

능동적 사고 방식의

ParkJuByeong

java

강사 박주병



Part02 연산자

01 연산자와 피연산자

02 연산자의 종류

03 연산자 우선순위

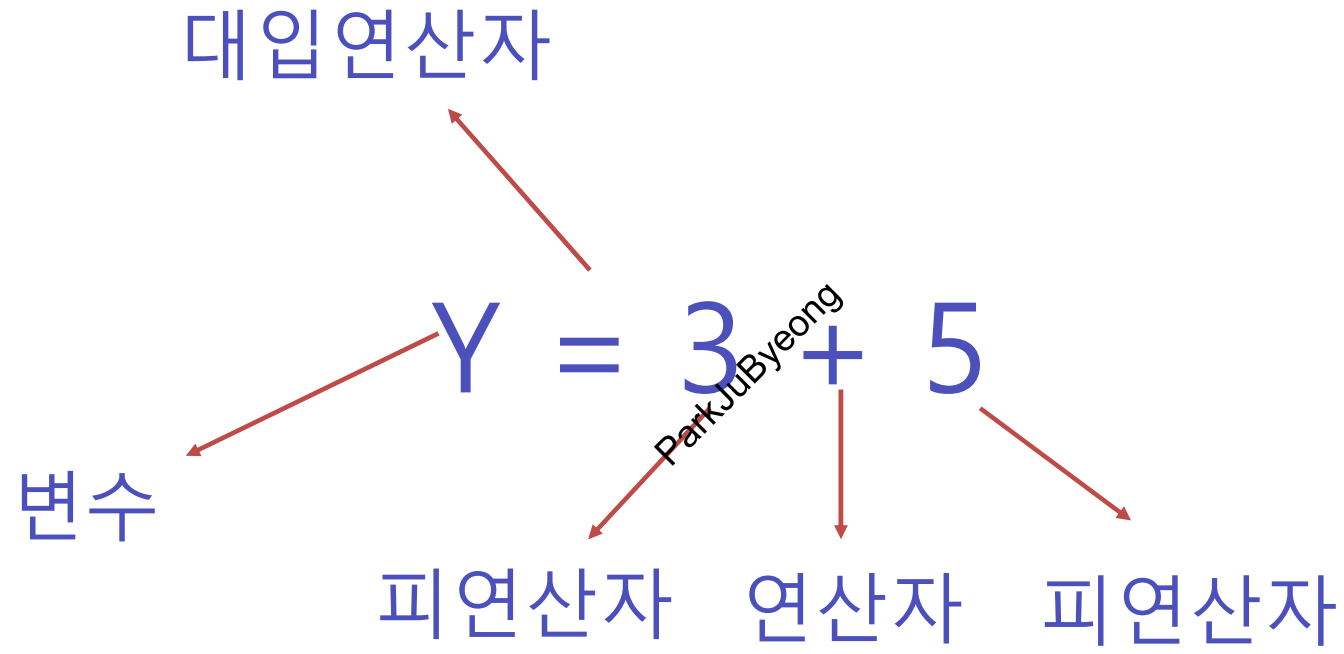
04 실습 문제



ParkJuByeong

01 —

연산자와 피연산자



02

연산자의 종류

ParkJuByeong

연산자의 종류

종류	연산자	설명
산술 연산자	+ - * / % << >>	사칙연산, 나머지 연산, 비트 연산
비교 연산자	> < >= <= == !=	크다, 작다, 같다, 다르다 등의 비교
대입연산자	=	우변의 값을 좌변에 저장
논리 연산자	&& !& ^ ~	조건을 연결
형변환 연산자	()	객체의 타입을 판별
삼항 연산자	?:	조건에 따라 분기
instanceof	instanceof	객체의 타입을 판별

