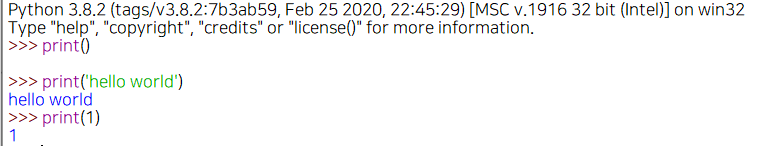
Python

1. Python

파이썬은 다른 언어의 비해 문법이 쉽고 비교적 빠르게 개발이 가능하다는 장점이 있다.

스크린샷이(가) 표시된 사진

자동 생성된 설명2. Input and Print

Interactive shell에서는 자동으로 print를 해주지만 edit에서는 자동으로 print해주지 않는다. 그래서 interactive shell에서만 aaaa가 출력되고 edit에서는 되지 않는다.

print(1, 2, 3)과 같이 콤마 뒤에는 공백을 주는 게 더 좋다.

텍스트이(가) 표시된 사진

자동 생성된 설명

3. Variable and Naming

Variable(변수): 변수에는 값을 할당할 수 있다. 변수에 할당된 값은 변할 수 있다. 변수명을 설정하는데 몇 가지 규칙이 있는데 숫자로 시작할 수 없으며 띄어쓰기가 불가능하다.

Naming(이름): 변수명을 짓기 위한 여러가지 규칙으로 의미 있는 이름을 지어주는 것이다.

4. Data type

자료형에는 숫자형, 문자형, 불린이 있다.

Numeric(숫자형): 숫자와 관련된 자료로 type() 함수를 사용하면 숫자가 실수형(float)인지 정수형(int)인지 알 수 있다.

String(문자형): 문자와 관련된 자료.

Boolean: 참과 거짓을 나타내는 자료.

텍스트이(가) 표시된 사진

자동 생성된 설명

자료형 변환하기

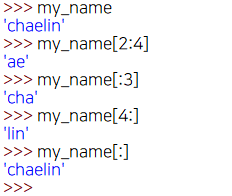
float()를 사용하여 실수형으로 변환하거나 str()를 사용하여 문자형으로 변환하는 것이 가능하다.

5. Comment

Comment(주석): 주석은 컴퓨터를 위한 글이 아니고 사람들끼리 보기 위한 글이다. 보통 코딩에 대한 설명을 첨부할 때 쓴다.

#을 쓰고 난 뒤에 주석을 단다.

6. String

스크린샷이(가) 표시된 사진

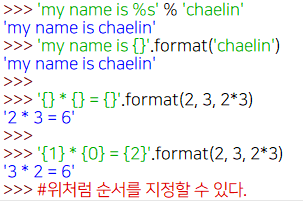
자동 생성된 설명문자들을 나열한 것으로 값을 변경할 수 없고 문자들의 순서를 바꿀 수 없다. “””, ‘’’을 사용할 수도 있다.

Formatting: 문자열을 좀 더 잘 표현하기 위한 것으로 퍼센트 연산자라는 것을 사용한다. 문자열에 숫자나 문자열 등을 대입할 수 있다.

ex) %d, %s, %f

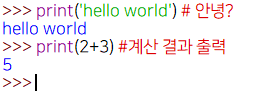
이런 스타일을 c스타일이라고 한다.

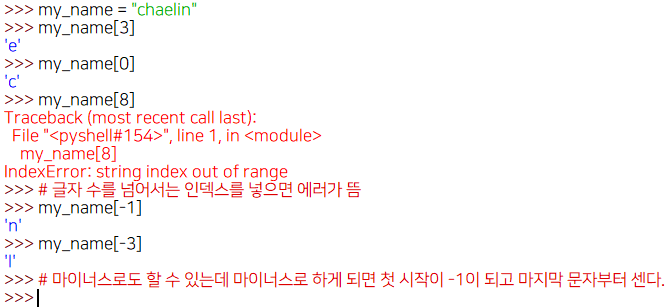
개체이(가) 표시된 사진

자동 생성된 설명Format()

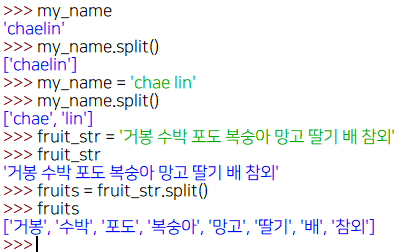
‘{ }’ .format( )

현재 python에서 퍼센트보다 format 메소드를 더 사용하는 추세다.



Indexing: 인덱스는 주소, 위치라고 보면 된다.

Slicing: 한 개가 아닌 여러 개도 뽑아 올 수 있다. slicing 한다고 해서 나머지 문자가 없어지는 것이 아니라 slicing 된 문자를 복사해서 가져오는 것이기 때문에 원래 있던 문자는 그대로 존재한다.

String method: String.split()

Docstring: 문자열을 쓸 때 큰따옴표 세 개씩 써 주석으로 쓸 수 있다. 원래 주석은 #으로 쓸 수 있는데 “””도 주석으로 사용할 수 있다. “””만으로 무언가 실행되지는 않는다. 보통 함수를 설명하면서 주석을 달 때 쓴다.

