Python Programming

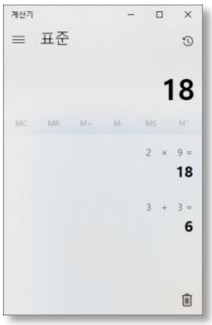
변수, 연산자

권기웅

1. 변수란

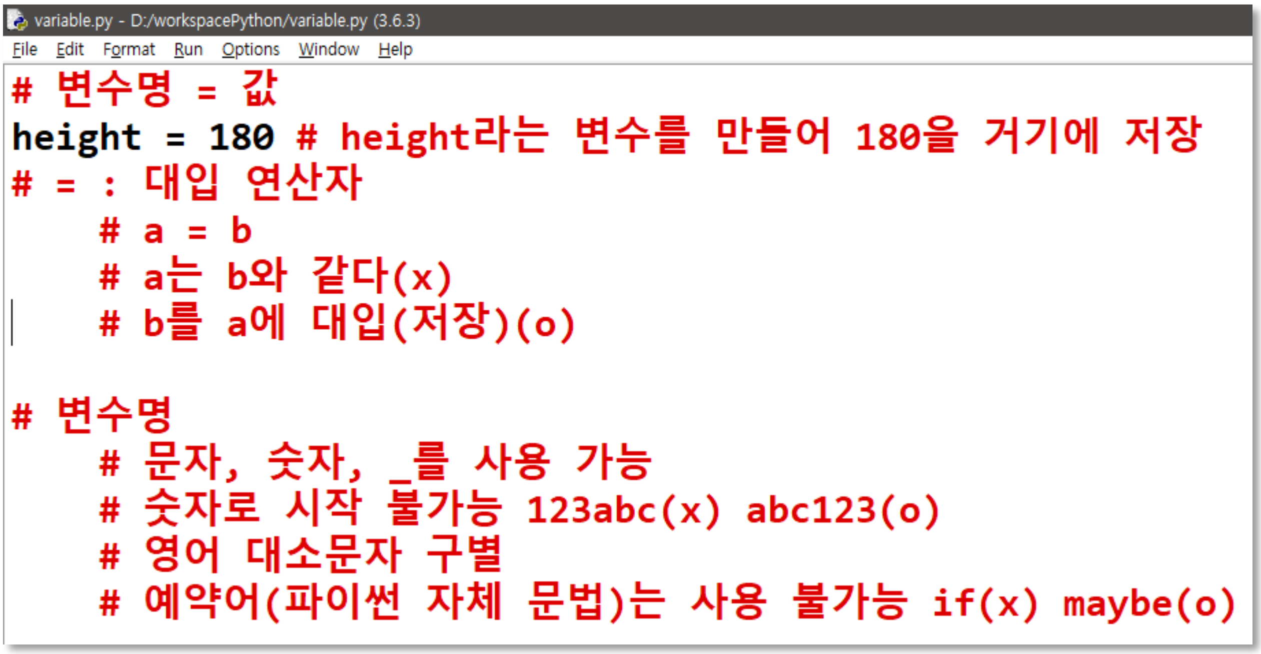
1.1 변수란

프로그램은 진행되는 도중에 값들을 임시로 저장했다가 나중에 찾아서 사용해야 하는 순간이 생기는데(ex)계산기의 히스토리 기능)



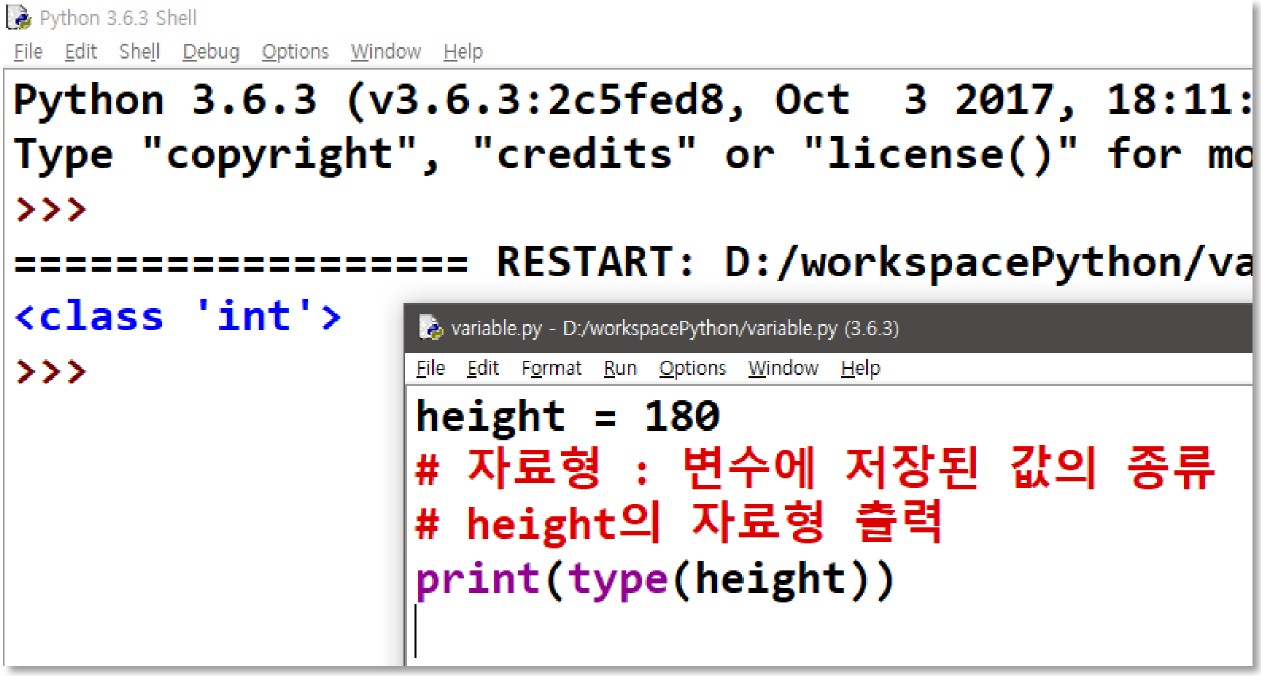
그 임시로 값들이 저장되는 공간을 변수라고 하며, 개발자가 필요할 때 변수를 만들어서 사용

1.2 변수 만들기


variable.py - D:/workspacePython/variable.py (3.6.3)
File Edit Format Run Options Window Help

```
# 변수명 = 값
height = 180 # height라는 변수를 만들어 180을 거기에 저장
# = : 대입 연산자
    # a = b
    # a는 b와 같다(x)
    # b를 a에 대입(저장)(o)

# 변수명
    # 문자, 숫자, _를 사용 가능
    # 숫자로 시작 불가능 123abc(x) abc123(o)
    # 영어 대소문자 구별
    # 예약어(파이썬 자체 문법)는 사용 불가능 if(x) maybe(o)
```

1.3 자료형 확인



Python 3.6.3 Shell
File Edit Shell Debug Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:
Type "copyright", "credits" or "license()" for mo
>>>
=================== RESTART: D:/workspacePython/va
<class 'int'>
>>>
```

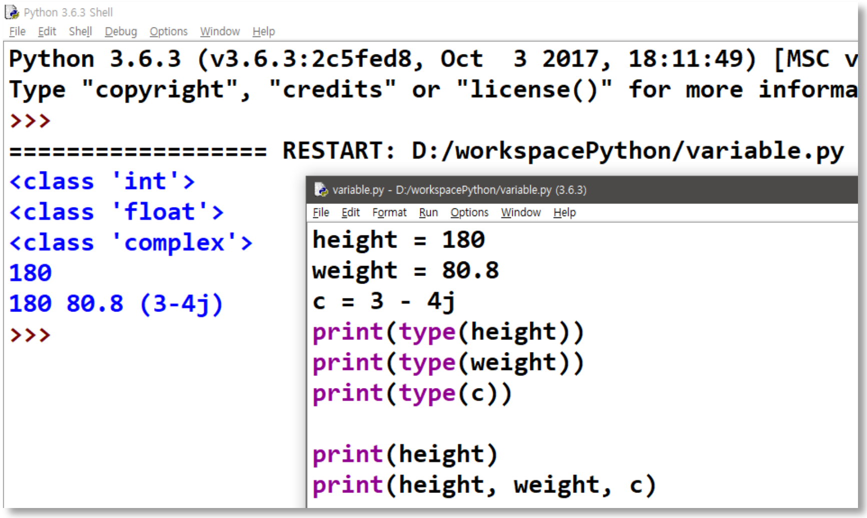
variable.py - D:/workspacePython/variable.py (3.6.3)
File Edit Format Run Options Window Help