단항연산자

$++3$, $--7$

이항연산자

$3+5$
ParkJubyong

삼항연산자

```
System.out.println(3==5? "같다": "다르다");
```

산술변환

이항 연산자의 경우 피연산자는
타입을 일치시켜야 한다.

```
int a = 10;  
float b = 20.0f;  
  
float result = (float)a+b;
```


산술변환은 자동타입캐스팅이 안될까?

```
int a = 10;  
float b = 20.0f;  
  
float result = a+b;
```

큰타입으로 변환시 자동타입캐스팅이
되기에 생략해도 된다!

long + int

-> long + long

float + int

-> float + float

double + float

-> double + double

byte + short

-> int + int

char + short

-> int + int

-> int보다 작은 타입은 int로
변환되어
이항연산자를 수행한다.

얼마가 출력될까?

```
float result = 5/3;  
System.out.println(result);
```

단항연산자

```
int a = 10;  
int b = ++a;
```

전위형

ParkJuByeong

```
int a = 10;  
int b = a--;
```

후위형

얼마가 출력될까?

```
int a = 10;  
int b = a--;  
  
System.out.println(a);  
System.out.println(b);
```

ParkJuByeong

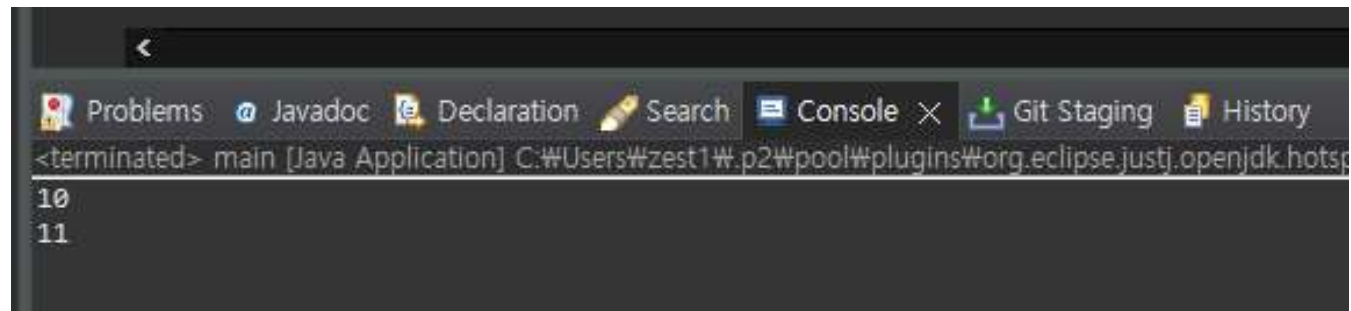
Problems Javadoc Declaration Search Console × Git Staging

<terminated> main [Java Application] C:\Users\zest1\p2\pool\plugins\org.eclipse.justj.open

9
10

얼마가 출력될까?

```
int a = 10;  
int b = a--;  
  
System.out.println(b++);  
System.out.println(b);
```



The screenshot shows the Eclipse IDE's console window. The toolbar includes icons for Problems, Javadoc, Declaration, Search, Console, Git Staging, and History. The console text shows the program has terminated and the output is 10 followed by 11 on the next line.

```
<terminated> main [Java Application] C:\Users\zest1\p2\pool\plugins\org.eclipse.justj.openjdk.hotsp  
10  
11
```

사칙 연산자

```
int result1 = 10+3;  
int result2 = 10-3;  
int result3 = 10*3;  
int result4 = 10/3;  
int result5 = 10%3;
```

*0으로 나눌시 에러를 발생시킨다.



실습 문제

1. 반지름 1.5cm인 원의 둘레를 구하는 프로그램을 만들어 보자.
둘레구하는 공식 : $2 * \text{반지름} * \pi$

```
<terminated> mai  
9.42
```

```
final double PI= 3.14;  
float radius = 1.5f;  
  
System.out.println(radius*2*PI);
```

2. int형 변수 a, b를 선언하여 각각 100만, 200만으로 값을 초기화 한뒤
C 변수에 담아 출력하는 프로그램을 만들어 보자.

