

2장. 변수와 자료형, 연산자



목 차

1

변수

2

숫자 자료형

3

연산자

4

문자 자료형

변수(variable)

■ 변수란?

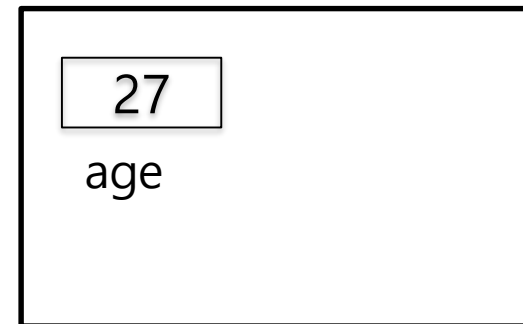
- 프로그램에서 사용되는 자료를 저장하기 위한 메모리 공간(영역)
- 할당받은 메모리의 주소 대신 부르는 이름
- 프로그램 실행 중에 값 변경 가능, variable 이라 함

■ 변수의 선언 및 초기화

- 변수 선언은 어떤 타입의 데이터를 저장할 것인지 그리고 변수이름은 무엇인지를 결정한다. (자료형)은 생략되어 있음.

(자료형) 변수이름 = 초기값;

예) age = 27 , name = "박민국"



변수(variable)

- 변수의 선언

변수명 = 값

- 변수명 작성시 주의

- 변수 이름은 숫자로 시작할 수 없고, 공백이 있으면 안됨.

틀린 예) 3box = 10; sky 123 = "id"

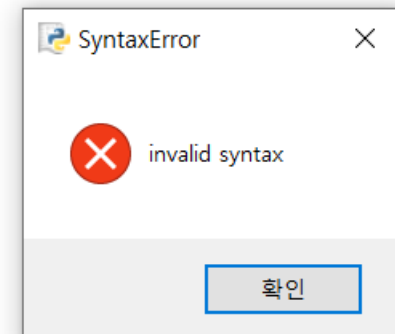
- 변수 이름은 영문, 특수문자, 숫자의 결합으로 만든다.
- 변수 이름은 대, 소문자를 구분한다.

var.py

```
n = 10
3box = 'k'

print(n)
print('n')

n = 20
print(n)
```



자료형(Type)

● 자료형이란?

- 사용할 데이터의 종류에 따라 메모리 공간을 적절하게 설정해 주는 것
- 파이썬에서는 표기하지 않음 (예, num=10, name="홍길동", pi=3.14)

분류	자료형	설명	예
정수	int	소수점이 없는 수	-2, -1, 0, 1, 2
실수	float	소수점(.)이 있는 수, 부동소수점수라고도 불린다.	-3.5, 0.0, 1.25
문자열	str	알파벳과 다른 문자로 이루어진 문장	"a", 'hello', "비"
논리형	bool	참과 거짓을 표현	True, False

자료형(Type)

- 자료형을 반환해주는 type(인자값) 함수

```
>>> x = 100
>>> type(x)
<class 'int'>
>>> y = 2.54
>>> type(y)
<class 'float'>
>>> season = '여름'
>>> type(season)
<class 'str'>
>>> ch = 'k'
>>> type(ch)
<class 'str'>
>>> 10 > 4
True
>>> type(10==5)
<class 'bool'>
```

```
>>> seasons[0]
'봄'
>>> seasons[3]
'겨울'
>>> seasons[-1]
'겨울'
>>> type(seasons)
<class 'list'>
```

숫자형 자료

- 숫자 형태로 이루어진 자료

분류	예
정수	-2, -1, 0, 1, 2
실수	-3.5, 0.0, 1.25
8진수	0o10, 0o34
16진수	0xA, 0xFF

컴퓨터에서 데이터 표현하기

■ 비트(binary digit)

bit(비트) : 컴퓨터가 표현하는 데이터의 최소 단위로 2진수 하나의 값을 저장할 수 있는 메모리의 크기

컴퓨터는 0과 1로만 데이터를 저장함(0-> 신호꺼짐, 1-> 신호켜짐)

■ 비트로 표현할 수 있는 수의 범위

비트수	표현할 수 있는 범위(십진수)	
1bit	0, 1(0~1)	2^1
2bit	00, 01, 10, 11(0~3)	2^2
3bit	000, 001, 010, 011, 100, 101, 110, 111(0~7)	2^3

10진수를 2진수로 바꾸기

■ 진수 표현

10진수	2진수	16진수	10진수	2진수	16진수
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F
8	1000	8	16	10000	10

