

# 생초보를 위한 파이썬 프로그래밍

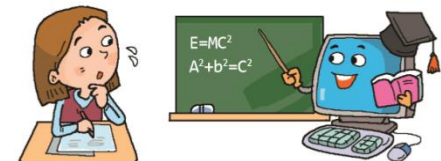
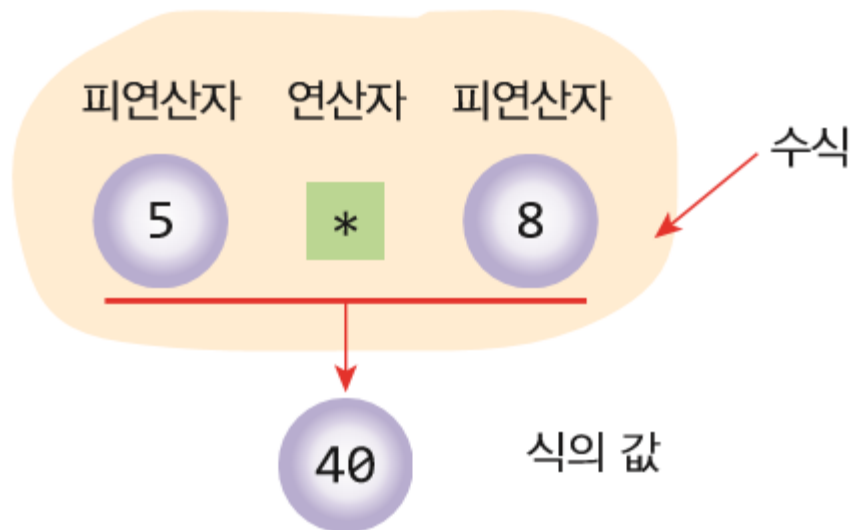
## 4장 연산자

- 산술 연산자

# 연산자와 피연산자

2/32

- 수식(expression)=피연산자들과 연산자의 조합
- 연산자(operator): 연산을 나타내는 기호
- 피연산자(operand): 연산의 대상이 되는 값



## □ 덧셈, 뺄셈, 곱셈, 나눗셈, 나머지 연산

문자열 결합시에도 사용

연산자	기호	사용례	결괏값
덧셈	+	$7 + 4$	11
뺄셈	-	$7 - 4$	3
곱셈	*	$7 * 4$	28
나눗셈	//	$7 // 4$	1
나눗셈	/	$7 / 4$	1.75
나머지	%	$7 \% 4$	3

나눗셈의 몫에서 정수값만 얻어짐

나눗셈의 몫에서 정수 + 실수값 모두 얻어짐

```
i=7  
j=4  
result = i+j  
print(result)  
result = i-j  
print(result)  
result = i*j  
print(result)
```

```
result = i/j  
print(result)  
result = i//j  
print(result)  
result = i%j  
print(result)
```

```
11  
3  
28  
1.75  
1  
3
```

- 나머지 연산자의 용도
  - ▣ 배수인지 아닌지 판단할때 많이 사용됨
    - 4 나누기 2 -> 몫 : 2 , 나머지:0
    - 7 나누기 2 -> 몫 : 3 , 나머지:1
    - 10 나누기 2 -> 몫 : 5 , 나머지:0

```
i=4  
j=2  
result = i%j  
print(result)
```

```
i=7  
result = i%j  
print(result)
```

```
i=10  
result = i%j  
print(result)
```

0

1

0

- 초단위의 시간을 받아서 몇 분 몇 초인지를 계산하여 보자.





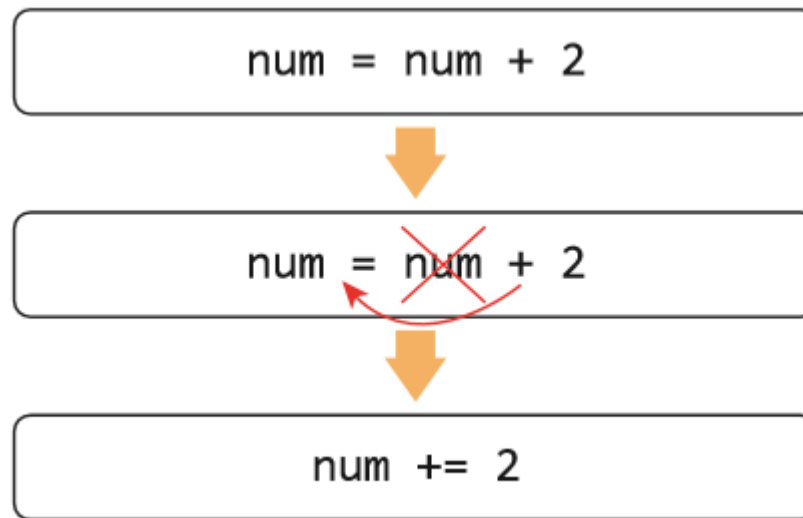
한림대학교 SW중심대학

# 생초보를 위한 파이썬 프로그래밍

## 4장 연산자

- 복합 연산자

- 복합 연산자(**compound operator**)란 +=처럼 대입 연산자와 다른 연산자를 합쳐 놓은 연산자이다.





