

```
package test;
```

```
import java.util.Scanner;
```

```
public class test {  
    public static void main(String[] args) {
```

정수형(명시적 형 변환)

```
float f1 = 1.234f; // 실수형은 소수점까지 표현 가능
```

```
int a = 7; // 정수형 소수점 표현 불가능
```

```
int result = a + (int)f1; int가 없으면 오류 정수형과 실수형은 계산 불가능  
System.out.println(result); 7+1
```

```
byte by = 10000; byte는 -128 ~ 127까지 표현 가능
```

```
byte cy = (byte)10000; by = 128; -128 overflow
```

자동 형변환

```
int ba = 6;
```

```
float baf = 13.01f; 실수형의 특징 항상 소수점이 붙는다
```

```
float result1 = ba;
```

```
System.out.println("result1 = " + result1);
```

자동 형변환 순서

```
byte -> short, char -> int -> long -> float -> double  
}
```

```
}
```

```
package test;
```

```
import java.util.Scanner;
```

```
public class scannerdemo {  
    public static void main(String[] args) {
```

```
Scanner scan = new Scanner(System.in);
```

```
float = f;
```

```
double = d;
```

```
char = c;
```

```
int test = scan.nextInt(); // 10
```

```
f = scan.nextFloat();
```

```
d = scan.nextDouble(): abcd 1 : a
```

```
c = scan.next().charAt(0);
```

```
System.out.println(test);
```

```
}
```

```
}
```

```
System.out.println("후위 증감연산자++ ; " + test++); // ++ = +1 -- = -1
```

```
System.out.println("전위 증감연산자++ : " + ++test);
```

```
System.out.println(test);
```

연산자 종류

산술 연산자

+ - * / %

관계 연산자

대소비교 연산자

>, <, >=, <=

등가비교 연산자

==, !=

논리 연산자

||, &&, !

증감 연산자

++, --

관계연산자 - 대소비교 연산자

int i1 = 1;

int i2 = 2;

System.out.println(i1 == i2); 1==2 ? false

System.out.println(i1 != i2); 1!=2 ? true

연산자 우선순위

괄호>산술>비교>논리>대입