```
height = 180
# 자료형 : 변수에 저장된 값의 종류
# height의 자료형 출력
print(type(height))
```

2. 변수

2.1 숫자형

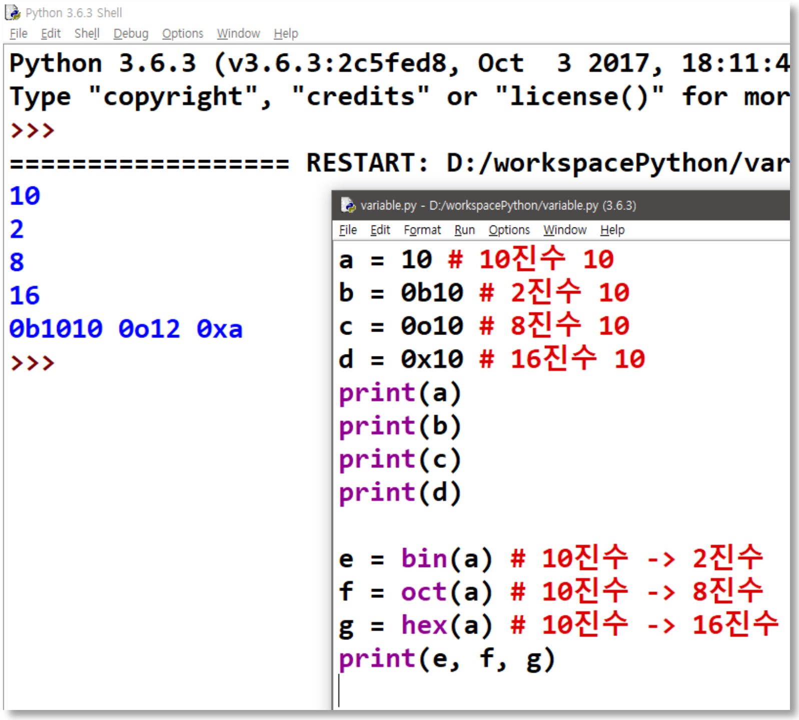
2.1.1 int, float, complex


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:/workspacePython/variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v
Type "copyright", "credits" or "license()" for more informa
>>>
================== RESTART: D:/workspacePython/variable.py
<class 'int'>
<class 'float'>
<class 'complex'>
180
180 80.8 (3-4j)
>>>
```

```
height = 180
weight = 80.8
c = 3 - 4j
print(type(height))
print(type(weight))
print(type(c))

print(height)
print(height, weight, c)
```

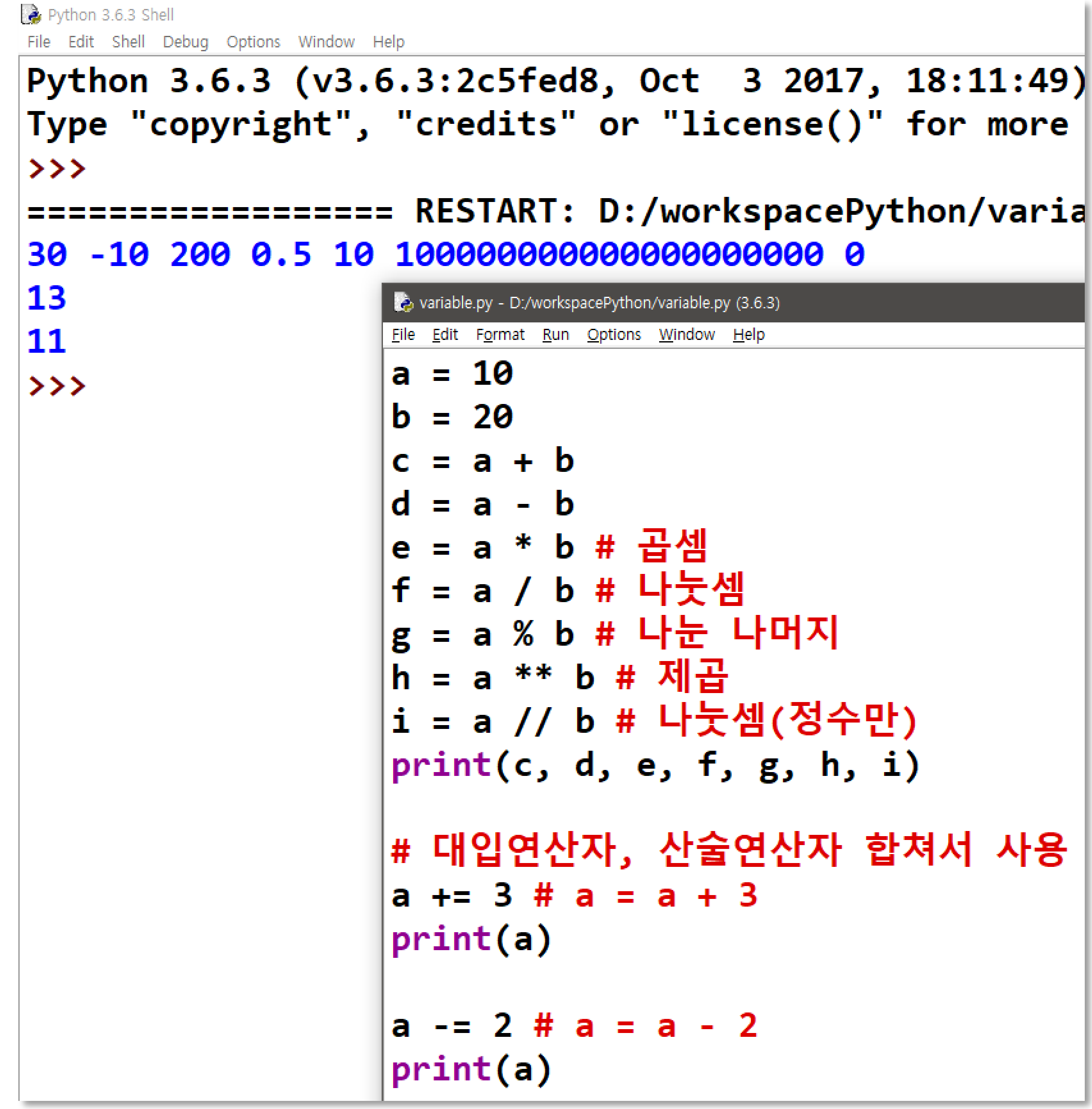
2.1.2 진법변환


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:/workspacePython/variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:4
Type "copyright", "credits" or "license()" for mor
>>>
================== RESTART: D:/workspacePython/var
10
2
8
16
0b1010 0o12 0xa
>>>
```