***오버플로우를 조심하자!**

```
2000000000000000
```

```
int a = 1000000;  
int b = 2000000;  
long c = (long)a*b;
```

3. char형 변수 a에 문자 '3'을 저장한후
int형 변수 b로 옮길때 사칙연산을 이용하여
숫자 3으로 변환하여 넣어보자.

```
char a = '3';  
int b =  
  
System.out.println(b);
```



```
char a = 'a';  
char b = a+1;
```

→ 더 작은 사이즈로 변환시 명시적
형변환 필요

```
char a = 'a';  
char b = 'a'+1;
```

→ 리터럴 간의 연산은 컴파일러가
코드를 최적화 시킨다.
따라서 char b= 'b'; 상태에서
실행파일로 컴파일 되어 진다.

성능저하 없이 가독성을
높이는 좋은 방법이다.

~~`int sec = 86400; // 하루를 초단위로 표현`~~

`int sec = 60*60*24; // 하루를 초단위로 표현`

비교

예시1

```
int a=10;  
int b=20;  
  
boolean isEqual = a>b;  
boolean isEqual2 = a<b;  
boolean isEqual3 = a>=b;  
boolean isEqual4 = a<=b;  
boolean isEqual5 = a==b;  
boolean isEqual6 = a!=b;
```

Problems @

<terminated> main

false
true
false
true
false
true

문제

```
int a=10;  
float b=10.0f;  
  
System.out.println(a==b);
```



```
Problems  
<terminated> ma  
true
```

```
System.out.println('0' == 0);
```



```
Problems  
<terminated> main  
false
```

```
System.out.println('A' == 65);
```



```
Problems  
<terminated> ma  
true
```

```
System.out.println(0.1 == 0.1f);
```



```
Problems  
<terminated> main  
false
```

```
System.out.println((float)0.1 == 0.1f);
```



```
Problem  
<terminated>  
true
```

0.1f -> 0.10000000149011612로 저장된다.

0.1 -> 0.100000000000000001로 저장된다.

논리

연산자

&&(AND) -> 피연자 모두 true 이면 true를 반환한다.

|| (OR) -> 피연자 둘중 하나만 true이면 true를 반환한다.

```
int x=15;
```

```
System.out.println(true && true);  
System.out.println(true && false);  
System.out.println(true || false);  
System.out.println(3==1 || 10<5);  
System.out.println(x>10 && x<20);
```

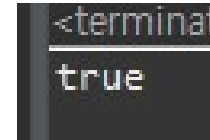
ParkJuByeong

논리 연산자의 우선순위

```
System.out.println(true || true && false);
```

```
System.out.println((true || true) && false);
```

```
System.out.println(true || (true && false));
```



&&가 우선순위가
높다

괄호로 묶어 우선순위를 변경할 수
있다.

괄호가 없어도 &&가 먼저 수행되지만
가독성을 위해 일부러 적어주기도
한다.

키보드로부터 값을 입력받는 Scanner 클래스 사용법

```
1 package joo;
2
3 import java.util.Scanner;
4
5 public class Main {
6
7     public static void main(String[] args) {
8
9         char ch;
10        Scanner scanner = new Scanner(System.in); // 입력을 받기 위한 스캐너 객체를 생성
11
12        System.out.println("값을 입력하세요:");
13        String input = scanner.nextLine(); // 키보드로부터 값을 입력 받아 input 변수에 저장한다.
14        ch = input.charAt(0); // input 변수의 타입을 char로 변경한다.
15
16        |
17
18        System.out.println(ch);
19
20    }
21
22 }
```