스크린샷이(가) 표시된 사진

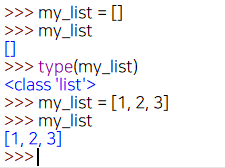
자동 생성된 설명End, Escape code:

Print(‘’, end=’’): 출력의 끝을 지정할 수 있다.

Escape code: 어느 특정한 기능을 수행하는 문자의 조합, 이 코드를 넣으면 특정한 행동을 하게 된다.

예) \n줄바꿈 \t탭

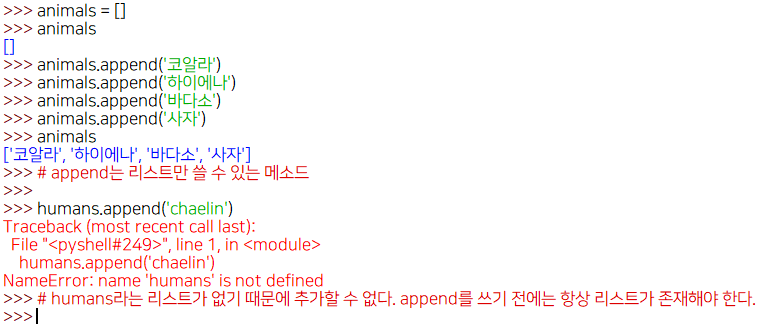
7. List

여러 개의 값을 한 곳에 모아서 저장할 수 있다. list는 숫자형, 문자형, 불린 등을 한꺼번에 모아서 쓸 수 있다. 또한 값이 변화할 수 있다.

[val1, val2, …]

Immutable: 예) 문자열, 값을 바꿀 수 없음

mutable: 예) 리스트, 값을 바꿀 수 있음

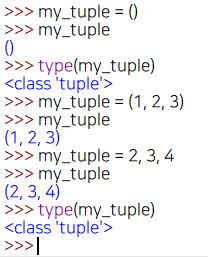
값 추가하기

List method

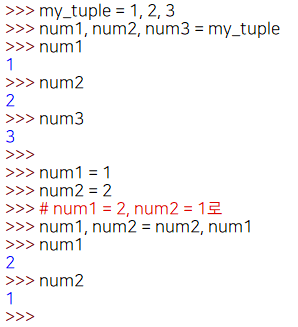
스크린샷이(가) 표시된 사진

자동 생성된 설명list.sort(), list.count()

8. Tuple

list와 비슷하게 여러 값을 모아 저장하는 것으로 다른 점은 그 값을 변경 할 수 없다는 것이다.

(val1, val2, …)



괄호를 쓰지 않고 값만 나열해도 똑같이 작동한다.

Packing: 여러 개의 값을 하나로 묶는 것

Unpacking: 묶여 있는 값을 여러 개로 나누는 것

9. for

스크린샷이(가) 표시된 사진

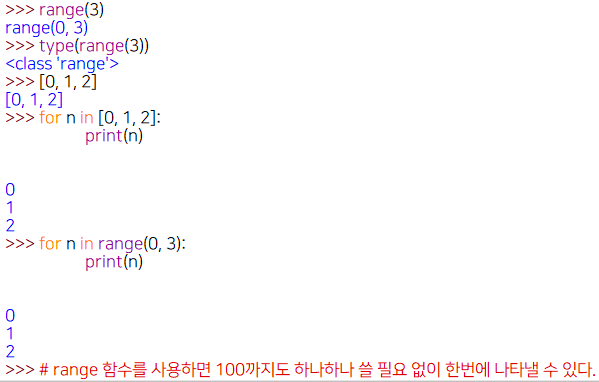
자동 생성된 설명python에서 반복문은 for, while 두가지 방식이 있다.

For 변수 in 컨테이너:

\_\_\_\_(띄어쓰기 혹은 탭) 실행할 명령 1

\_\_\_\_ 실행할 명령 2

반복하는 부분을 code block이라고 한다(콜론 뒷부분). 띄어쓰기는 한번을 써도 되고 두 번을 써도 되지만 처음 한번 띄어쓰기를 했으면 끝까지 한번만 해야 된다. 또한 처음에 탭으로 들여쓰기를 하면 끝까지 탭으로 똑같이 해야 한다. 띄어쓰기 혹은 탭은 필수다. python에서 권장하는 것은 띄어쓰기 4번이다.

range()

스크린샷이(가) 표시된 사진

자동 생성된 설명

for문을 중복해서 사용할 수 있다.

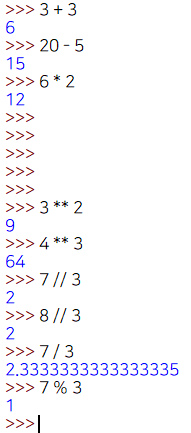
스크린샷이(가) 표시된 사진

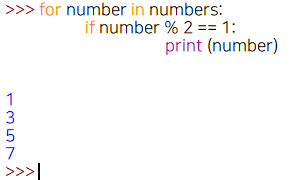
자동 생성된 설명Comprehension: 여러 줄의 반복문을 한 줄로 쓸 수 있다.