자리 올림 발생

진수 표현

▪ 10진수, 2진수, 16진수

※ 10진수를 2진수로 바꾸기

10 = 1010(2)

8 4 2 1

1 0 1 0 ($1 \times 2^3 + 1 \times 2^1 \rightarrow 8 + 2$)

```
>>> bin(10)
'0b1010'
>>> bin(15)
'0b1111'
>>> bin(32)
'0b100000'
>>> bin(65)
'0b1000001'
>>> hex(10)
'0xa'
>>> hex(16)
'0x10'
```

```
computer.py - C:/Python/computer.py (3.7.4)
File Edit Format Run Options Window Help

# 10진수, 2진수, 16진수
num = 10
b_num = 0b1010
h_num = 0xA

print(num)
print(b_num)
print(h_num)
```

```
Python 3.7.4 Shell
File Edit Shell Debug Op

Python 3.7.4
(Intel)] on w:
Type "help",
>>>
=====
10
10
10
```

아스키 코드와 유니코드

- 숫자, 문자 표현 – 컴퓨터 내부에서는 숫자뿐만 아니라 문자도 2진수로 표현

- 아스키 코드(ASCII)**

숫자, 영문알파벳 등 128개의 문자를 표기하도록 정한 코드값

- 유니코드(Unicode)**

한글, 중국어 등 아스키 코드로 표현할 수 없는 문자 표기 가능

예) 한글은 약 11,000자

- 총 65536개의 문자 표현 가능

문자	코드값	문자	코드값
	32	A	65
!	33	B	66
"	34	C	67
...
0	48	a	97
1	49	b	98
2	50	c	99

```
>>> chr(35)
'#'
>>> chr(48)
'0'
>>> chr(65)
'A'
```

항과 연산자

■ 항(operand)

- 연산에 사용되는 값

■ 연산자(operator)

- 연산에 사용되는 기호

예) $3 + 7$ (3과 7은 항, '+'는 연산자)



■ 연산자의 종류

구분	연산자	연산 예
대입연산자	'='	num = 10
산술연산자	+, -, *, /, %	3 + 7
비교연산자	>, >=, <, <=, ==, !=	7 == 3
논리연산자	and, or, not	7 >= 3 and 3 != 3
복합대입연산자	+=, -=, *=, /=	n += 1

대입 연산자

■ 대입 연산자

- 오른쪽의 값을 왼쪽의 변수에 대입
- '=' 연산자를 사용.

age = 20

user_id = "abc123"

user_pw = 0000

```
print('*** 회원가입 ***')
userid = 'hangu1'
userpw = 'han1234'
name = '한글'
phone = '010-1234-5678'
age = 35
```

```
print("아이디 :", userid)
print("비밀번호 :", userpw)
print("이름 :", name)
print("전화번호 :", phone)
print("나이 :", age)
```

대입 연산자 연습문제

변수 값 교환하기

변수 blue에 1이 저장되어 있고, red에 2가 저장되어 있을때 새로운 변수 yellow를 사용하여 값을 교환해 보세요

swap.py

```
# 변수값 교환
blue = 1
red = 2
print("교환전: ")
print("blue =", blue, ", red =", red)

#교환 처리
'''
yellow = blue
blue = red
red = yellow
'''

# 파이썬은 직접 교환함
blue, red = red, blue

print("교환후: ")
print("blue =", blue, ", red =", red)
```

산술 연산자

■ 산술 연산자

연산자	연산 작업	설명
+	$n1 + n2$	더하기
-	$n1 - n2$	빼기
*	$n1 * n2$	곱하기
/	$n1 / n2$	나누기
//	$n1 // n2$	몫
%	$n1 \% n2$	나머지
**	$n1 ** n2$	제곱

```
calc-op.py - C:/pyworks/ch02/calc-op.py (3.9.5)
File Edit Format Run Options Window Help

# 산술 연산자 사용하기
print("10 + 4 =", 10 + 4)
print("10 - 4 =", 10 - 4)
print("10 * 4 =", 10 * 4)
print("10 / 4 =", 10 / 4)
print("10 // 4 =", 10 // 4)
print("10 % 4 =", 10 % 4)
print("10 ** 2 =", 10 ** 2)
```

산술 연산자 활용

과목의 총점과 평균을 계산하는 프로그램 작성 하기

```
# 성적의 총점과 평균
# 변수 선언
math_score = 80
eng_score = 85

# 합계
total_score = math_score + eng_score

# 평균
avg_score = total_score / 2

#출력
print("합계 :", total_score)
print("평균 :", avg_score)
```


