

*Be as proud of Sogang as Sogang is proud of you*

# 파이썬 프로그래밍 :: 연산자



서강대학교  
SOGANG UNIVERSITY

- 수식(expression)은 피연산자들과 연산자의 조합
- 피연산자가 모두 정수형이 아닐 경우, 결과는 항상 실수형

operator	meaning	example
+	덧셈 또는 + 부호	+10 <sup>(*)</sup> 15 + 20
-	뺄셈 또는 - 부호	-10 2.5 - 1.5
*	곱셈	10 * 20 = 200
/	나눗셈(실수)	1/2 = 0.5 (결과는 항상 실수)
%	나눈 후 나머지 (modulo)	9 % 4 = 1 (9를 4로 나눈 나머지)
//	나눈 후 몫 (floor division)	15//4 = 3 (15를 4로 나눈 몫)
**	지수승	2**4 = 16 (2 <sup>4</sup> = 16)

**a = 10; b = 4; x = 10.0; y = 4.0**

```
print(a + b)  # a + b = 10 + 4 = 14
print(a + x)  # a + x = 10 + 10.0 = 20.0
print(a * y)  # a * y = 10 * 4.0 = 40.0
```

(\*) 양수의 경우 부호 생략 가능

- += 처럼 대입 연산자와 다른 연산자를 합쳐 놓은 연산자

assignment	예	설명
<b>+=</b>	$x += y$	$x = x + y$ 와 동일
<b>-=</b>	$x -= y$	$x = x - y$ 와 동일
<b>*=</b>	$x *= y$	$x = x * y$ 와 동일
<b>/=</b>	$x /= y$	$x = x / y$ 와 동일
<b>//=</b>	$x //= y$	$x = x // y$ 와 동일
<b>%=</b>	$x \% = y$	$x = x \% y$ 와 동일
<b>**=</b>	$x ** = y$	$x = x ** y$ 와 동일

## ■ 수치연산 관련 내장함수

함수	설명	예
<code>abs()</code>	절대값 반환	<code>abs(-10)</code> # 10
<code>round()</code>	반올림 계산	
<code>divmod(x,y)</code>	x를 y로 나눈 몫과 나머지 반환, ( <code>x//y</code> , <code>x%y</code> ) 쌍을 반환	<code>divmod(17,4)</code> #(4,1)

```
>>> 0.1 + 0.2                # 실수는 근사치 사용, 정확한 0.3이 아님
0.30000000000000004
>>> 0.1 + 0.1 + 0.1 == 0.3
False
>>> round(0.1 + 0.1 + 0.1, 10) == round(0.3, 10)
True
>>> round(3.123456, 4)      # 반올림 후 소수점 이하 자릿수 4
3.1235
```

`round(실수)` 또는 `round(실수, 자릿수)`

: 자릿수는 반올림 후의 소수점 이하 자릿수를 의미하며, 지정하지 않으면 정수 반환

- python의 math 관련 함수들을 모아둔 모듈
- math 모듈의 함수를 사용하기 위한 import문(3가지 방법)

```
from math import * # 이 경우 모듈 이름이 불필요  
a = sqrt(2.0)      # sqrt() 함수를 함수명으로만 호출
```

```
import math         # 이 경우 math.을 붙여야 함  
a = math.sqrt(2.0)  # sqrt() 함수 앞에 해당 모듈명을 명시해야 함
```

```
import math as m    # 이 경우 m.을 붙여야 함  
a = m.sqrt(2.0)     # m은 math의 별칭에 해당
```

- math 모듈에는 많은 함수들이 존재
  - trunc() 함수 : 인수로 받은 값의 버림 계산( `math.trunc(1.5) == 1` )
  - pow(x,y) 함수 :  $x^y$ 을 반환 ( `math.pow(81, 0.5) == 9.0` )

- 두 값을 비교하는 연산자 : 결과는 True(참) 또는 False(거짓)

operator	description	example
<b>==</b>	equal	5 == 7 # False
<b>!=</b>	not equal	5 != 7 # True
<b>&gt;</b>	greater than	5 > 7 # False
<b>&lt;</b>	less than	5 < 7 # True
<b>&gt;=</b>	greater than or equal	5 >= 7 # False
<b>&lt;=</b>	less than or equal	5 <= 7 # True

```
>>> a= 100; b=100;
```

```
>>> print(a==b)      # 두 수가 같으면 True
```

```
True
```

```
>>> print(a!=b)
```

```
False
```

```
>>> print(a>b, a<b)
```

```
False False
```

```
>>> print(a<=b, a>=b)
```

```
True True
```

## ■ 논리 연산자의 종류

- 복잡한 조건을 표현하려면 논리연산자를 사용
- 몇 개의 조건식을 조합하여 명령문의 수행여부를 결정할 때 사용

operator	description	example
<b>and</b>	logical and. ~이고 그리고	모두 True이어야 True
<b>or</b>	logical or. ~이거나 또는	하나라도 True이면 True
<b>not</b>	negates the truth value. 부정	참이면 거짓. 거짓이면 참

```
>>> a = 99
>>> (a > 100) and (a < 200)
False
>>> (a == 100) or (a != 200)
True
>>> not (a == 100)
True
```

- 논리 연산자의 연산은 다음 진리표(truth table)에 의한다.

입력값		x 그리고 y
x	y	x and y
False	False	False
False	True	False
True	False	False
True	True	True

입력값		x 또는 y
x	y	x or y
False	False	False
False	True	True
True	False	True
True	True	True

입력값	x가 아니다
x	not x
False	True
True	False

!x 과 동일



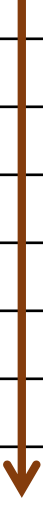
## ■ 해당 년도가 윤년인지 확인하기

### ■ 윤년의 정의

- 2월이 29일까지 있는 년도
- 4로 나뉘 떨어져야 하고, 100으로 나뉘 떨어지면 안 됨. 또는 400으로 나뉘 떨어지면 윤년

```
year = 2020
if ((year % 4 == 0) and (year % 100 != 0)) or (year % 400 == 0) :
    print("윤년")
else :
    print("윤년 아님")
```

## 우선순위

( )	anything in brackets is done first	Highest
**	exponentiation	
-x, +x	arithmetic operators	
*, /, %, //	arithmetic operators	
+, -	arithmetic operators	
<, >, <=, >=, !=, ==	relational operators	
=, +=, -=, *=, etc	assignment operators	
not	logical operator	
and	logical operator	
or	logical operator	Lowest

- 같은 우선순위를 갖는 operator는 왼쪽부터 계산
- 단, \*\* operator는 오른쪽부터 (예:  $2^{**}2^{**}3 = 2^{**}8 = 256$ )
- 애매하면 괄호 ( ) 를 사용 (예:  $(2^{**}2)^{**}3 = 4^{**}3 = 64$ )

1. 어느 학생의 과제 점수가 21.9, 37, 13.6 일 때, 학생의 평균 과제 점수를 구하는 script 작성 할 것
  - ① 각 점수는 변수에 저장할 것
  - ② 평균 값의 소수점 이하는 버리고 출력할 것
  - ③ math 모듈의 함수를 사용할 것

출력

average : 24

2. 3928원의 금액을 500원 동전으로 교환하고 나머지 금액을 100원 동전으로 교환하고자 한다. 500원 동전, 100원 동전의 개수를 계산하여 출력하고 남은 금액을 출력하는 script 작성하라.

출력

500원짜리 동전 : 7 개  
100원짜리 동전 : 4 개  
남은 금액 : 28 원