```
a = 10 # 10진수 10
b = 0b10 # 2진수 10
c = 0o10 # 8진수 10
d = 0x10 # 16진수 10
print(a)
print(b)
print(c)
print(d)

e = bin(a) # 10진수 -> 2진수
f = oct(a) # 10진수 -> 8진수
g = hex(a) # 10진수 -> 16진수
print(e, f, g)
```

2.1.3 산술연산자



Python 3.6.3 Shell
File Edit Shell Debug Options Window Help

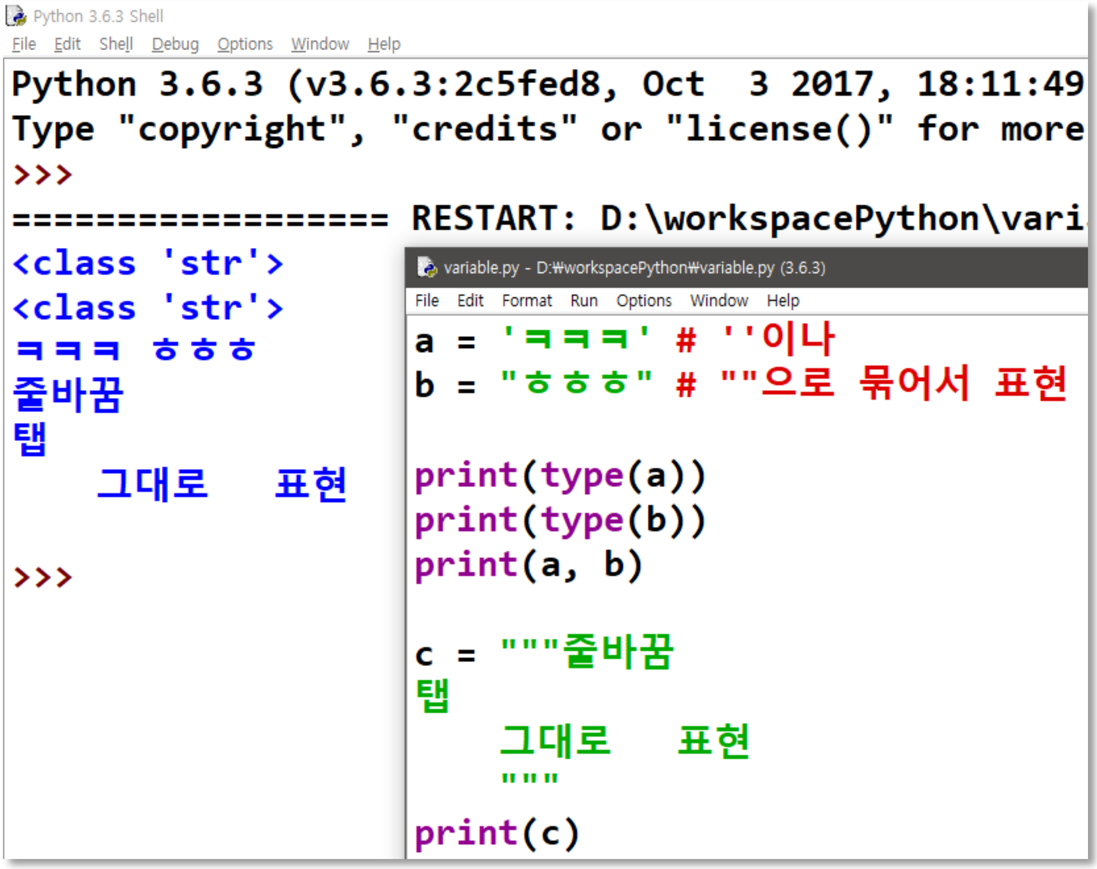
```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49)
Type "copyright", "credits" or "license()" for more
>>> 
================== RESTART: D:/workspacePython/varia
30 -10 200 0.5 10 100000000000000000000 0
13
11
>>> 
```

variable.py - D:/workspacePython/variable.py (3.6.3)
File Edit Format Run Options Window Help

```python
a = 10
b = 20
c = a + b
d = a - b
e = a * b # 곱셈
f = a / b # 나눗셈
g = a % b # 나눈 나머지
h = a ** b # 제곱
i = a // b # 나눗셈(정수만)
print(c, d, e, f, g, h, i)

# 대입연산자, 산술연산자 합쳐서 사용
a += 3 # a = a + 3
print(a)

a -= 2 # a = a - 2
print(a)
```

2.2 문자형

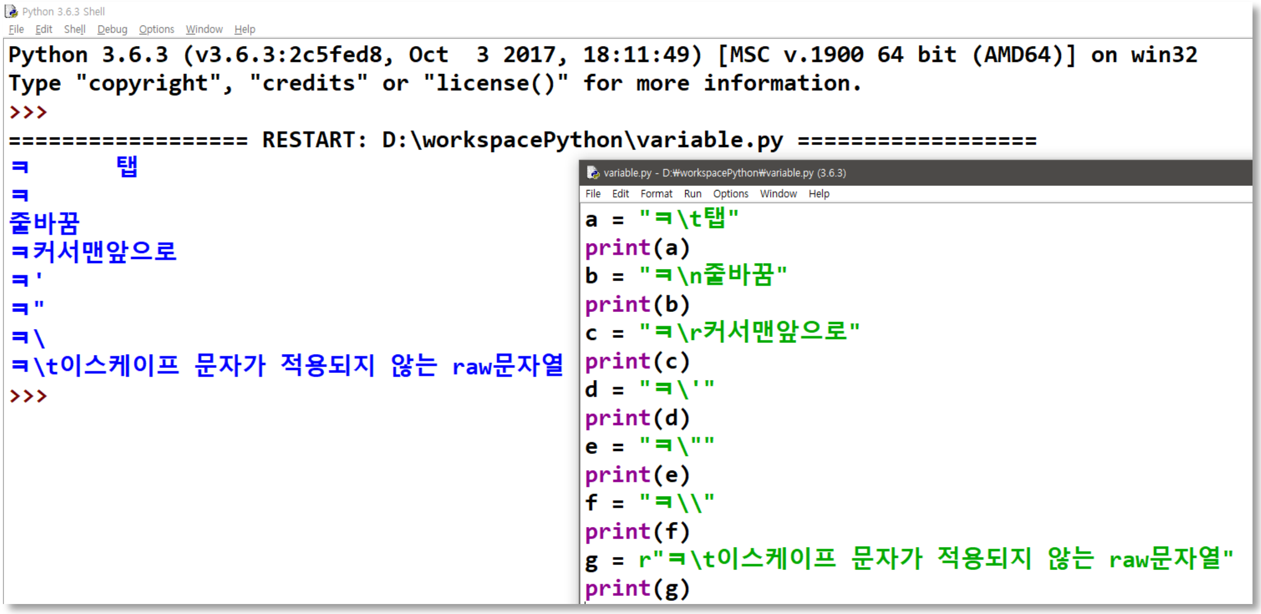
2.2.1 str


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49
Type "copyright", "credits" or "license()" for more
>>>
================== RESTART: D:\workspacePython\vari
<class 'str'>
<class 'str'>
ㅋㅋㅋ ㅎㅎㅎ
줄바꿈
탭
    그대로    표현

>>>
```

```
a = 'ㅋㅋㅋ' # ''이나
b = "ㅎㅎㅎ" # ""으로 묶어서 표현

print(type(a))
print(type(b))
print(a, b)

c = """줄바꿈
탭
    그대로    표현
    """

print(c)
```

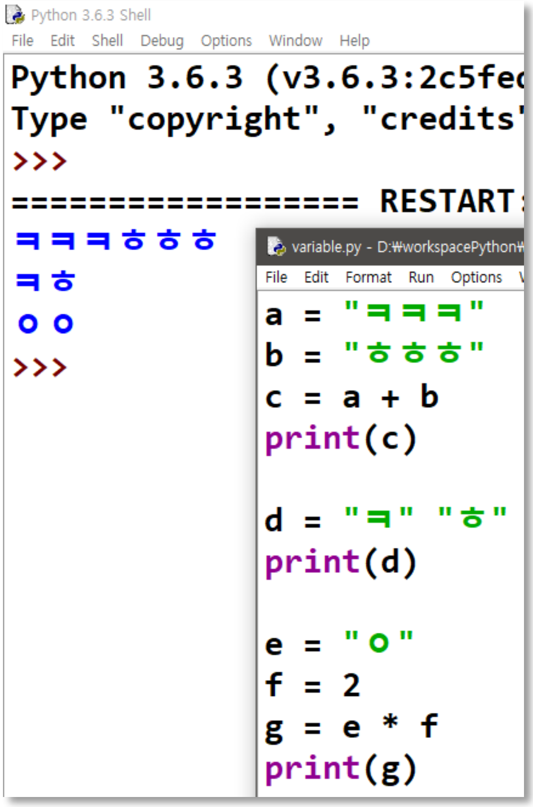
2.2.2 이스케이프 문자


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
================= RESTART: D:\workspacePython\variable.py =================
ㅋ       탭
ㅋ
줄바꿈
ㅋ커서맨앞으로
ㅋ'
ㅋ"
ㅋ\
ㅋ\t이스케이프 문자가 적용되지 않는 raw문자열
>>>
```