대입 연산자, 복합대입 연산자

▪ 복합대입연산자

`+=, -=, *=, /=, %=`

```
#청바지 수량  
jean = 0  
jean = 1  
#jean = jean + 1  
jean += 1  
print("구매할 청바지 수량 ", jean)
```

실습 문제

도형의 넓이 계산하기

1. 한 변의 길이가 5cm인 정사각형의 넓이
2. 한 변의 길이가 5cm이고, 높이 7cm인 삼각형의 넓이

figure.py

```
# 정사각형의 넓이
size = 5

area = size * size

print("정사각형의 넓이 : ", area)

# 삼각형의 넓이
width = 5
height = 7

area = (width * height) / 2

print("삼각형의 넓이 : ", area)
```

비교 연산

■ 비교 연산자

연산자	의미	예	결과
<	보다 작다	9 < 10	True
>	보다 크다	9 > 10	False
<=	작거나 같다	9<=10	True
>=	크거나 같다	9>=10	False
==	같다	9==10	False
!=	같지 않다	9!=10	True
is	같다(객체)	a is b	
is not	같지 않다(객체)	a is not b	

```
>>> 1+1
2
>>> 1+1==2
True
>>> 9 > 10
False
>>> 9 <= 10
True
>>> "abc" == "abc"
True
>>> 'abc' == 'ABC'
False
>>> x = 10
>>> y = 20
>>> x is not y
True
>>> x != y
True
>>> x is y
False
```

논리 연산

■ 논리 연산자

연산자	규칙
x and y	x, y 가 모두 참이면 참, 나머지는 거짓
x or y	x, y 중 둘 중 하나가 참이면 참
not x	x가 참이면 거짓, 거짓이면 참

👉 **Python Docs > Library Reference > Built-in Types**

Operation	Meaning
<	strictly less than
<=	less than or equal
>	strictly greater than
>=	greater than or equal
==	equal
!=	not equal
is	object identity
is not	negated object identity

Boolean Operations — and, or, not ¶

These are the Boolean operations, ordered by ascending priority:

Operation	Result	Notes
x or y	if x is false, then y, else x	(1)
x and y	if x is false, then x, else y	(2)
not x	if x is false, then True, else False	(3)

논리 연산

■ 논리 연산자

logic_op.py

비교, 논리 연산자

x = 10

y = -10

print(x > 0)

print(y > 0)

print()

print(x > y)

print(x < y)

print()

print(x == 10)

print(x == y)

print(x != y)

print()

print(x > 0 and y > 0)

print(x > 0 or y > 0)

print(not x > 0)

True
False

True
False

True
False
True

False
True
False

비트 연산

➤ 비트 이동(shift) 연산자

연산자	규칙
$a \ll 2$	a를 왼쪽으로 2비트 이동
$b \gg 3$	b를 오른쪽으로 3비트 이동

원본 데이터

움길 비트의 수

```
>>> num=5
>>> num << 2
20
>>> num >> 2
1
```



```
num      : 0 0 0 0 0 1 0 1
num << 2 : 0 0 0 1 0 1 0 0
num >> 2 : 0 0 0 0 0 0 0 1
```

비트 연산

➤ 비트 논리 연산자

연산자	규칙
9 & 8	9와 8의 비트 논리곱을 수행 (1001 & 1000)
9 8	9와 8의 비트 논리합을 수행 (1001 1000)
~9	1은 0으로 0은 1로 반전

```
>>> num1 = 8
>>> num2 = 9
>>> num1 & num2
8
>>> num1 | num2
9
```



```
num1 : 0 0 0 0 1 0 0 0
& num2 : 0 0 0 0 1 0 0 1
          0 0 0 0 1 0 0 0
```



```
num1 : 0 0 0 0 1 0 0 0
| num2 : 0 0 0 0 1 0 0 1
          0 0 0 0 1 0 0 1
```

문자형 자료형(Type)

▪ 문자열 만들기

- 큰따옴표(")로 양쪽 둘러싸기 – "Hello World"
- 작은 따옴표(')로 양쪽 둘러싸기 – '파이썬은 재미있다'
- 큰따옴표 또는 작은 따옴표 3개를 연속으로 써서 양쪽 둘러싸기

"""Python is very fun."""

▪ 문자열 안에 작은 따옴표나 큰 따옴표 포함

sentence = "'힘내세요' 라고 말했습니다. "

say = "'Python is very easy.' he says.'