실습 문제

1.키보드로부터 문자 하나를 입력 받아 대문자이면 true 소문자이면 false를 출력하는 프로그램을 만들자.

```
<terminated> main [Java A  
값을 입력하세요:  
a  
false
```

```
Problems @ Javadoc  
<terminated> main [Java A  
값을 입력하세요:  
A  
true
```

```
char ch;  
Scanner scanner = new Scanner(System.in); //입력을 받기 위한 스캐너 객체를 생성  
  
System.out.println("값을 입력하세요:");  
String input = scanner.nextLine(); //키보드로부터 값을 입력 받아 input 변수에 저장한다.  
ch = input.charAt(0); //input 변수의 타입을 char로 변경한다.  
  
System.out.println(ch>='A' && ch<='Z');
```


노란 밑줄은 왜 생겼을까?

```
System.out.println(true || false);  
System.out.println(false && true);
```

논리 부정 연산자

```
System.out.println(! (true && false));
```



```
<terminated  
true
```

ParkJuByeong

조건 연산자(삼항연산자)

```
char result = 10 == 20 ? 'a' : 'b';  
System.out.println( result);
```

조건 연산자 중첩

양수이면 1

0이면 0

음수이면 -1

```
int a=10;  
int result = a>0 ? 1 : ( a==0 ? 0 : -1);  
  
System.out.println( result);
```

실습 문제

1. 65명의 사람이 있고 한번에 4명이 탈수 있는 차가 있다면 모든 사람이 한번에 이동하기 위해서 몇대의 차가 필요한지 계산하는 프로그램을 만드시오

```
11  
12     int people = 65;  
13     int carPassengers = 4;  
14  
15  
16     System.out.println((people/carPassengers)+(people % 4 > 0 ? 1:0));  
17
```

대입연산자

```
int a = 10;
```

```
17  
18 int a = 10;  
19 System.out.println( a=30);  
20 System.out.println( a);  
21  
<  
Problems Javadoc Declaration Search Console  
<terminated> main [Java Application] C:\Users\Wzest1\p2\pool\plu  
30  
30
```

1. 대입연산자 역시 값을 반환한다.
2. 대입연산자의 왼쪽은 값이 변경가능한것만 올수 있다.
(리터럴, 상수 불가능)

ParkJuByeong

복합 대입연산자

```
int a = 10;  
a += 20;  
a *= 3;  
a %= 10;
```

=
ParkJuByeong

```
int a = 10;  
a = a + 20;  
a = a * 3;  
a = a % 10;
```

ParkJuByeong

03

연산자 우선순위

$X+3*y$

$X+3 > y-2$

$X>3 \ \&\& \ x<5$

$\text{true} \ || \ \text{true} \ \&\& \ \text{false}$

ParkJuByeong

04

실습문제

ParkJuByeong

문제1

Int형 변수 num에 100이상의 숫자를 저장후 백의자리 이하는 버리고 출력하시오

```
18 int num = 4556;  
19 System.out.println( );  
  
<terminated> main [Java Application] C:\Users\zest1\p2\pool\plugins  
4500
```

문제2

키보드로 부터 영문자를 입력 받아 대문자를 소문자로 변경하여 출력하시오
(유니코드표를 참조하지 않아도 되는방법도 생각해볼것)

```
값을 입력하세요:  
A  
a
```

```
System.out.println( ch >= 'A' && ch <= 'Z' ? (char)(ch+('a'-'A')) : ch );
```

문제3

Int형 변수 num의 숫자가 홀수, 짝수인지 출력 하시오

```
20 int num = 30;  
21 System.out.println( );  
  
<terminated> main [Java Application] C:\Users\zest1\p2\pool\plugins\org.eclipse.justi.c  
짝수
```

문제4

화씨 100도를 섭씨로 변환하여 출력하시오
(소수점 둘째자리를 반올림할것)
변환공식 -> 섭씨 = 5/9 x (화씨 -32)

```
<terminated> ma  
화씨 : 100  
섭씨 : 37.8
```

```
int f = 100;  
float c = (int)(5f/9 * (f-32)*10+0.5)/10f;
```



THANK YOU

ParkJuByeong



강사 박주병