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

```
x = 1000
print("초깃값 x=", x)
x += 2;
print("x += 2 후의 x=", x)
x -= 2;
print("x -= 2 후의 x=", x)
```

초깃값 x= 1000  
x += 2 후의 x= 1002  
x -= 2 후의 x= 1000

+= 연산자의 경우 문자열도 가능  
Email= 'apple'  
Email=email+'@naver.com'



Email= 'apple'  
Email+= ' @naver.com'

- 다음을 풀어서 표현하시오

```
x = 1000
print("초깃값 x=", x)
x += 2;
print("x += 2 후의 x=", x)
x -= 2;
print("x -= 2 후의 x=", x)
```

- 복합연산자를 이용하여 1-5까지의 합을 구하시오

# Lab: 평균 구하기



12/32

- 평균을 구하고자 한다. 잘못된 부분은 어디일까?

```
x = int(input("첫 번째 수를 입력하시오: "))  
y = int(input("두 번째 수를 입력하시오: "))  
z = int(input("세 번째 수를 입력하시오: "))  
avg = x + y + z / 3  
print("평균 =", avg)
```



첫 번째 수를 입력하시오: 10  
두 번째 수를 입력하시오: 20  
세 번째 수를 입력하시오: 30  
평균 = 40.0

# Solution

13/32

```
x = int(input("첫 번째 수를 입력하시오: "))  
y = int(input("두 번째 수를 입력하시오: "))  
z = int(input("세 번째 수를 입력하시오: "))  
avg = (x + y + z) / 3  
print("평균 =", avg)
```

첫 번째 수를 입력하시오: 10  
두 번째 수를 입력하시오: 20  
세 번째 수를 입력하시오: 30  
평균 = 20.0



한림대학교 SW중심대학

# 생초보를 위한 파이썬 프로그래밍

## 4장 연산자

### □ 비교 연산자

연산자	설명
==	같다
!=	다르다
<	작다
>	크다
<=	작거나 같다
>=	크거나 같다

- boolean 상수를 리턴
  - ▣ 참 : True
  - ▣ 거짓 : False

```
x=2  
y=7  
result=(x==y)  
print(result)
```

```
result=(x!=y)  
print(result)
```

```
result=(x<=y)  
print(result)
```

```
result=(x>=y)  
print(result)
```

False

True

True

False



- 문자열도 ==와 != 연산자로 비교 가능
- 단어가 같아도 대소문자가 다르면 다른 문자열로 판단

```
x='Python'  
y='python'  
z='Python'  
result=(x==z)  
print(result)
```

```
result=(x!=y)  
print(result)
```

True  
True

# 생초보를 위한 파이썬 프로그래밍

## 4장 연산자

- 논리 연산자

Operator	Description	Example
and	논리 AND 연산. 둘다 참일때만 참	(a and b) = False
or	논리 OR 연산. 둘 중 하나만 참이여도 참	(a or b) = True
not	논리 NOT 연산. 논리 상태를 반전	not(a and b) = True

구분	연산식		결과	설명
AND (논리곱)	true	and	true	피 연산자 모두가 true 일 경우에만 연산 결과는 true
	true		false	
	false		true	
	false		false	
OR (논리합)	true	or	true	피 연산자 중 하나만 true 이면 연산 결과는 true
	true		false	
	false		true	
	false		false	
NOT (논리부정)		not	true	피 연산자의 논리값을 바꿈
			false	

```
result=(10 == 10 and 10 != 5 )  
print(result)
```

```
result=(10 > 5 or 10 < 3 )  
print(result)
```

```
result=not (10 > 5 )  
print(result)
```

True  
True  
False



한림대학교 SW중심대학

# 생초보를 위한 파이썬 프로그래밍

## 4장 연산자

- 지수연산자, 연산자 우선순위

- 지수(power)를 계산하려면 \*\* 연산자를 사용한다.

```
x=2  
y=7  
result=x**y  
print(result)
```

---결과----

128

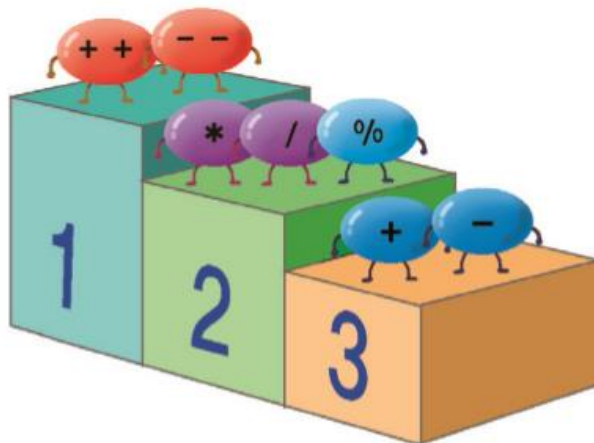
순위	연산자	설명
1	**	지수 연산자
2	~ + -	단항 연산자
3	* / % //	곱셈, 나눗셈, 나머지 연산자
4	+ -	덧셈, 뺄셈
5	>> <<	비트 이동 연산자
6	&	비트 AND 연산자
7	^	비트 XOR 연산자, 비트 OR 연산자

# 연산자의 우선 순위

24/32

$$x + \underbrace{y * z}_{\textcircled{1}} \\ \underbrace{\hspace{1.5cm}}_{\textcircled{2}}$$

$$\underbrace{(x + y) * z}_{\textcircled{1}} \\ \underbrace{\hspace{1.5cm}}_{\textcircled{2}}$$





연산자의 우선순위를 생각하지 말고 ()를 사용하여, 수학적 수식과 같이 계산되도록 하는 것이 좋다.

```
result = 10 + 20 / 2  
print(result)
```



```
10+(20/2)
```

```
result = (10 + 20) / 2  
print(result)
```

```
20  
15
```