

Lesson 5

연산자(Operator)



목차

1. 연산자란?
2. 연산자의 종류
 - 산술 연산자
 - 복합 대입 연산자
 - 관계 연산자
 - 논리 연산자
 - 멤버 연산자
 - 삼항 연산자
3. 연산자 우선 순위

연산자(operator)란?

연산에 사용하는 기호

+ : 더하기

- : 빼기

* : 곱하기

/ : 나누기

** : 거듭제곱

// : 나누기의 몫

% : 나누기의 나머지



`divmod()` 함수

몫과 나머지를 튜플로 반환함.


코드

```
print(10**2)
print(10**10)
print(10//3)
print(10%3)
print(divmod(10, 3))
```

```
100
10000000000
3
1
(3, 1)
```

복합 대입 연산자

숫자를 계산할 때 사용. 산술 연산 후 해당 변수에 대입하는 연산자

Python 

`+=`

`**=`

`-=`

`//=`

`*=`

`%=`

`/=`

코드

```
a=1
```

```
a+=1
```

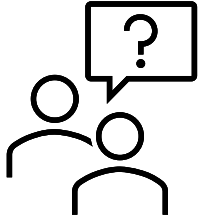
```
a*=5
```

```
print(a)
```



`a+=1`은 `a=a+1`와 동일한 수식이다.
`a*=5`는 `a=a*5`와 동일한 수식이다.

10



1. 아래 수식의 결과는 얼마인가?

$a = 0$

$a += 5$

$a * = 3$

코드

```
#####  
# 10~99 사이의 정수 입력: 55  
# 십자리: 5  
# 일자리: 5  
#####
```

코드

```
#####
```

```
# 10~99 사이의 정수 입력: 55
```

```
# 십자리: 5
```

```
# 일자리: 5
```

```
#####
```

```
number = int(input("10~99 사이의 정수 입력:"))
```

```
unit_10 = number // 10
```

```
unit_1 = number % 10
```

```
print("십자리:", unit_10)
```

```
print("일자리:", unit_1)
```


도전!

[문제] 동전 교환 프로그램 작성하기

금액을 입력하세요: 752

500원 짜리: 1

100원 짜리: 2

50원 짜리: 1

10원 짜리: 0

나머지 돈: 2

키보드에서 손을
떼고 잠시 어떻게
짜야할 지 종이에
적어본 후 코딩
합시다!!!

도전!

[문제] 동전 교환 프로그램 작성하기

금액을 입력하세요: 752

500원 짜리: 1

100원 짜리: 2

50원 짜리: 1

10원 짜리: 0

나머지 돈: 2

```
money = int(input('금액을 입력하세요: '))
m500 = money // 500
money %= 500
m100 = money // 100
money %= 100
m50 = money // 50
money %= 50
m10 = money // 10
money %= 10
print(f"500원짜리: {m500}")
print(f"100원짜리: {m100}")
print(f"50원짜리: {m50}")
print(f"10원짜리: {m10}")
print(f"나머지 돈: {money}")
```

> : 초과

< : 미만

>= : 이상

<= : 이하

== : 같다

!= : 같지 않다

코드

```
print(5==5)
print(10<5)
print("ace"!="")
```

```
True
False
True
```

and : 조건들을 "~이고"로 지정할 때 사용.
조건이 모두 참이면 True를 반환함.
조건1 and 조건2

or : 조건들을 "~이거나"로 지정할 때 사용.
조건이 하나라도 참이면 True를 반환함.
조건1 or 조건2

not : 논리값을 반전 시킬 때 사용.
True를 False로, False를 True로 변경.
not 조건

코드

입사 지원 조건: 40세 미만이고 토익이 850이상

```
age=40
```

```
toeic=860
```

```
print(age<40 and toeic>=850)
```

False

입사 지원 조건: 40세 미만이거나 토익이 850이상

```
age=40
```

```
toeic=860
```

```
print(age<40 or toeic>=850)
```

True

도전!

