

문자열 연산

이렇게 문자열 사이에 덧셈(+) 연산자를 사용하면 문자열을 연결해줄 수 있습니다.

```
String myString = "Hello " + "Codeit!";
System.out.println(myString);
```

```
Hello Codeit!
```

문자열과 숫자를 더하면 숫자가 저절로 문자열로 바뀌고, 문자열과 불린을 더하면 불린이 저절로 문자열로 바뀝니다.

```
System.out.println("I am " + 27 + " years old.");
System.out.println("The result is " + false + ".");
```

```
I am 27 years old.
The result is false.
```

문자열 사이에 특수한 문자를 표현하고 싶을 수 있습니다. 예를 들어서 큰 따옴표(")로 둘러싸인 문자열 내에 큰 따옴표(")를 넣는 경우가 있겠죠. 이런 경우에 맥에서는 역슬래쉬(\), 윈도우즈에서는 원화 표시(₩)를 사용하면 됩니다.

```
System.out.println("데카르트는 \"나는 생각한다. 고로 존재한다.\"라고 말했다.");
```

```
데카르트는 "나는 생각한다. 고로 존재한다."라고 말했다.
```

이렇게 \ 또는 ₩를 이용하여 문자열 안에 확장된 표현을 하는 문자를 이스케이프 문자(Escape Character)라고 합니다.

이스케이프 문자에는 이런 것들이 있는데, 사실 저희는 "\n", "\", "\'" 정도만 자주 쓸 것입니다.

Escape Sequence	Description
\t	탭
\b	백스페이스
\n	줄 바꿈 (new line)
\r	줄 바꿈 (carriage return)
\f	폼 피드(form feed)
'	작은 따옴표
"	큰 따옴표

..	큰 따옴표
₩	역슬래시

줄 바꿈을 위해서 맥에서는 `"\n"`, 윈도우즈에서는 `"\r\n"` 을 사용하셔야 합니다.

숫자 연산

이제 숫자 연산들을 봅시다. 이런 연산자들이 있는데요:

연산자	설명
+	덧셈
-	뺄셈
*	곱셈
/	나눗셈
%	나머지

정수형과 정수형 간의 연산은 같은 자료형의 결과값을 갖습니다.

```
System.out.println(9 / 5);
```

1

서로 다른 자료형의 연산은 랭크가 더 높은 자료형의 결과값을 갖습니다. 랭크에 대하여는 '형 변환' 강의에서 배우겠습니다.

소수형은 정수형보다 랭크가 높기 때문에 소수형과 정수형 간의 연산의 결과값으로는 소수형이 나옵니다.

```
System.out.println(9.0 / 5);
```

1.8

단항 연산자

자바에는 단항 연산자도 있습니다. 몇 가지만 간단하게 소개드리자면:

```
int a = 3;
```

```
System.out.println(-a); // 양, 음 부호가 바뀜
```

```
System.out.println(++a); // a에 1을 추가
```

```
System.out.println(--a); // a에 1을 감소
```

-3

4

3

불린 연산 비교 연산자 (Comparison Operators)

두 숫자형 값을 비교하고 싶으면 비교 연산자를 쓰면 됩니다.

```
int a = 3;
int b = 5;

System.out.println(a > b); // a가 b보다 크다
System.out.println(a >= b); // a가 b보다 크거나 같다
System.out.println(a < b); // a가 b보다 작다
System.out.println(a <= b); // a가 b보다 작거나 같다
System.out.println(a == b); // a는 b와 같다
System.out.println(a != b); // a는 b와 같지 않다
```

```
false
false
true
true
false
true
```

불린 연산자 (Boolean Operators)

불린 연산도 어느정도 익숙하시죠? AND(&&)의 경우 양쪽 다 **true** 이어야만 **true** 가 나오고, 한쪽이라도 **false** 면 **false** 가 나옵니다.

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

```
true
false
false
false
```

OR(||)의 경우 양쪽 다 **false** 이어야만 **false** 가 나오고, 한쪽이라도 **true** 면 **true** 가 나옵니다.

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

```
true
true
true
```

```
true
false
```

마지막으로 NOT(!) 연산은 불린값을 뒤집어줍니다.

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

```
false
true
```

숫자 연산처럼 섞을 수도 있습니다!

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

```
true
```



수업을 완료하셨으면 체크해주세요.



수강생 Q&A 보기



[\(/questions?assignment_id=262&sort_by=popular\)](#)
질문하기

[assignment_id=262&sort_by=popular\)](#)
[\(/questions/new?](#)

[assignment_id=262&op1=%EA%B0%9D%EC%B2%B4+%EC%A7](#)

< 이전 강의
연산자 [\(/assignments/252\)](#)

다음 강의 > [\(/assignments/253\)](#)
형 변환