```
a = "ㅋ\t탭"
print(a)
b = "ㅋ\n줄바꿈"
print(b)
c = "ㅋ\r커서맨앞으로"
print(c)
d = "ㅋ\'"
print(d)
e = "ㅋ\""
print(e)
f = "ㅋ\\"
print(f)
g = r"ㅋ\t이스케이프 문자가 적용되지 않는 raw문자열"
print(g)
```

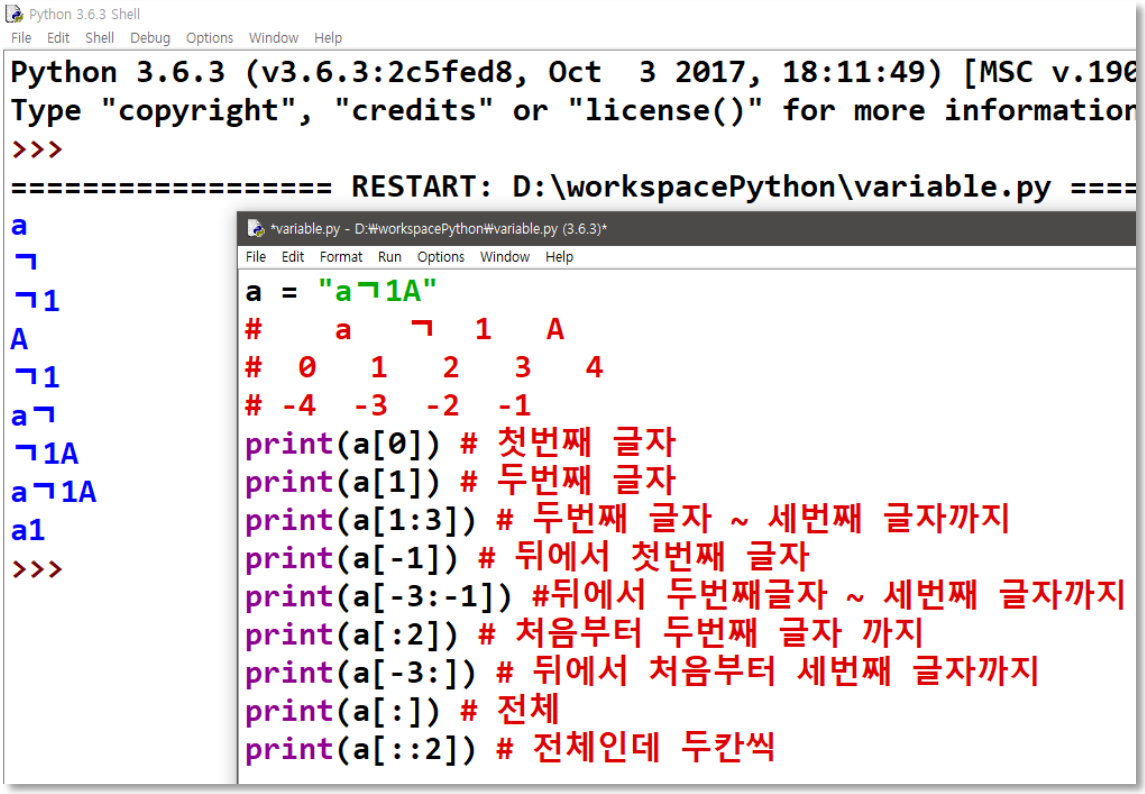
2.2.3 연산자


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩
File Edit Format Run Options

```
Python 3.6.3 (v3.6.3:2c5fed
Type "copyright", "credits'
>>>
================== RESTART:
ㅋㅋㅋㅎㅎㅎ
ㅋㅎ
ㅇㅇ
>>>
```

```
a = "ㅋㅋㅋ"
b = "ㅎㅎㅎ"
c = a + b
print(c)

d = "ㅋ" "ㅎ"
print(d)

e = "ㅇ"
f = 2
g = e * f
print(g)
```

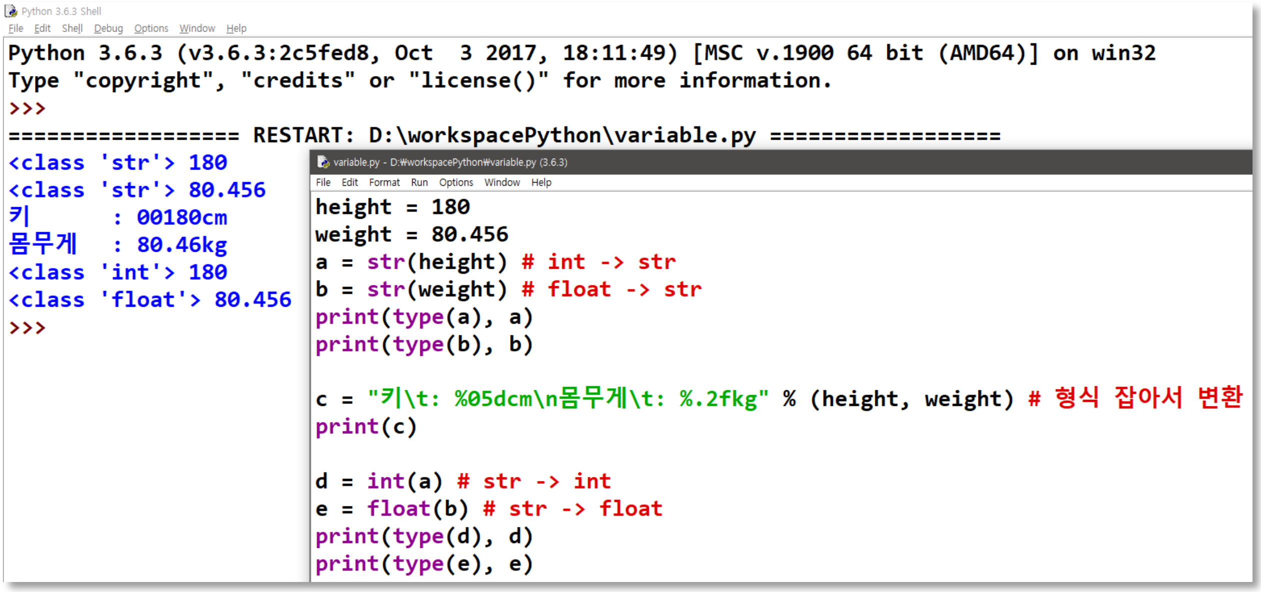
2.2.4 인덱싱


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.190
Type "copyright", "credits" or "license()" for more information
>>>
================== RESTART: D:\workspacePython\variable.py ====
a
ㄱ
ㄱ1
A
ㄱ1
aㄱ
ㄱ1A
aㄱ1A
a1
>>>
```

```
a = "aㄱ1A"
#   a   ㄱ  1   A
#  0   1   2   3   4
# -4  -3  -2  -1
print(a[0]) # 첫번째 글자
print(a[1]) # 두번째 글자
print(a[1:3]) # 두번째 글자 ~ 세번째 글자까지
print(a[-1]) # 뒤에서 첫번째 글자
print(a[-3:-1]) #뒤에서 두번째글자 ~ 세번째 글자까지
print(a[:2]) # 처음부터 두번째 글자 까지
print(a[-3:]) # 뒤에서 처음부터 세번째 글자까지
print(a[:]) # 전체
print(a[::2]) # 전체인데 두칸씩
```