[문제] 연도를 입력받아서 입력한 연도가 윤년인지 아닌지 확인하는 프로그램 작성하기



윤년이란? 4의 배수이고 100의 배수가 아닌 연도 또는 400의 배수인 연도

```
판별할 연도 입력: 2022
2022년은 False
판별할 연도 입력: 2024
2024년은 True
```

키보드에서 손을
떼고 잠시 어떻게
짜야할 지 종이에
적어본 후 코딩
합시다!!!

도전!

[문제] 연도를 입력받아서 입력한 연도가 윤년인지 아닌지 확인하는 프로그램 작성하기



윤년이란? 4의 배수이고 100의 배수가 아닌 연도 또는 400의 배수인 연도

```
판별할 연도 입력: 2022
```

```
2022년은 False
```

```
판별할 연도 입력: 2024
```

```
2024년은 True
```

```
year = int(input('판별할 연도 입력: '))  
leaf_year = (year % 4 == 0 and year % 100 != 0) or year % 400 == 0  
print(f'{year}년은 {leaf_year}')
```

참 **if** 조건식 **else** 거짓

코드

```
print("짝수" if 10%2==0 else "홀수")
```

짝수

```
age=20
```

```
print("성인" if age>=19 else "청소년")
```

성인

도전!

[문제] 국어, 영어, 수학 점수를 입력받아서 평균 80이상이면 "합격", 아니면 "불합격"표시.

```
국어 점수:90
영어 점수:45
수학 점수:60
평균 65.0점이고, 결과는 불합격입니다.
```

키보드에서 손을
떼고 잠시 어떻게
짜야할 지 종이에
적어본 후 코딩
합시다!!!

`in` : 포함하고 있으면 `True`

`not in` : 포함하고 있지 않으면 `True`

코드

```
print('h' in "happy")
```

`True`

```
print(10 in [1,2,3,4])
```

`False`

도전!

[문제] 입력받은 문자가 word에 있는지 표시하기.

```
word="python"
```

```
찾을 문자 입력:t
```

```
0
```

```
찾을 문자 입력:s
```

```
X
```

비트 연산자

정수를 2진수로 변환한 후 각 자리의 비트끼리 연산하는 연산자.

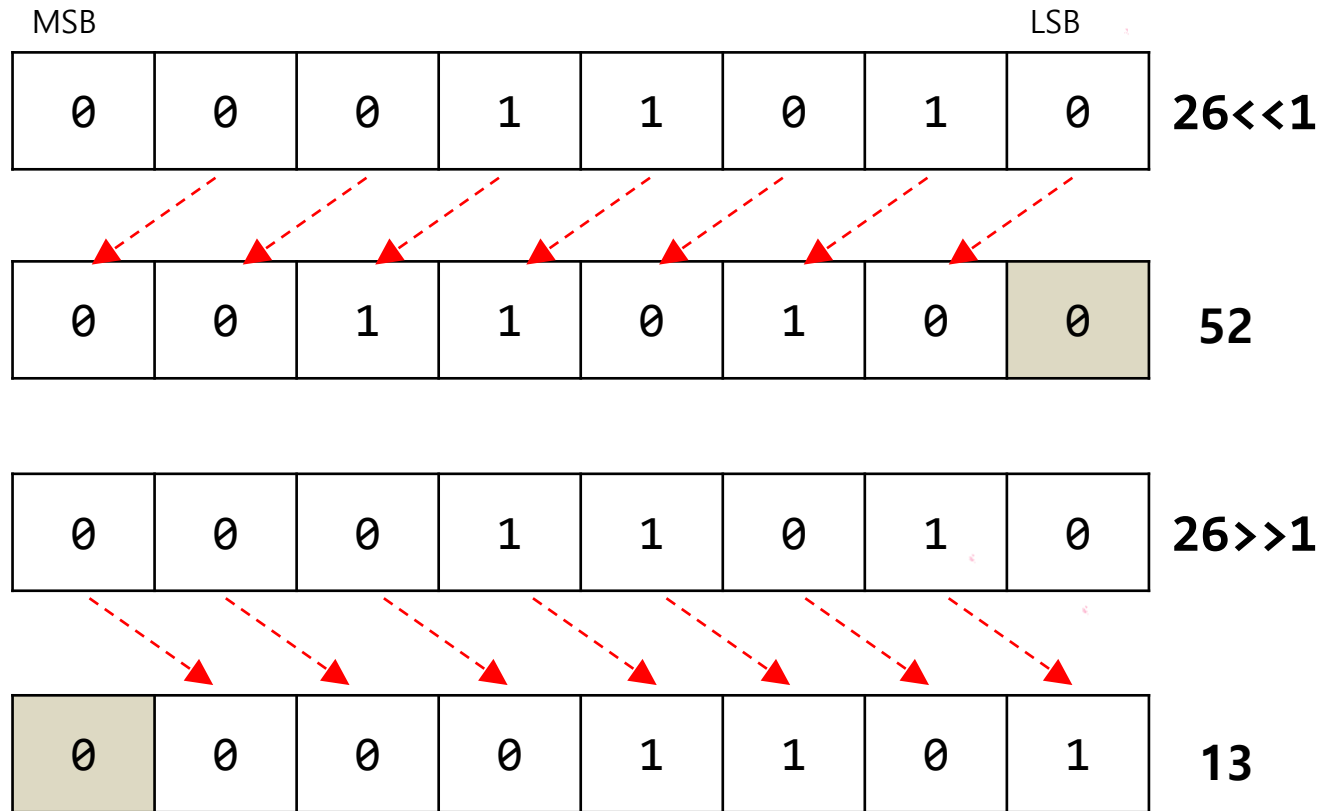
- & : 비트 and, 모두 1이면 1
- | : 비트 or, 하나라도 1이면 1
- ~ : 비트 not, 1->0, 0->1. $\sim a \Rightarrow -(a+1)$
- ^ : 비트 xor, 서로 다른 값이면 1

