

Chapter 02

파이썬의 기본 문법

변수란?

- 변수 : 컴퓨터에 데이터가 저장되는 메모리 공간의 위치

변수	컴퓨터 메모리
	...
num	15
x	사과
y	오렌지
	...

두 수의 합 구하기

ex2-1.py

```
a = 20  
b = 30  
c = a + b  
  
print(c)
```

a = 20

변수 a에 20을 저장, 즉 변수 a가 가리키는 메모리 공간에 20을 저장한다는 의미

ex2-1.py

:: 실행 결과

50

변수명 규칙

- 변수명의 기본 규칙은 영문 대소문자, 밑줄(_), 숫자를 조합해서 사용

예) a, b, x, y, i, j, str, animal, Computer, _age, sum, type1, type2, num1, num2, _Fruit 등

- 특수문자(!@#\$%^&*)나 공백 ' '은 변수명에 사용할 수 없음
- 변수명은 숫자로 시작하면 안됨

예) 7person : 숫자로 시작했기 때문에 잘못된 변수명

- 영문에서 대문자와 소문자는 서로 다름

예) 변수명 age와 Age는 서로 다른 변수임

올바른 변수명

ex2-2.py

```
x = 20
Computer = 'Mac'
Age = 30
My_score = 70
_name = '홍길동'
myBirthYear = 1997
data2 = 20.3

print(x, Computer, Age, my_score, _name, myBirthYear, data2)
```

:: 실행 결과

20 Mac 30 70 홍길동 1997 20.3

잘못된 변수명

ex2-3.py

```
eng score = 90
7font = '굴림'
my_age = 20
percent% = 100
animal# = '사슴'
```

:: 실행 결과

File "<ipython-input-10-ebf715fab605>", line 1
eng score = 90 ^

SyntaxError: invalid syntax

자주 쓰이는 데이터 형

- ① 숫자 : -30, -46656, 0, 23, 333, -0.3737, -376.0, 0, 3.14, 256.333
- ② 문자열 : 'a', 'b', 'abc', 'apple', 'I am happy!', “안녕하세요”
- ③ 불 : True, False
- ④ 리스트 : ['홍길동', 32, '010-2222-3333', 'hong@korea.com']
- ⑤ 튜플 : ('짜장면', '짬뽕', '탕수육', '우동')
- ⑥ 딕셔너리 : {'red': '빨간색', 'yellow': '노란색', 'blue': '파란색', 'green': '초록색'}

숫자

- 정수형(Integer) : 음수, 0, 양수로 구성된 숫자

예) 33, -44, -3, 0, 7, 88, 38378 등

- 실수형(Floating Point, 부동 소수점) : 소수점이 있는 숫자

예) -333.44, -38.333, 0.0, 3.14, 3663.373 등

정수형

ex2-4.py

```
x = 30  
print(x)  
print(type(x))
```

Type() 함수는 변수나 데이터의 데이터 형을
구하는 데 사용

함수 : Function, 어떤 기능을 수행
예) print() 함수 : 화면에 출력 기능
type() 함수 : 데이터 형을 구하는 기능

:: 실행 결과

```
30  
<class 'int'>
```

실수형

ex2-5.py

```
x = 3.3764  
y = 6/2  
print(x, y)  
print(type(x), type(y))
```

:: 실행 결과

```
3.3764 3.0  
<class 'float'> <class 'float'>
```

문자열

- 문자열(String) :

- 하나 또는 여러 개의 문자로 구성됨
- 문자열의 앞 뒤에는 **단따옴표(')** 또는 **쌍따옴표(")**로 감쌘
- 예) '가 ' , ' 가다나 ' , ' a ' , ' b ' , ' abc ' , ' I am happy! ' ,
' 2020/10/20 ' , '010-1234-5678 ' , "apple",
"사자와 호랑이", "###", "%^&^%&%", '30', '5', '-10' 등

문자열의 데이터 형

ex2-6.py

```
a = 'x'
b = 'I am ok.'
c = "안녕하세요."

print(a)
print(b)
print(c)
print(type(c))
```

:: 실행 결과

```
x
I am ok.
안녕하세요.
<class 'str'>
```

30과 '30'의 차이

ex2-7.py

```
a = 30
print(a)
print(type(a))

b = '30'
print(b)
print(type(b))
```

30 : 정수형 숫자
10진수가 2진수로 변환된 형태로 저장

'30' : 문자열
'3'과 '0'의 컴퓨터 코드(예: 아스키 코드)가 저장
※ **키보드에 입력되는 데이터는 문자열로 저장됨.**

:: 실행 결과

```
30
<class 'int'>
30
<class 'str'>
```

문자열의 인덱스

ex2-8.py

```
x = 'I am happy!'
```

```
print(x)
```

```
print(x[0])
```

```
print(x[0:3])
```

```
print(x[5:])
```

```
print(x[-1])
```

```
print(x[-3:])
```

```
print(x[-4:-2])
```

인덱스란?