2.2.5 형변환

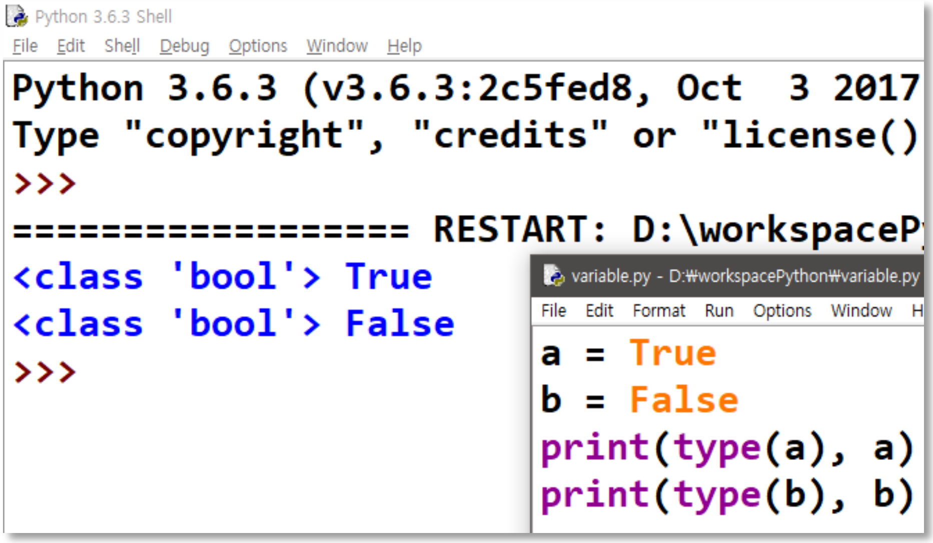

Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> 
================== RESTART: D:\workspacePython\variable.py ==================
<class 'str'> 180
<class 'str'> 80.456
키      : 00180cm
몸무게   : 80.46kg
<class 'int'> 180
<class 'float'> 80.456
>>> 
```

```python
height = 180
weight = 80.456
a = str(height) # int -> str
b = str(weight) # float -> str
print(type(a), a)
print(type(b), b)

c = "키\t: %05dcm\n몸무게\t: %.2fkg" % (height, weight) # 형식 잡아서 변환
print(c)

d = int(a) # str -> int
e = float(b) # str -> float
print(type(d), d)
print(type(e), e)
```

2.3 논리형

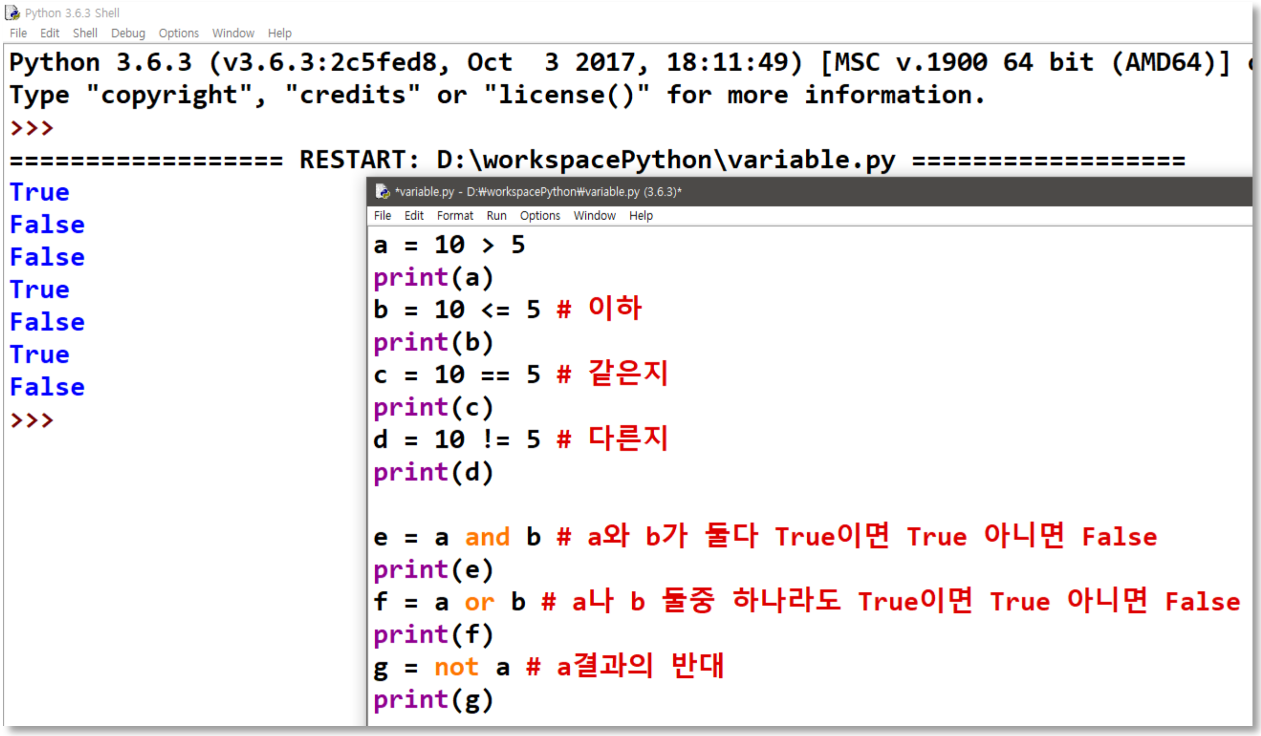
2.3.1 bool


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py
File Edit Format Run Options Window H

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017
Type "copyright", "credits" or "license()
>>> 
=================== RESTART: D:\workspaceP
<class 'bool'> True
<class 'bool'> False
>>>
```

```
a = True
b = False
print(type(a), a)
print(type(b), b)
```

