




# Python 기초 문법

## Python Programming

국민대학교 경영대학원 AI빅데이터전공  
문현실 (hsmoon@kookmin.ac.kr)



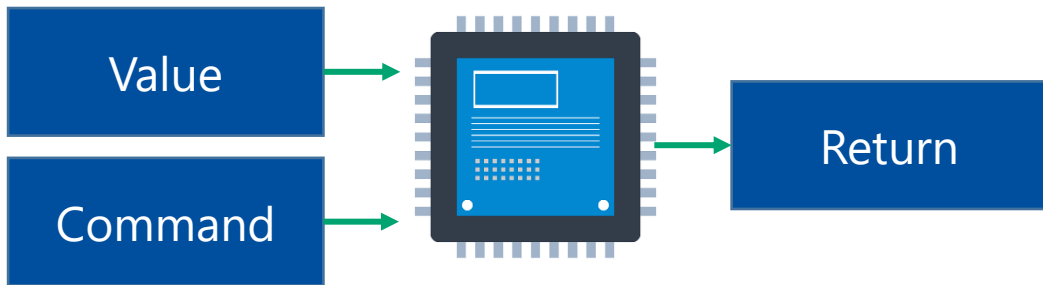


Ch.01

# 값과 변수

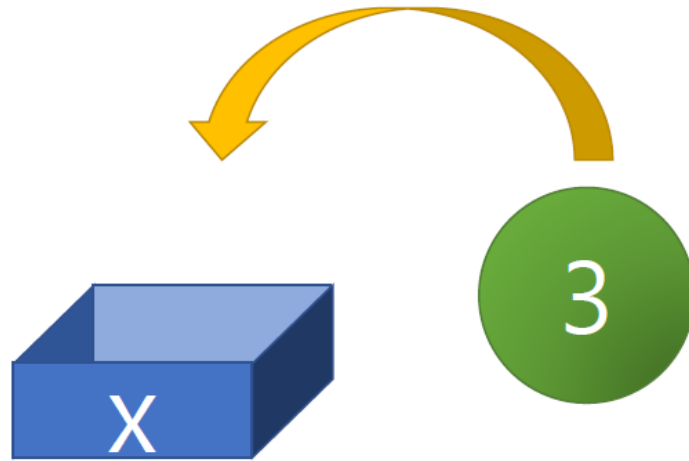
# 값 (Value)

- 프로그래밍에서 계산 처리의 대상
  - 객체 (object) 라고도 함
- 수, 문자, 이미지, 소리 등등



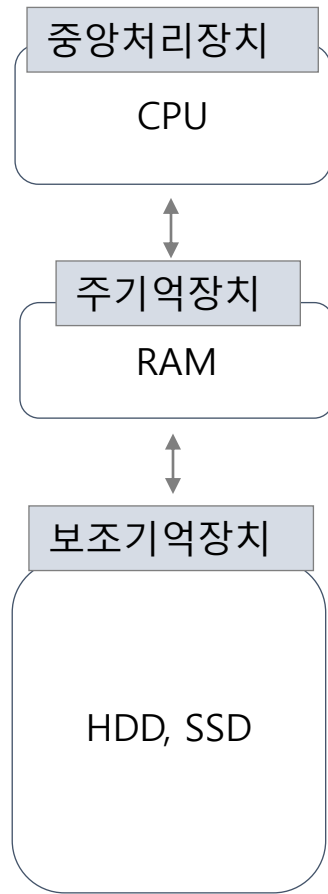
# 변수

- 값을 저장한 위치와 그 이름
- $x=3$ 
  - $x$ 의 이름 3이다?
  - $x$ 는 3이다?
  - $x$ 와 3은 같다?
  - $x$ 에 3을 넣어라?