- 인덱스(Index)는 문자열 요소의 위치를 가리킴
- 인덱스는 0부터 시작

:: 실행 결과

```
I am happy!
```

```
I
```

```
I a
```

```
happy!
```

```
!
```

```
py!
```

```
pp
```

불 데이터 형

ex2-9.py

```
a = True
```

```
b = False
```

```
print(a)
```

```
print(b)
```

```
c = 10 > 20
```

```
print(c)
```

```
print(type(a))
```

불(Bool)이란?

- 참 : True
- 거짓 : False

:: 실행 결과

```
True
```

```
False
```

```
False
```

```
<class 'bool'>
```

산술 연산자

산술 연산자	설명
+	더하기
-	빼기
*	곱하기
/	나누기
%	나머지 연산
//	소수점 이하 절삭
**	거듭제곱 구하기

사칙연산자

ex2-10.py

```
a = 10
```

```
b = 20
```

```
c = a + b * 10 - 5 / 5
```

```
print(c)
```

:: 실행 결과

209.0

나머지와 소수점 절삭 연산자

ex2-11.py

```
x = 10 % 3
```

```
print(x)
```

```
y = 7//3
```

```
print(y)
```

:: 실행 결과

1

2

거듭제곱 연산자

ex2-12.py

```
x = 2**3
```

```
print(x)
```

```
y = 10**4
```

```
print(y)
```

:: 실행 결과

8

10000

할당 연산자 : +=

ex2-13.py

```
x = 10  
x += 20      # x = x + 20 과 동일  
print(x)
```

:: 실행 결과

30

할당 연산자 : *=

ex2-14.py

```
x = 3  
y = 5  
x *= x + y    # x = x * (x + y) 과 동일  
print(x)
```

:: 실행 결과

24

할당 연산자

할당 연산자	사용 예	설명
=	$x = 2$	x에 2를 저장한다.
+=	$x += 3$	x의 값에 3을 더해서 얻은 값을 다시 x에 저장한다. ※ $x = x + 3$ 와 동일한 표현임.
-=	$x -= 2$	x의 값에 2를 빼서 얻은 값을 다시 x에 저장한다. ※ $x = x - 2$ 와 동일한 표현임.
*=	$x *= 3$	x의 값에 3을 곱해서 얻은 값을 다시 x에 저장한다. ※ $x = x * 3$ 와 동일한 표현임.
/=	$x /= 3$	x의 값을 3으로 나누어 얻은 값을 다시 x에 저장한다. ※ $x = x / 3$ 와 동일한 표현임.
%=	$x \% = 4$	x를 4로 나눈 나머지 값을 다시 x에 저장한다. ※ $x = x \% 4$ 와 동일한 표현임.
**=	$x ** = 3$	x의 3승을 구한 값을 다시 x에 저장한다. ※ $x = x ** 3$ 와 동일한 표현임.

숫자 연산자

Q2-3.py

```
x = 20
```

```
y = 4
```

```
x = y % x
```

```
print(x)
```

```
y -= x * 2
```

```
print(x, y)
```

:: 실행 결과

① _____
② _____ ③ _____

문자열 반복 : *

ex2-15.py

```
hello = '안녕' * 5
```

```
print(hello)
```

:: 실행 결과

안녕안녕안녕안녕안녕

문자열 길이 : len()

ex2-16.py

```
a = '쥐 구멍에 별 들 날 있다.'
```

```
b = len(a)
```

```
print(b)
```

:: 실행 결과

15

문자열 연결 : +

ex2-17.py

```
name = '홍지수'  
greet = name + '님 안녕하세요!'  
print(greet)
```

:: 실행 결과

홍지수님 안녕하세요!