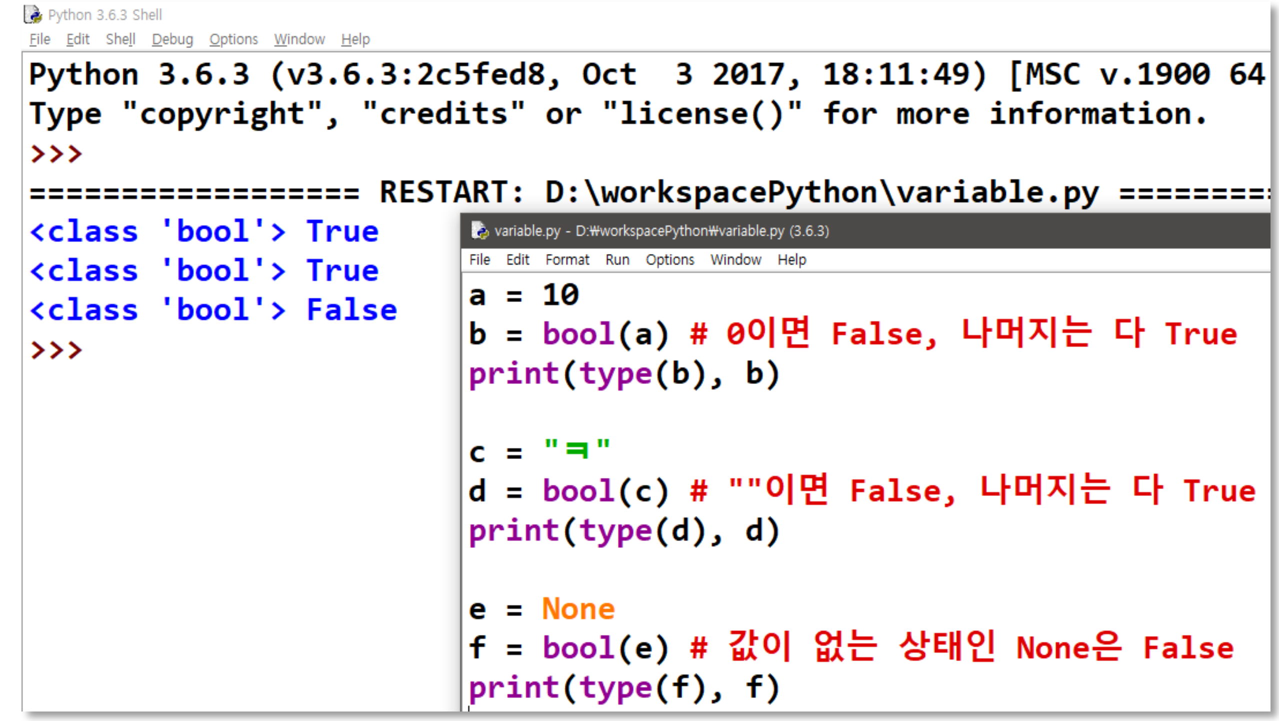
2.3.2 비교연산자, 논리연산자


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)] o
Type "copyright", "credits" or "license()" for more information.
>>> 
================= RESTART: D:\workspacePython\variable.py =================
True
False
False
True
False
True
False
>>>
```

```
a = 10 > 5
print(a)
b = 10 <= 5 # 이하
print(b)
c = 10 == 5 # 같은지
print(c)
d = 10 != 5 # 다른지
print(d)

e = a and b # a와 b가 둘다 True이면 True 아니면 False
print(e)
f = a or b # a나 b 둘중 하나라도 True이면 True 아니면 False
print(f)
g = not a # a결과의 반대
print(g)
```

2.3.3 형변환


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

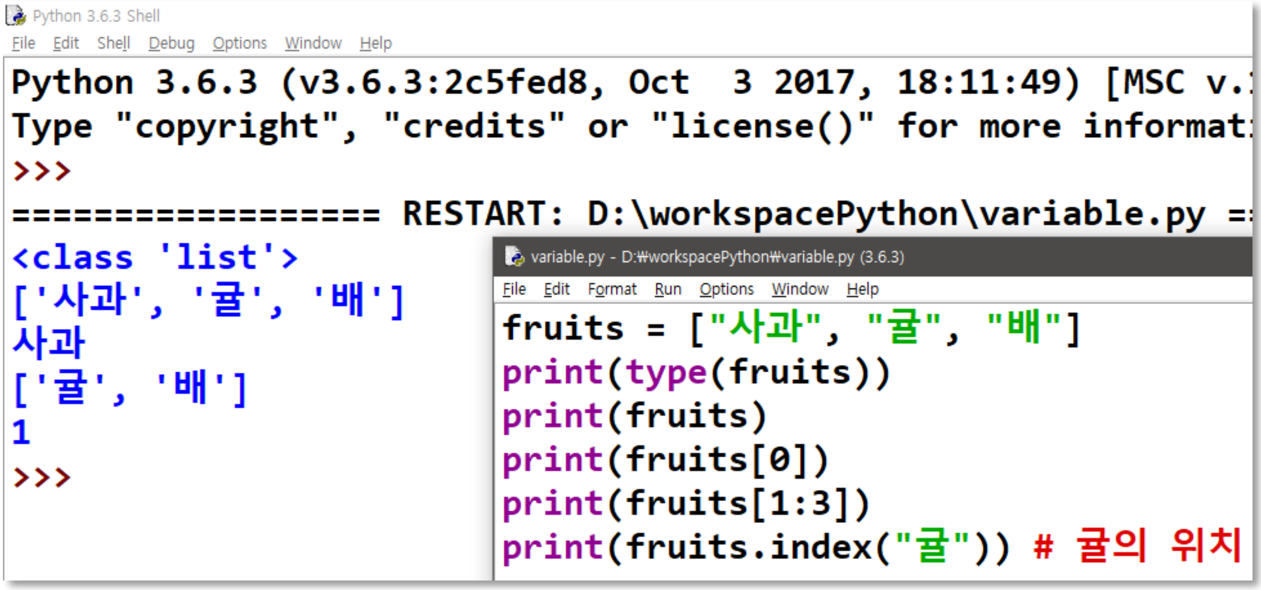
```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64
Type "copyright", "credits" or "license()" for more information.
>>> 
================== RESTART: D:\workspacePython\variable.py ========
<class 'bool'> True
<class 'bool'> True
<class 'bool'> False
>>>
```

```
a = 10
b = bool(a) # 0이면 False, 나머지는 다 True
print(type(b), b)

c = "ㅋ"
d = bool(c) # ""이면 False, 나머지는 다 True
print(type(d), d)

e = None
f = bool(e) # 값이 없는 상태인 None은 False
print(type(f), f)
```

3. 컬렉션

3.1 list

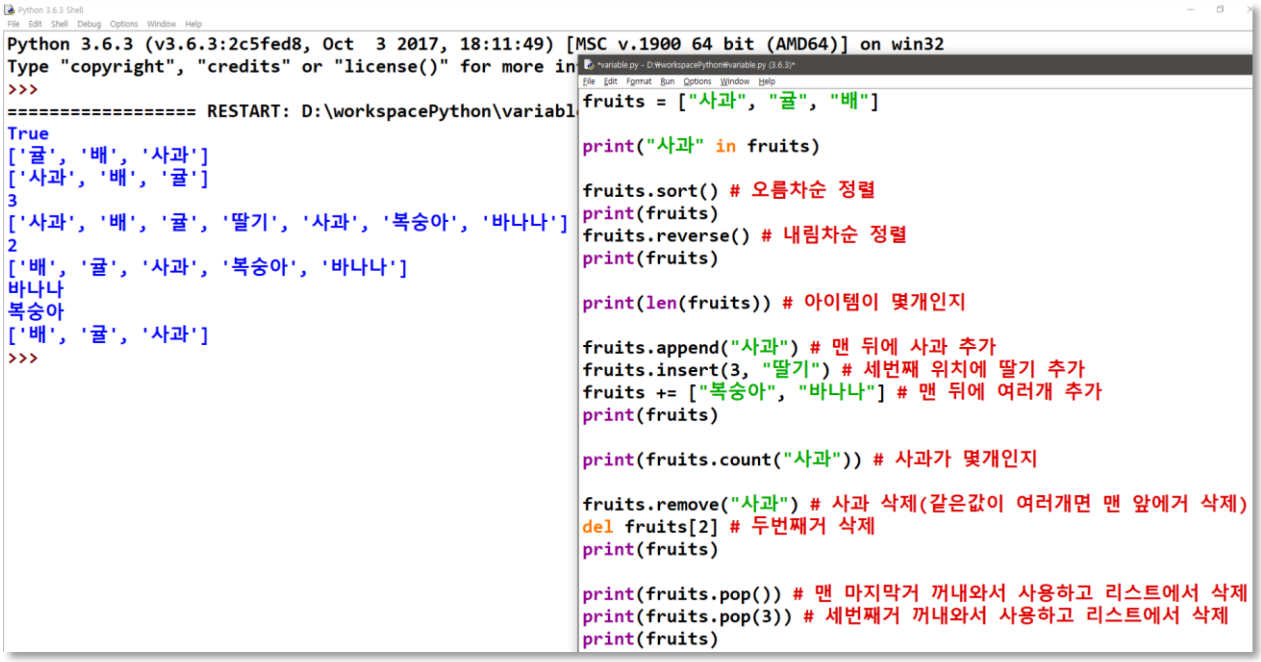
3.1.1 기본사용


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.
Type "copyright", "credits" or "license()" for more informat
>>> 
================== RESTART: D:\workspacePython\variable.py ==
<class 'list'>
['사과', '귤', '배']
사과
['귤', '배']
1
>>> 
```

```python
fruits = ["사과", "귤", "배"]
print(type(fruits))
print(fruits)
print(fruits[0])
print(fruits[1:3])
print(fruits.index("귤")) # 귤의 위치
```