# 변수와 메모리

- 변수를 선언하여 값을 저장하면 메모리(RAM) 공간 어딘가에 저장
  - 변수는 프로그램에서 특정 값을 저장하는 공간의 이름
  - 변수에 값을 넣으라고 선언하는 순간 물리적으로 메모리 어딘가에 물리적 공간을 확보할 수 있게 메모리 저장 위치를 할당  
-> 메모리 주소
- $x=3$ 
  - $x$ 라고 하는 메모리 공간에,  
 $x$ 라고 하는 메모리 주소에,  
3이라는 값을 넣어라



# 변수의 사용

- 값과 똑같이 쓸 수 있음
  - $3 + 5$
  - $x + 5$
- 변수의 값을 바꿀 수 있음
  - $x = 5$

# 변수명

- 영문자와 숫자, 밑줄문자(\_) 로 작명
  - 알파벳 (a, b, c, ..., A, B, ...), 숫자 (0, 1, 2,...), 한글 (가, 나, ...), 밑줄(\_), 기타 일반 문자 가능 (한글, 일본어, 한자 등)
  - 길이 제한 없음
- 변수명은 의미 있는 단어로 표기
  - 여러 사람과의 의사 소통
  - 변수명은 소문자 추천



# 변수명

- 주의사항
  - 대소 문자를 구분!
  - 변수명 첫 글자에 숫자는 쓸 수 없음
  - 중간에 공백(space, tab)등은 포함하면 안됨
  - Python 문법에서 쓰는 키워드 사용 불가 (if, for, else 등)

# Wrap-up Exercise

- 원의 넓이 구하기
- 화씨온도 변환기 만들기
  - 화씨온도 = (섭씨온도 \* 1.8) + 32

# 자료형(data type)

- 값의 종류
- 대표적인 자료형
  - 논리형 (불린형; bool): True, False
    - 참고 : True = 1, False = 0
  - 정수형 (int): 1, 2, 3, ...
    - 컴퓨터 시스템의 타입(32bit 또는 64bit)에 따라 상이
  - 실수형 (float): 1.4, 5.33
  - 문자열 (str): 'hello', "안녕하세요"
- 그 외에도 다양 : 직접 만들 수도 있음

# 동적 타이핑

- 변수의 메모리 공간을 확보하는 행위가 실행 시점에 발생
  - 변수에 값을 저장하기 위해서는 일정한 메모리 공간의 할당이 필요
    - C나 자바는 `int x=8` 과 같이 자료형의 선언이 필요
    - 파이썬은 할당받는 메모리 공간도 동적으로 다르게 할당 받을 수 있음
- 파이썬은 프로그래머가 아닌 인터프리터가 스스로 자료형을 판단
  - 실행시점에 동적으로 판단

# 자료형 변환

- 자료형의 중요성
  - 자료형별 연산 및 처리의 차이
  - 메모리 용량의 효율적 사용 -> 연산 속도의 향상 및 메모리 관리
    - `str >> float > int > bool`
- 자료형의 확인
  - `type( )`
- 정수형과 실수형 변환
  - `float( )` : 실수형으로 변환
  - `int( )` : 정수형으로 변환
    - 소수점 이하 내림

# Wrap-up Exercise

- 화씨온도 변환기 업그레이드
  - 사용자가 입력받을 수 있도록 : `input()` 사용
  - 화씨온도 = (섭씨온도 \* 1.8) + 32

# 문자열(string)

- 문자로 구성된 데이터
  - String = sequence of characters
- 작은 따옴표 또는 큰 따옴표로 감싸서 표현
  - 반드시 따옴표의 짝이 맞아야 함
- 여러 줄의 문자열
  - \n (개행문자) 사용
  - 작은 따옴표 세개 또는 큰 따옴표 세개 사용

# None

- 없음을 나타내는 Python의 특수값
- 변수 대입은 가능
- 다른 계산은 불가능





## Ch.02 Python으로 연산하기

# 수식(Expression)

- 피연산자들과 연산자의 조합
  - 연산자(Operator) : 어떤 연산을 나타내는 기호 ex)  $+$ ,  $-$ ,  $*$ ,  $/$  등
  - 피연산자(Operand) : 연산의 대상
  - 정확한 수식으로 표현된 어떤 연산을 프로그래밍 언어로 작성하면 컴퓨터는 정확하게 계산
- Wrap-up Exercise : 두 점 사이의 거리 구하기

# 연산 우선 순위

- 여러 연산자가 혼합되어 있으면 우선 순위(precedence)에 따라 연산을 순서대로 수행
  - 소괄호() 를 사용하여 연산 순서 변경 : 추천되는 방식

우선순위	연산명	연산자
1	지수연산	**
2	단항연산자	~, +, -
3	곱셈, 나눗셈, 나머지, 몫	*, /, %, //
4	덧셈, 뺄셈	+, -