문자열 연결 시 오류

ex2-18.py

```
eng = 80
result = '영어 점수 : ' + eng + '점'
print(result)
```

:: 실행 결과

```
TypeError                                Traceback (most recent call last)
<ipython-input-58-8b1847209801> in <module>
      1 # 파일명 : ex2-14.py
      2 eng = 80
----> 3 result = '영어 점수 : ' + eng + '점'
      4 print(result)

TypeError: can only concatenate str (not "int") to str
```

정수 -> 문자열 : str()

ex2-19.py

```
eng = 80  
result = '영어 점수 : ' + str(eng) + '점'  
print(result)
```

:: 실행 결과

영어 점수 : 80점

문자열 포매팅 : %s

ex2-20.py

```
name = '김수영'  
a = '나는 %s입니다.' % name  
print(a)
```

:: 실행 결과

나는 김수영입니다.

문자열 포매팅 : %d

ex2-21.py

```
age = 20  
a = '나이는 %d살 입니다.' % age  
print(a)
```

:: 실행 결과

나이는 20살 입니다.

문자열 포매팅 : %2d

ex2-22.py

```
year = 2020
```

```
month = 3
```

```
day = 5
```

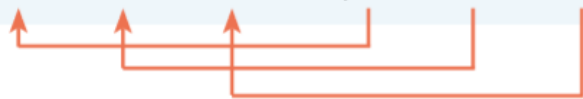
```
a = '%d-%02d-%02d' % (year, month, day)
```

```
print(a)
```

:: 실행 결과

2020-03-05

'%d-%02d-%02d' % (year, month, day)



문자열 포매팅 : %.2f

ex2-23.py

```
height = 172.5  
a = '키는 %.2f입니다.' % height  
  
print(a)
```

:: 실행 결과

키는 172.50입니다.

문자열 포매팅 기호

포맷 기호	사용되는 데이터 형
%d	정수형 숫자
%s	문자열
%f	실수형 숫자
%05d	정수형 숫자 다섯 자리, 남는 부분을 0으로 채움.
%.2f	소수점 둘째 자리의 실수형 숫자

Format() 이용한 문자열 포매팅

ex2-24.py

```
name = '황예린'
age = 18
eyesight = 1.2

a = '이름 : {}'.format(name)
b = '나이 : {}세'.format(age)
c = '시력 : {}'.format(eyesight)

print(a)
print(b)
print(c)
```

:: 실행 결과

```
이름 : 황예린
나이 : 18세
시력 : 1.2
```

문자열 메소드

기호	설명
str.format()	문자열 포매팅하기
str.count()	문자열 개수 세기
str.find()	문자열 내부의 특정 문자열 찾기
str.upper()	문자열 대문자로 바꾸기
str.lower()	문자열 소문자로 바꾸기
str.replace()	문자열 내부의 특정 문자열 바꾸기
str.split()	문자열 분리하기
str.isspace()	문자열에 공백이 있는지 체크하기

실행 결과 화면 출력 : print()

- ① 콤마(,)를 이용한 출력
- ② 문자열 연결 기호(+)를 이용한 출력
- ③ 문자열 포매팅(%)을 이용한 출력
- ④ format()을 이용한 출력
- ⑤ 키워드 sep을 이용한 출력
- ⑥ 키워드 end를 이용한 출력

콤마(,)를 이용한 출력

ex2-25.py

```
name = '홍길동'  
age = 30  
height = 173.7  
print(name, age, height)
```

:: 실행 결과

홍길동 30 173.7

문자열 연결기호(+)를 이용한 출력

ex2-26.py

```
x = 10  
y = 20  
  
print('x = ' + str(x) + ', y = ' + str(y))
```

:: 실행 결과

x = 10, y = 20

문자열 포매팅(%)을 이용한 출력

ex2-27.py

```
score1 = 80
score2 = 87

sum = score1 + score2
avg = sum/2

print('두 과목 점수 : %d, %d' % (score1, score2))
print('합계 : %d, 평균 : %.2f' % (sum, avg))
```

:: 실행 결과

두 과목 점수 : 80, 87
합계 : 167, 평균 : 83.50

키워드 sep을 이용한 출력

ex2-29.py

```
year = 2020  
month = 3  
day = 5  
  
print(year, month, day, sep='/')
```

:: 실행 결과

2020/3/5

- sep : separator

키워드 end를 이용한 출력

ex2-30.py

```
a = '안녕하세요.'  
b = '반갑습니다.'  
  
print(a)  
print(b)  
  
print('\n\n')  
  
print(a, end="")  
print(b)
```

:: 실행 결과

안녕하세요.
반갑습니다.

안녕하세요.반갑습니다.

- " 또는 "" : NULL

널(Null)과 개행문자(\n)

- Null이란?

빈 문자열을 뜻하는 것으로 파이썬에서는 Null에 해당 되는 것으로 **None**을 사용함

- 개행 문자(\n)란?