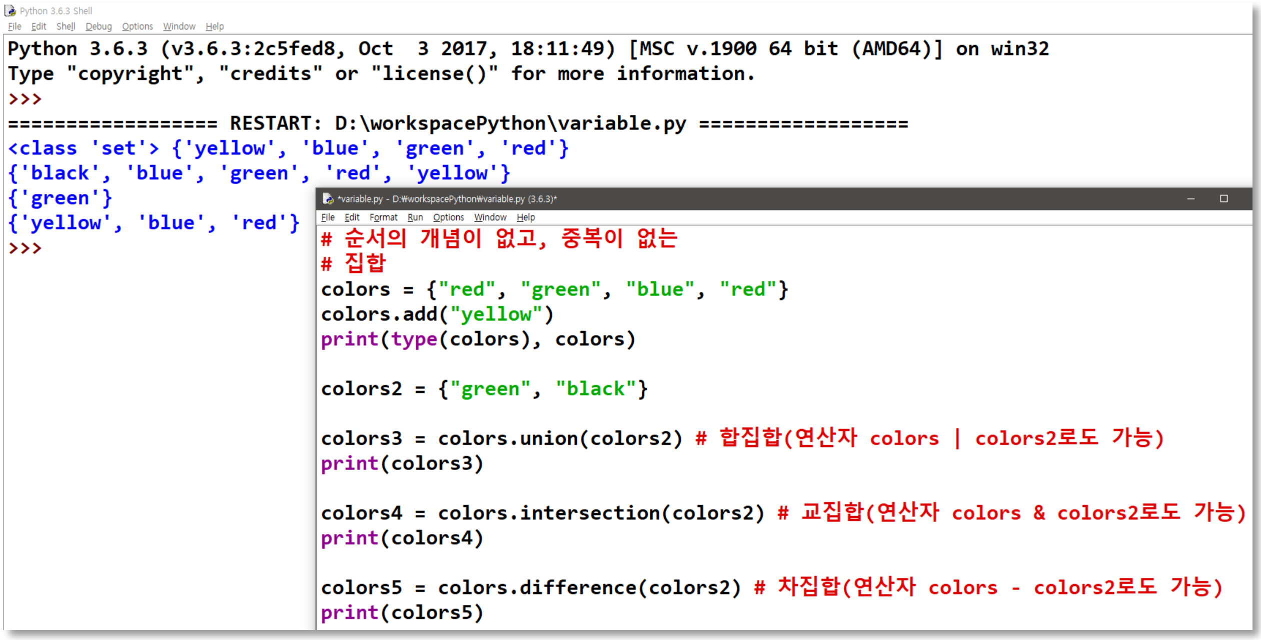
3.1.2 list 제어


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more in
>>> 
================== RESTART: D:\workspacePython\variabl
True
['귤', '배', '사과']
['사과', '배', '귤']
3
['사과', '배', '귤', '딸기', '사과', '복숭아', '바나나']
2
['배', '귤', '사과', '복숭아', '바나나']
바나나
복숭아
['배', '귤', '사과']
>>> 
```

```python
fruits = ["사과", "귤", "배"]

print("사과" in fruits)

fruits.sort() # 오름차순 정렬
print(fruits)
fruits.reverse() # 내림차순 정렬
print(fruits)

print(len(fruits)) # 아이템이 몇개인지

fruits.append("사과") # 맨 뒤에 사과 추가
fruits.insert(3, "딸기") # 세번째 위치에 딸기 추가
fruits += ["복숭아", "바나나"] # 맨 뒤에 여러개 추가
print(fruits)

print(fruits.count("사과")) # 사과가 몇개인지

fruits.remove("사과") # 사과 삭제(같은값이 여러개면 맨 앞에거 삭제)
del fruits[2] # 두번째거 삭제
print(fruits)

print(fruits.pop()) # 맨 마지막거 꺼내와서 사용하고 리스트에서 삭제
print(fruits.pop(3)) # 세번째거 꺼내와서 사용하고 리스트에서 삭제
print(fruits)
```

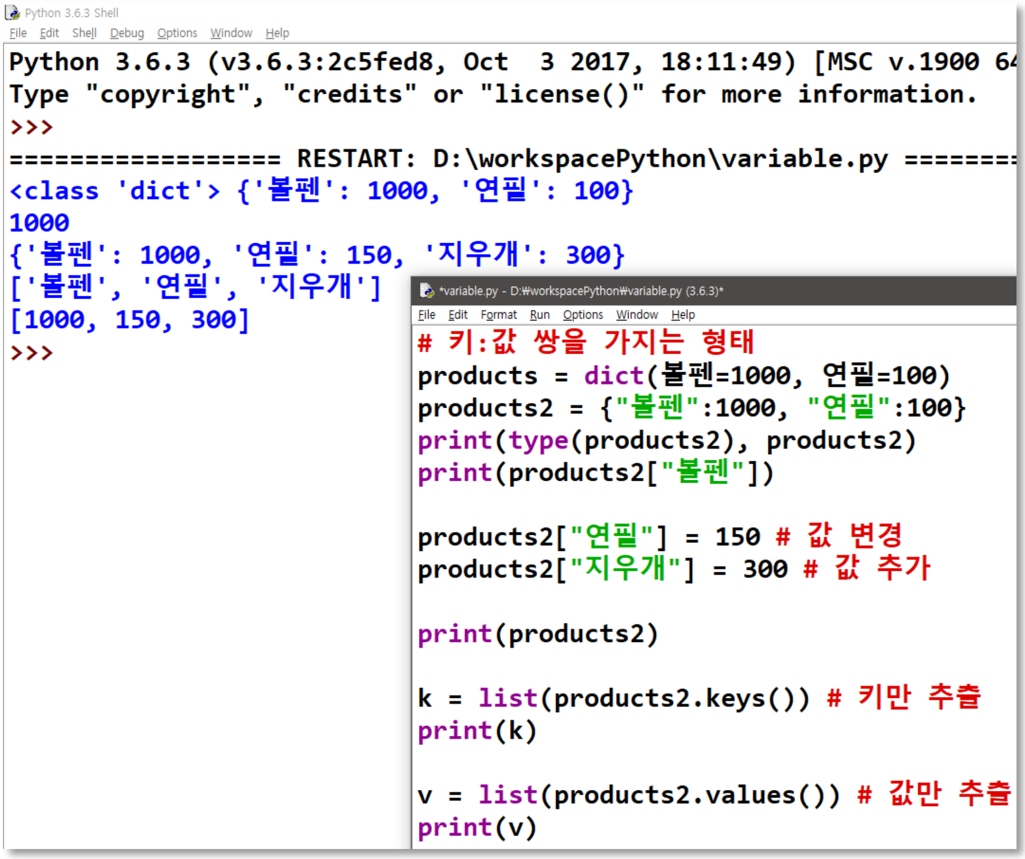
3.2 set


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
================= RESTART: D:\workspacePython\variable.py =================
<class 'set'> {'yellow', 'blue', 'green', 'red'}
{'black', 'blue', 'green', 'red', 'yellow'}
{'green'}
{'yellow', 'blue', 'red'}
>>>
```

```python
# 순서의 개념이 없고, 중복이 없는
# 집합
colors = {"red", "green", "blue", "red"}
colors.add("yellow")
print(type(colors), colors)

colors2 = {"green", "black"}

colors3 = colors.union(colors2) # 합집합(연산자 colors | colors2로도 가능)
print(colors3)

colors4 = colors.intersection(colors2) # 교집합(연산자 colors & colors2로도 가능)
print(colors4)

colors5 = colors.difference(colors2) # 차집합(연산자 colors - colors2로도 가능)
print(colors5)
```

