
0	1	0	1	1	1
1	1	1	1	0	0

# <예제> 비트 단위 논리 연산자

```
01: class Opr09 //비트 단위 논리 연산자  
02: {  
03:
```

# <예제> 비트 단위 논리 연산자

```
11:      c = a | b; //비트 단위 값이 둘 다 0일대만 0
12:      System.out.println(a+" | "+b+" -> "+c);
13:
14:      c = a ^ b; //비트 단위값이 다르면 1 같으면
        0
15:      System.out.println(a+" ^ "+b+" -> "+c);
16:
17:      c = ~a; //비트단위값이 1이면 0로0이면1로
18:      System.out.println("~" + a + " -> " + c);
19:  }
20: }
```

# <예제> 비트 단위 쉬프트 연산자

```
01: class Opr10 //비트 단위 쉬프트 연산자
02: {
03:     public static void main(String[] args)
04:     {
05:         byte x = 15;
06:
07:         System.out.println("x << 2 : "+(x << 2));
08:         System.out.println("x >> 2 : " (x >> 2));
09:     }
10: }
```

No.21

# <문제>

1. 짝수인지 홀수인지 판단하는 프로그램을 조건 연산자를 이용하여 작성하시오.(Ex04\_01.java)

[결과] 7은 홀수이다.

8은 짝수이다.