파이썬에서 **줄바꿈**을 할 때는 개행 문자(New Line Character)가 사용되는데 개행 문자는 다른 말로 라인 피드(Line Feed)라고도 부름

키보드 입력

ex2-31.py

```
name = input('이름을 입력하세요 : ')
```

```
print('%s님 반갑습니다.' % name)
```

:: 실행 결과

```
이름을 입력하세요 : 홍길동  
홍길동님 반갑습니다.
```

키보드 입력받은 두 수 더하기

ex2-33.py

```
a = input('첫 번째 숫자를 입력하세요 :')  
b = input('두 번째 숫자를 입력하세요 :')  
  
c = int(a) + int(b)  
  
print(c)
```

```
a = int(input('첫 번째 숫자를 입력하세요 :'))  
b = int(input('두 번째 숫자를 입력하세요 :'))  
c = a + b
```

:: 실행 결과

```
첫 번째 숫자를 입력하세요 : 10  
두 번째 숫자를 입력하세요 : 20  
30
```

인치(inch)를 센티미터(cm)로 변환

ex2-34.py

```
inch = float(input('인치(inch)를 입력하세요 : '))
```

```
cm = inch * 2.54
```

```
print('센티미터 : %.2f' % cm)
```

:: 실행 결과

```
인치(inch)를 입력하세요 : 32.8  
센티미터 : 83.31
```

주석문

ex2-35.py

```
# 파일명 : ex2-35.py
```

```
"""
```

```
프로그램명 : 두 수 더하기
```

```
작성자 : 홍길동
```

```
작성일 : 2021.9.20
```

```
"""
```

```
a = 10      # 변수 a에 10 저장
```

```
b = 20      # 변수 b에 20 저장
```

```
c = a + b   # 두 수를 더해 변수 c에 저장
```

```
# 결과를 출력한다.
```

```
print(c)
```

:: 실행 결과

30

삼각형의 면적 구하기

Q2-1.py

```
width = 10
height = 3
area = ❶ _____ * height / 2
print('삼각형의 밑변 길이 :', width)
print('삼각형의 높이 :', ❷ _____)
print('삼각형의 면적 :', ❸ _____)
```

:: 실행 결과

```
삼각형의 밑변 길이 : 10
삼각형의 높이 : 3
삼각형의 면적 : 15.0
```

거스름돈 계산하기

Q2-2.py

```
price = 800
buy = 3
pay = 5000
❶ _____ = ❷ _____ - price * buy
print('물건가격 : ', price, '원')
print('구매개수 : ', ❸ _____, '개')
print('지불금액 : ', pay, '원')
print('거스름돈 : ', change, '원')
```

:: 실행 결과

물건가격 : 800 원
구매개수 : 3 개
지불금액 : 5000 원
거스름돈 : 2600 원

다음 프로그램의 실행 결과는?

Q2-3.py

```
x = 20
```

```
y = 4
```

```
x = y % x
```

```
print(x)
```

```
y -= x * 2
```

```
print(x, y)
```

:: 실행 결과

문자열 추출과 문자열 포매팅

Q2-4.py

```
name = "홍길동"
```

```
age = 30
```

```
height = 171.5
```

```
print('성:', name[❶_____])
```

```
print('이름:', name[❷_____])
```

```
print("❸_____의 나이 : ❹_____세, 키 : ❺_____cm" % (name,  
age, height))
```

```
print("❸_____의 나이 : ❹_____세, 키 : ❺_____cm" % (name,  
age, height))
```

:: 실행 결과

성 : 홍

이름 : 길동

홍길동의 나이 : 30세, 키 : 171.50cm

성적 평균 구하기

Q2-5.py

```
name = input('학생 이름을 입력하세요 :')  
kor = ❶_____(input('국어 성적을 입력하세요 :'))  
eng = ❶_____(input('영어 성적을 입력하세요 :')) math  
= ❶_____(input('수학 성적을 입력하세요 :'))  
  
❷_____ = kor + eng + math  
❸_____ = total / ❹_____  
  
print('이름:❺_____, 국어:%d, 영어:%d,수학:%d, 평균:%.1f  
점' % (name, kor, eng, math, avg))
```

:: 실행 결과

```
학생 이름을 입력하세요 :황예린  
국어 성적을 입력하세요 :83  
영어 성적을 입력하세요 :85  
수학 성적을 입력하세요 :91  
이름:황예린, 국어:83, 영어:85,수학:91, 평균:86.3  
점
```