x	y	x & y	x y	x ^ y	x	~x
0	0	0	0	0	0	1
0	1	0	1	1	1	0
1	0	0	1	1		
1	1	1	1	0		

<< : 비트 left shift
>> : 비트 right shift



MSB(Most Significant Bit)는 어떤 데이터의 최상위 비트
LSB(Least Significant Bit)는 어떤 데이터의 최하위 비트



왼쪽 Shift 는 2^n 을 곱한 효과
오른쪽 Shift는 2^n 으로 나눈 효과



하드웨어 제어, 그래픽 처리,
암호화, 마스킹 작업 등에 많이
사용함

코드

```
data = 0b0
mask = 0b1
bit_and = data & mask
bit_or = data | mask
bit_xor = data ^ mask
bit_not = ~data & mask

print("data      : {}".format(data))
print("mask      : {}".format(mask))
print('-' * 15)
print("and : {}".format(bit_and))
print("or  : {}".format(bit_or))
print("xor : {}".format(bit_xor))
print("not : {}".format(bit_not))
```

```
data : 0
mask : 1
-----
and  : 0
or   : 1
xor  : 1
not  : 0
```

코드

```
print('-' * 30)
data=0b0001101
```

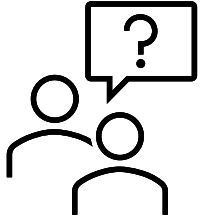
```
print("before: {:08b}".format(data))
print("<<1 :    {:08b}".format(data<<1))
print("<<2 :    {:08b}".format(data<<2))
print(">>1 :    {:08b}".format(data>>1))
print(">>2 :    {:08b}".format(data>>2))
```

```
before: 00001101
<<1 :   00011010
<<2 :   00110100
>>1 :   00000110
>>2 :   00000011
```

연산자 우선 순위

$3 + 4 / 4 > 6$ and $2 < 4$

우선순위	연산자	의미
1	(), [], { }	괄호, 리스트, 딕셔너리, 셋...
2	**	거듭제곱
3	+ - ~	단항 연산자
4	* / // % + -	산술 연산자
5	<<, >>, &, ^,	비트 연산자
6	<, <=, >, >=, ==, !=	관계(비교) 연산자
7	=, %=, /=, //=, +=, -=, *=, /=, **=	대입 연산자
8	not, and, or	논리 연산자
9	참 if 조건 else 거짓	삼항 연산자



1. 아래 수식의 결과는 얼마인가?

$$3 + 4 / 4 > 6 \text{ and } 2 < 4$$