변수 및 자료형

백석대학교 강윤희

변수 사용하기

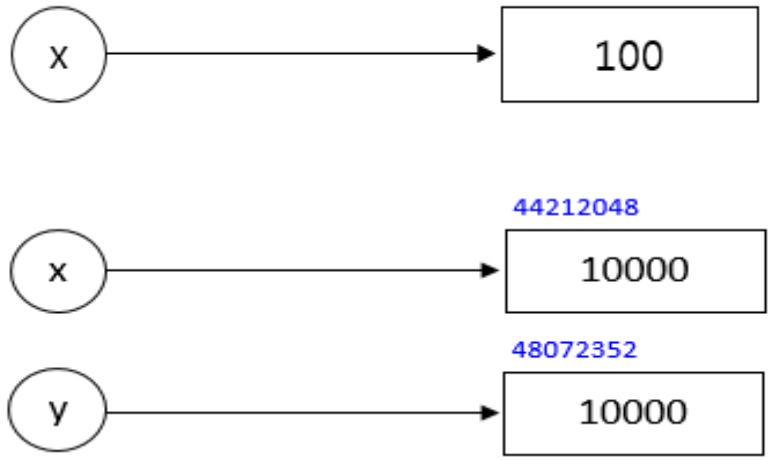
- 변수 명(식별자)는 자료형을 지정하지 않고 값을 변수에 저장할 수 있음 (by 배정문)
- C 언어와 동일하게 식별자로 변수명을 사용함

```
name = "Bob"

age = 15
```

```
age = 15

age += 1 # increment age by 1

print(age)
```

변수 사용하기

```
>>> x=100
>>> id(x)
주소값
>>> type(x)
<class 'int'>

>>> y=100
>>> id(y)
주소값
>>> type(y)
<class 'int'>
```



변수와 객체 바인딩

자료형

- 숫자형(Number)
  - 정수형, 실수형, 복소수
  - 8진수와 16진수
- 문자열(String)
  - 문자, 단어 등으로 구성된 문자들의 집합
- 불(bool)
  - 참(True)과 거짓(False)을 나타내는 자료형
- 리스트(List)
  - 값의 생성, 삭제, 수정이 가능한 가변 배열
- 튜플(Tuple)
  - 값을 변경할 수 없는 리스트
- 딕셔너리(Dictionary)
  - 키를 사용하여 값을 접근하는 연관 배열(Associative array)
- 집합(Set)
  - 중복을 허용하지 않고 순서가 없는 리스트

리스트, 튜플, 딕셔너리, 집합을

컬렉션 자료형으로 구분함

자료형

- 숫자형

```
x = 1    # int
y = 2.8  # float
z = 1j   # complex

print(type(x))
print(type(y))
print(type(z))
```

자료형

- 형변환

```
x = int(1.0)   # x will be 1
y = int(2.8) # y will be 2
z = int("3") # z will be 3
```

실수형으로 자료값 변환을 위해 float() 사용

자료형

- 문자열

```
"Life is too short, You need Python" "a" "123"
```

'hello’ 과 "hello " 동일

print("hello") # 문자열 출력하기

자료형

- 문자열 결합 : 연산자 + 또는 문자열 나열
  - word = 'Help' + 'a'
  - word = 'Help' 'a'
- 문자열 첨자
  - 'Hello'[2] → 'l'
  - 'Hello'[1:3] → 'el'
  - word[-1] → 마지막 문자
  - len(word) → 5

```
In [3]: word = 'Help' 'a'
        word

Out[3]: 'Helpa'
```

인덱싱과 슬라이싱

(문제풀이) 자료형

```
x = "a" "b"
print (x)
print(type(x))
x = 10
print (x)
print(type(x))
```

(문제풀이) 문자열

```
b = "Hello, World!"
print(b[2:5])
```

예상 출력결과는 

자료형

- 문자열

```
a = "    Hello, World!   "    # 공백문자 제거
print(a.strip()) # returns “Hello, World!”

a = "Hello, World! "    #문자열 길이 구하기
print(len(a))
```

자료형

- 불(bool)
  - 참(True)과 거짓(False)을 나타내는 자료형
    - True - 참
    - False – 거짓

  - >>> a = **True**
  - >>> b = **False**
  - >>> type(a) <**class** '**bool**'>
  - >>> type(b) <**class** '**bool**'>

자료형

- List
  - 값의 나열로 여러 자료형의 값을 담을 수 있음
  - [ ] 에 요소를 , 로 분리하여 유지함
  - 인덱싱과 슬라이싱 지원
  - 리스트의 두번째 자료를 출력하고자함

  fruits = ["apple", "banana", "cherry"]
  print( )
  type(fruits) # 출력 내용은

- 복잡한 데이터를 다루기 위해 사용됨

  a = [1, 2, ['a', 'b', ['Life', 'is']]]
  a[2][2][0] 의 값은

자료형

```
>>> lst = ["easy", "simple", "cheap", "free"]
 >>> lst[-1]
'free’
>>> lst = [3, 5, 7]
>>> lst.append(42)
>>> lst
[3, 5, 7, 42]
 >>> lst = [3, 5, 7]
>>> lst = lst.append(42)
>>> print(lst)
None
```

b와 c는 어떤 값을 갖는가 ?

```
>>> a = [1, 2, 3, 4, 5]
>>> b = a[:2]
>> c = a[2:]
```

리스트 a를 정렬하기 위한 함수는 무엇인가 ?