문자형 자료형(Type)

- 문자열 안에 따옴표 사용하기 – 서로 중복이 되지 않게 함

```
# 문자열안에 따옴표 포함하기
```

```
say = "'힘내세요!!' 라고 말했습니다."  
print(say)
```

```
say2 = '"Python is very easy." he says.'  
print(say2)
```

quotes.py

문자형 자료형(Type)

- 여러 줄로 출력하기 – 쌍따옴표 또는 홀따옴표 3개 사용("""~~""")

```
# 여러 줄로 출력하기
```

```
song1 = '''  
동해물과 백두산이 마르고 닳도록  
하느님이 보우하사 우리나라 만세  
'''  
print(song1)  
  
song2 = """  
남산위에 저 소나무 철갑을 두른 듯  
바람서리 불변함은 우리 기상 일세  
"""  
print(song2)
```

multiline.py

문자열 연산

■ 문자열 더해서 연결하기(Concatenation)

+ : 연결 연산자

■ 문자열 곱하기

* : 곱하기 연산자

```
>>> head = "Python"
>>> tail = " is fun."
>>> head + tail
'Python is fun.'
>>> str = "안녕"
>>> str*3
'안녕안녕안녕'
```

multistring.py

```
=====
My Program
=====
```

```
print("="*30)
print("My Program")
print("="*30)
```

문자형 자료형

■ 이스케이프 문자 사용하기

코드	내 용
\n	줄바꿈
\t	탭 - 문자열 간격
\b	백스페이스

```
'힘내세요!!'  
라고 말했습니다.
```

```
이름      나이      지역  
최영      65       개성  
세종대왕  50       서울  
이순신    45       평택
```

```
say2 = "'힘내세요!!' \n라고 말했습니다."  
print(say2)
```

```
say3 = """  
이름\t나이\t지역  
최영\t65\t개성  
세종대왕\t50\t서울  
이순신\t45\t평택  
"""  
print(say3)
```

형 변환(Type Conversion)

■ 형 변환

- 자료형은 각각 사용하는 메모리 크기와 방식이 다름
- **int**(문자) : 숫자로 변환함, **str**(숫자) : 문자로 변환함

```
# 성적의 총점과 평균

math_score = 80
eng_score = 85

total_score = math_score + eng_score

# 평균
avg_score = total_score / 2

# 숫자와 문자가 함께 사용될 때 문자로 형 변환됨 (문자 > 숫자)
#print("합계 :", total_score)
print("합계 : " + str(total_score))

#print("평균 :", avg_score)
print("평균 : " + str(avg_score))
```

파이썬의 입력 처리

➤ input() – 입력 함수

```
# 문자 입력
print("문자 입력: ")
ch = input()
print(ch)

#숫자 입력
num = input("숫자 입력 : ")
num = int(num)    #입력한 문자를 숫자로 변환
print(num)
print(num + 5)
```

파이썬의 입력 처리

➤ input() – 입력 함수

```
print("이름을 입력해 주세요:")  
name = input()  
print(name + '님 반갑습니다.')  
age = input('나이를 입력해 주세요: ')  
age = int(age)  
print("당신의 나이는 " + str(age) + "세 이군요!")
```

파이썬의 입력 처리

➤ input() -입력함수

```
print("첫째 수 입력: ")
x = int(input())
print("둘째 수 입력: ")
y = int(input())

sum = x + y
print("두 수의 합은", sum)
```

x는 str형이므로 int형으로
형변환해야함

실습 문제

나이 계산 프로그램

나이를 입력 받아 아래의 결과처럼 계산하는 프로그램을 작성하세요.

👉 실행 결과

태어난 연도를 입력하세요 : 1990
1990년에 태어난 사람의 나이는 32세입니다.

```
current_year = 2021

birth_year = int(input("태어난 연도를 입력하세요 : "))

age = current_year - birth_year + 1

print("당신의 나이는 " + str(age) + "세 입니다.")
```

실습 문제

사각형의 계산하는 프로그램

가로와 세로의 길이를 입력 받아 넓이를 계산하는 프로그램을 작성하세요.

👉 실행 결과

```
가로의 길이 : 4
세로의 길이 : 5
가로 길이 : 4cm
세로 길이 : 5cm
면적 : 20cm
```

사각형의 넓이 구하기

```
width = input("가로의 길이 : ")
height = input("세로의 길이 : ")
area = int(width) * int(height)

print("가로 길이 : " + str(width) + 'cm')
print("세로 길이 : " + str(height) + 'cm')
print("면적 : " + str(area) + 'cm')
```