

# 4. 파이썬 연산자

Made By. 김규일

- 연산자는 단어 그대로 연산을 위한 구조이다.
- 피연산자의 값을 조작할 수 있다.
- ex)  $4 + 9$  에서 4와 9는 피연산자가 되고 +는 연산자가 된다.
- 연산자는 기본적으로 피연산자의 데이터 타입이 같아야 하지만 파이썬 자체에서 데이터 타입을 변경하여 연산을 해주는 경우도 있다.
  - 각 연산자에 맞게 피연산자의 데이터 타입이 정해진 경우가 많고 피연산자들끼리의 데이터 타입이 맞지 않아 오류가 생기는 경우가 많으니 연산자에 맞게 피연산자의 데이터 타입을 맞추는 것은 기본이다.
- 파이썬에는 산술연산자, 비교연산자, 할당연산자, 논리연산자, Bitwise 연산자, 멤버쉽연산자, Identity 연산자 등이 있다.

## 산술연산자

- 산술연산자에는 사칙연산자(+ - \*/)와 제곱을 나타내는 \*\*, 나머지를 산출하는 % (Modulus), 나누기에 소수점 이하를 버리는 //(Floor Division) 등이 있다.

Operator	Example
+ 더하기	$a + b = 30$
- 빼기	$a - b = 30$
* 곱하기	$a * b = 200$
/ 나누기	$a = 5, b = 3 \ a / b = 1.6666666$
** 제곱	$a = 3, b = 4 \ a ** b = 81$
// 정수 몫	$9 // 2 = 4 \ 9.0 // 2.0 = 4.0$
% 나머지	$a \% b = 0$

# 비교연산자

- 비교연산자는 관계연산자로도 불린다.
- 피연산자들을 비교하는 연산이고 부등호, 등호가 존재한다.
- 비교연산자의 연산 결과는 True, False로 나뉘지는 불 타입이다.

Operator	Example
==	a == b a와 b는 같다.
!=	a != b a와 b는 같지 않다.
>	a > b a가 b보다 크다.
<	a < b b가 a보다 크다.
>=	a >= b a가 b보다 크거나 같다.
<=	a <= b b가 a보다 크거나 같다.
is	a is b 객체가 같다.
is not	a is not b 객체가 같지 않다.

# 할당연산자

- 할당연산자는 변수에 값을 할당할 때 사용하는 연산자이다. 기본적으로 등호를 사용한다.

Operator	Description	Example
=	왼쪽 변수에 오른쪽 값을 할당한다	c = a
+=	왼쪽 변수에 오른쪽 값을 더하고 결과를 왼쪽변수에 할당	c += a → c = c + a
-=	왼쪽 변수에서 오른쪽 값을 빼고 결과를 왼쪽변수에 할당	c -= a → c = c - a
*=	왼쪽 변수에 오른쪽 값을 곱하고 결과를 왼쪽변수에 할당	c *= a → c = c * a
/=	왼쪽 변수에서 오른쪽 값을 나누고 결과를 왼쪽변수에 할당	c /= a → c = c / a
%=	왼쪽 변수에서 오른쪽 값을 나눈 나머지의 결과를 왼쪽변수에 할당	c %= a → c = c % a
**=	왼쪽 변수에 오른쪽 값만큼 제곱을 하고 결과를 왼쪽변수에 할당	c **= a → c = c ** a
//=	왼쪽 변수에서 오른쪽 값을 나눈 몫의 결과를 왼쪽변수에 할당	c //= a → c = c //

## 논리연산자

- 논리연산자에는 and, or, not 연산자가 존재한다.
- and는 피연산자 2개 모두 참일 경우 참이고 그 외의 경우에는 거짓이다.
- or은 피연산자 중 하나라도 참이면 참이고 그 외의 경우에는 거짓이다.
- not은 피연산자가 참이면 거짓, 거짓이면 참이다.
- 논리연산의 결과는 참과 거짓인 불 타입이다.

### and 연산자

T	T	= T
T	F	= F
F	T	= F
F	F	= F

### or 연산자

T	T	= T
T	F	= T
F	T	= T
F	F	= F

### not 연산자

T	= F
F	= T

Operator	Example
and	a and b
or	a or b

not	not b
-----	-------

- 다른 언어는 && || 등 기호를 사용하지만 파이썬에서는 and, or, not을 직접 사용한다.

## 비트연산자

- 비트연산자에는 &(AND), |(OR), ^(XOR), ~(Complement), <<, >>(Shift)가 존재한다.
- 비트 단위의 연산을 하는데 사용된다.

Operator	Description	Example
&	AND 연산. 둘다 참일때만 만족	(a & b) = 12 → 0000 1100
	OR 연산. 둘 중 하나만 참이어도 만족	(a   b) = 61 → 0011 1101
^	XOR 연산. 둘 중 하나만 참일 때 만족	(a ^ b) = 49 → 0011 0001
~	보수 연산.	(~a) = -61 → 1100 0011
<<	왼쪽 시프트 연산자. 변수의 값을 왼쪽으로 지정된 비트 수 만큼 이동	a << 2 = 240 → 1111 0000
>>	오른쪽 시프트 연산자. 변수의 값을 오른쪽으로 지정된 비트 수 만큼 이동	a >> 2 = 15 → 0000 1111

## 멤버쉽연산자

- 멤버쉽연산자에는 in, not in이 존재한다.
- 이는 좌측 Operand가 우측 컬렉션에 속해 있는지 아닌지를 체크한다.
- 파이썬 멤버쉽 연산은 문자열, 리스트, 튜플과 같은 시퀀스 멤버쉽을 테스트하는데 자주 사용되고 반복문과 함께 자주 사용된다.

Operator	Description	Example
in	list 내에 포함되어 있으면 참	(a in list) = False
not in	list 내에 포함되어 있지 않으면 참	(b not in list) = True

## 식별연산자(Identity)

- 식별연산자에는 is, not is가 존재한다.

- 두 객체의 메모리 위치를 체크한다.
- 객체 단위의 `==`, `!=` 연산으로 이해하면 편하다.

Operator	Description	Example
is	개체메모리 위치나 값이 같다면 참	(a is b) = True
is not	개체메모리 위치나 값이 같지 않다면 참	(a is not b) = False

## 연산자 우선순위

- 연산자가 여러 개일 경우 괄호를 통해 우선순위를 정할 수 있다.
- 괄호가 없다면 아래의 순서에 따라 우선순위가 결정되고 우선순위에 따라 먼저 연산이 진행된다.
- 아래 표는 우선순위가 높은 순부터 낮은 순을 나타낸다.

Operator	Description
**	지수 (전원으로 인상)
~ + -	Ccomplement, 단항 플러스와 마이너스 (마지막 두의 메서드 이름은 + @이며, - @)
* / % //	곱하기, 나누기, 나머지, 몫
+ -	덧셈과 뺄셈
>> <<	좌우 비트 시프트
&	비트 'AND'
^	비트 전용 'OR'와 정기적 인 'OR'
<= < > >=	비교 연산자
<> == !=	평등 연산자
= %= /= //= -= += *= **=	할당 연산자
is is not	식별 연산자
in not in	멤버 연산자
not or and	논리 연산자