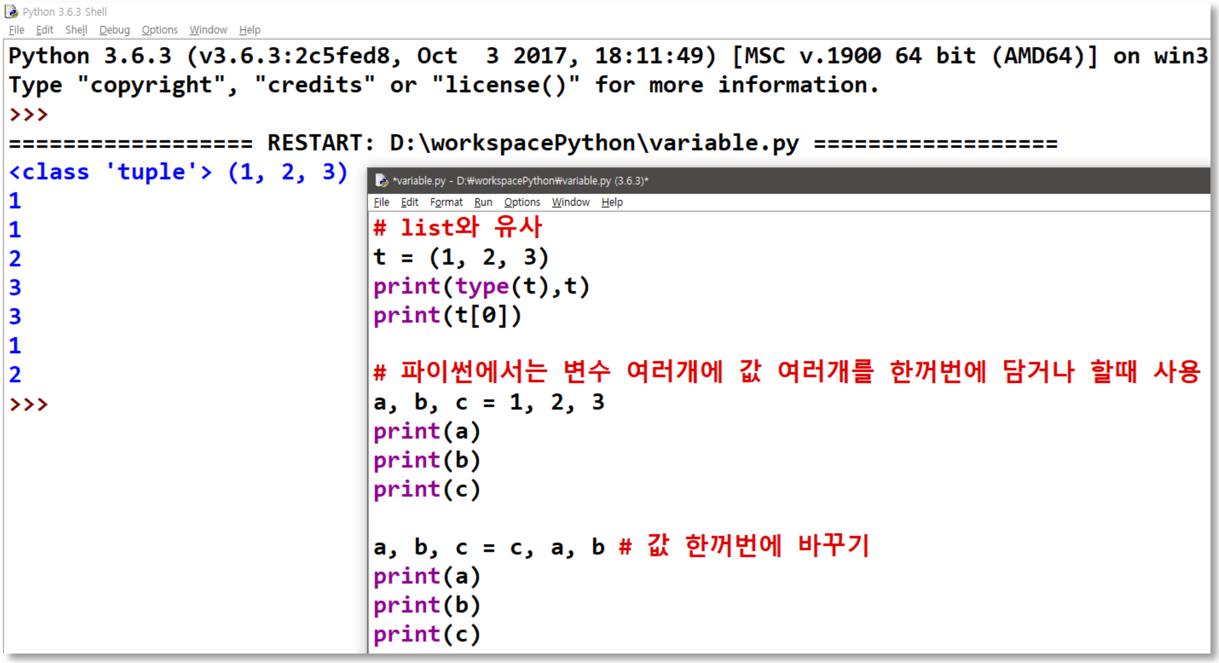
3.3 dict


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64
Type "copyright", "credits" or "license()" for more information.
>>>
================= RESTART: D:\workspacePython\variable.py =======
<class 'dict'> {'볼펜': 1000, '연필': 100}
1000
{'볼펜': 1000, '연필': 150, '지우개': 300}
['볼펜', '연필', '지우개']
[1000, 150, 300]
>>>
```

```python
# 키:값 쌍을 가지는 형태
products = dict(볼펜=1000, 연필=100)
products2 = {"볼펜":1000, "연필":100}
print(type(products2), products2)
print(products2["볼펜"])

products2["연필"] = 150 # 값 변경
products2["지우개"] = 300 # 값 추가

print(products2)

k = list(products2.keys()) # 키만 추출
print(k)

v = list(products2.values()) # 값만 추출
print(v)
```

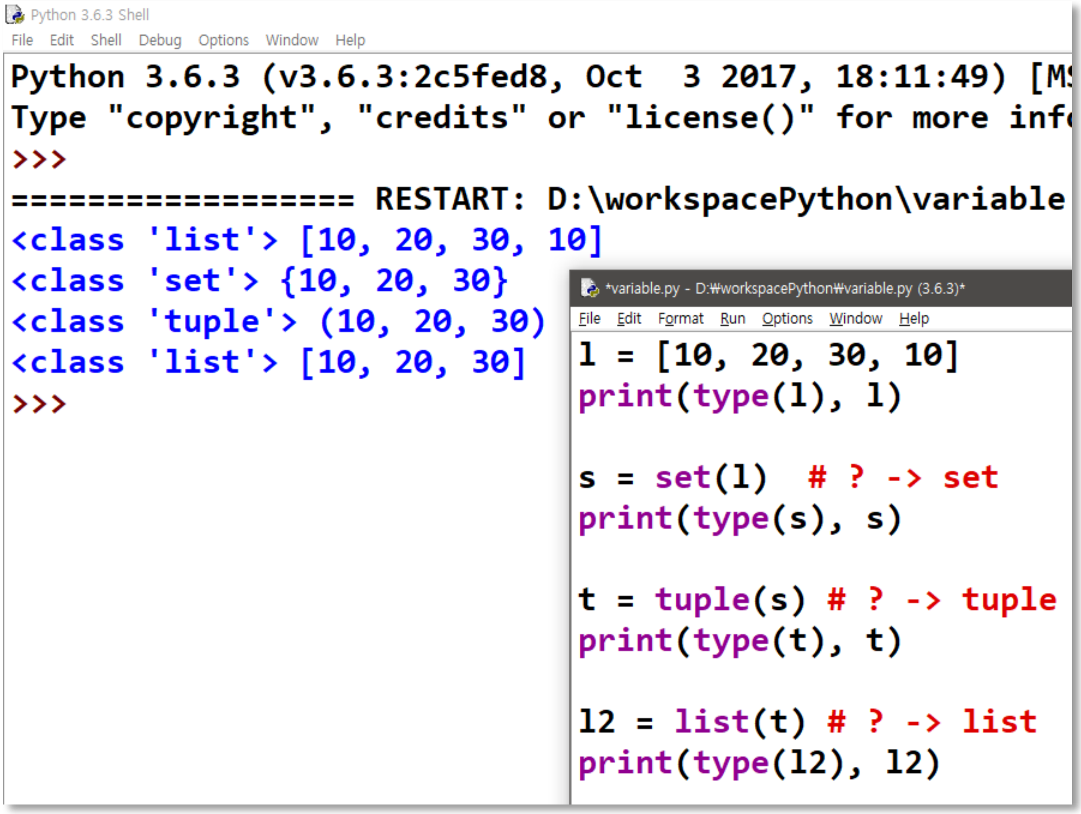
3.4 tuple


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)] on win3
Type "copyright", "credits" or "license()" for more information.
>>>
================= RESTART: D:\workspacePython\variable.py =================
<class 'tuple'> (1, 2, 3)
1
1
2
3
3
1
2
>>>
```

```
# list와 유사
t = (1, 2, 3)
print(type(t),t)
print(t[0])

# 파이썬에서는 변수 여러개에 값 여러개를 한꺼번에 담거나 할때 사용
a, b, c = 1, 2, 3
print(a)
print(b)
print(c)

a, b, c = c, a, b # 값 한꺼번에 바꾸기
print(a)
print(b)
print(c)
```

3.5 컬렉션간의 변환


Python 3.6.3 Shell
File Edit Shell Debug Options Window Help
variable.py - D:₩workspacePython₩variable.py (3.6.3)
File Edit Format Run Options Window Help

```
Python 3.6.3 (v3.6.3:2c5fed8, Oct  3 2017, 18:11:49) [M
Type "copyright", "credits" or "license()" for more inf
>>>
================== RESTART: D:\workspacePython\variable
<class 'list'> [10, 20, 30, 10]
<class 'set'> {10, 20, 30}
<class 'tuple'> (10, 20, 30)
<class 'list'> [10, 20, 30]
>>>
```

```
l = [10, 20, 30, 10]
print(type(l), l)

s = set(l)  # ? -> set
print(type(s), s)

t = tuple(s) # ? -> tuple
print(type(t), t)

l2 = list(t) # ? -> list
print(type(l2), l2)
```