# Wrap-up Exercise I

- 초를 00분 00초로 바꾸기
- 갈 때  $10\text{m/s}$ , 올 때  $20\text{m/s}$ 로 주행했을 때, 평균 속력은?
  - 조화평균  $= \frac{2ab}{a+b}$

# 복합 대입 연산자 (Compound Assignment Operator)

- 대입 연산자(=)와 다른 연산자를 합친 것

복합대입연산자	같은 의미
$x += y$	$x = x + y$
$x -= y$	$x = x - y$
$x *= y$	$x = x * y$
$x /= y$	$x = x / y$
$x //= y$	$x = x // y$
$x \% = y$	$x = x \% y$

- 주의사항

- 연산자의 우선 순위에 항상 신경 : 대입 연산자는 가장 마지막

# Wrap-up Exercise II

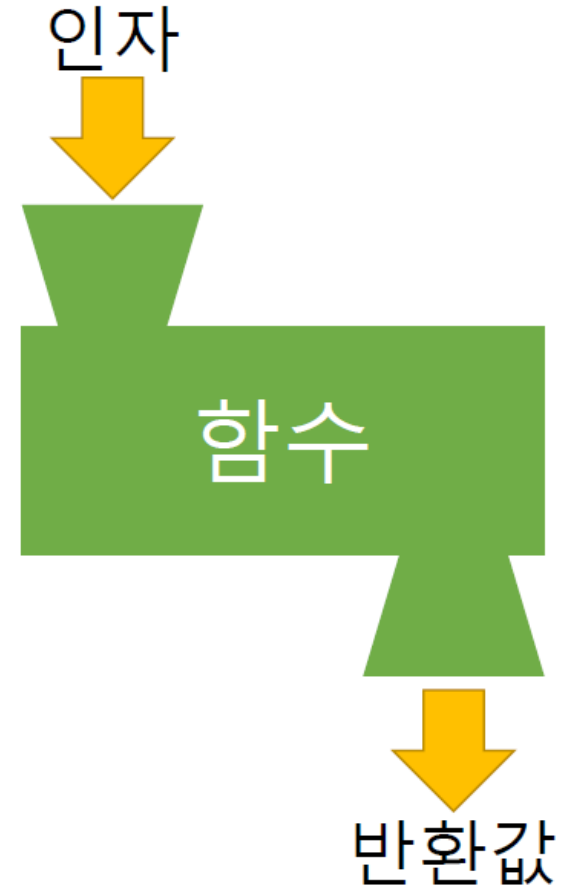
- 거스름돈을 알려주는 프로그램
  - 물건의 가격과 지불한 금액을 입력
  - 물건 가격과 지불한 금액은 100원 단위
  - 거스름돈을 계산해주되 동전의 개수는 최대한 적게
  - 거스름돈은 500원, 100원짜리로만



## Ch.03 함수(function)

# 함수

- 일정한 계산이나 처리를 수행하는 명령
- 수학의 함수와 유사
- 함수명(인자) 형태로 사용
  - 인자: argument
  - 함수의 실행 결과는 반환 (return)





# 함수 예제

- `max()`
  - 더 큰 값을 반환하는 함수
  - `max(1, 3)`
  - `max(1, 3) + 4`
  - `a = max(1, 3)`

# 함수 예제

- `print()` : 화면에 값을 출력하는 함수
  - 컴퓨터에 모니터가 없던 시절에 유래
  - `print(3.14)`
  - `print('hello python')`
  - 반환값이 없음(`None`)




# 함수 예제

- input( ) : 사용자의 입력을 받는 함수
  - 인자 : 입력받을 때 출력할 문자열
  - 반환 : 사용자의 입력값을 문자열(str)로 반환

# Wrap-up Exercise

- 천둥은 어디서 쳤나?
  - 번개를 본 다음에 천둥이 몇 초 뒤에 들렸는지 안다.
  - 소리는 1초에  $340\text{m/s}$  이동
- 정수 더하기
  - 사용자가 1234 를 입력하면  $1+2+3+4 = 10$  출력
- 대략적인 평수 구하기
  - 1의 자리 이하를 버린 뒤 3을 곱하고 버렸던 1의 자리를 3으로 나눈 몫을 더해주기
  - $64.5\text{m}^2 \doteq 6 \times 3 + 1 = 19$



# Q&A

Thank you for your listening