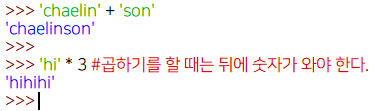
10. Operator

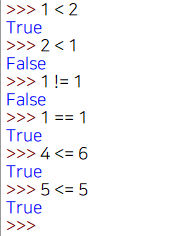
스크린샷이(가) 표시된 사진

자동 생성된 설명Assign(할당): =, +=, -=, \*=, /=(할당 연산자)

Arithmetic(산술연산자): +, -, \*, /, \*\*제곱, //몫, %나머지 (특수 연산자)

%로 홀짝 구분하기

문자열 연산자: +, \*

Comparison(비교): ==(같냐), !=(다르냐), >, <, >=, <=

스크린샷이(가) 표시된 사진

자동 생성된 설명Logical(논리 연산자)

And: 둘 중 하나라도 거짓이면 결과는 거짓

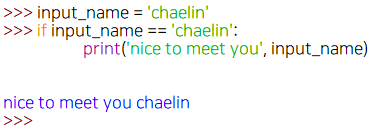
Or: 하나라도 참이면 결과는 참

Not: 참을 거짓으로 거짓을 참으로 바꿈

스크린샷이(가) 표시된 사진

자동 생성된 설명Membership(멤버십 연산자): list 안에 값이 있는지 없는지를 in, not in을 써서 알 수 있다.

11. If

If 조건이 참인지 거짓인지 판단하는 것

If 조건:

\_\_\_\_실행할 명령 1

\_\_\_\_실행할 명령 2

Else, elif

If 조건:

\_\_\_\_실행할 명령 1

\_\_\_\_실행할 명령 2

Else:

\_\_\_\_실행할 명령 1

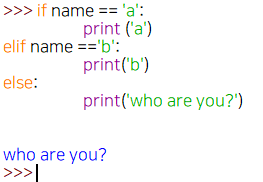
\_\_\_\_실행할 명령 2

If 조건:

\_\_\_\_실행할 명령 1

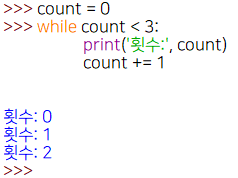
\_\_\_\_실행할 명령 2

elif 조건:

\_\_\_\_실행할 명령 1

\_\_\_\_실행할 명령 2

(if 조건이 f여서 elif 조건으로 넘어간다 elif가 t면 실행 f면 그냥 넘어간다. elif는 여러 개 사용이 가능하다. If, else, elif를 섞어 쓸 수도 있다.)



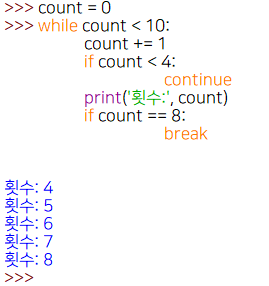
12. While

While 조건:

\_\_\_\_실행할 명령1

\_\_\_\_실행할 명령2

T면 계속 명령을 실행하다가 어느 순간 F면 멈춘다.



Continue, Break

Continue는 밑에 있는 명령을 수행하지 않고 바로 위에 있는 조건으로 넘어간다. Break는 더 이상 반복하지 않고 멈추게 한다.

13. Dictionary

여러 개의 값을 한꺼번에 모아서 다룰 수 있다. 리스트와 다른 점은 관련된 정보를 서로 연관시켜 놓는다는 점이다.

텍스트이(가) 표시된 사진

자동 생성된 설명{key1: val1,…}

Dictionary method

스크린샷이(가) 표시된 사진

자동 생성된 설명dict.values(), dict.keys(), dict.items()

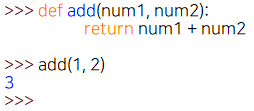
14. Function

함수는 반복되는 코드를 묶어서 이름을 붙여 놓은 것이다.

내장함수: python에서 기본으로 제공하는 함수

모듈의 함수: import를 해서 가져 다가 쓸 수 있는 함수

사용자 정의 함수: 우리가 함수를 직접 만들어서 쓰는 함수

사용자 정의 함수 만드는 법

Def 함수 이름(인자(=매개변수) 1, …):

\_\_\_\_실행할 명령1

\_\_\_\_실행할 명령2

\_\_\_\_return 결과

Why function?

왜 함수가 중요할까? 함수는 만들어 놓으면 다시 사용할 수 있다, 코드를 관리하기 쉬워진다, 조립해서 사용할 수 있다는 큰 장점을 가지고 있기 때문에 많이 사용된다.

스크린샷이(가) 표시된 사진

자동 생성된 설명여러 개 돌려주기

15. Module

python에서 제공하는 모듈, 남들이 만들어 놓은 모듈, 내가 만든 모듈이 있다. 남들이 만들어 놓은 효율적이고 검증된 모듈을 사용할 수 있어 유용하다.

스크린샷이(가) 표시된 사진

자동 생성된 설명Random module

난수를 만들거나 무작위와 관련된 함수를 포함하고 있다.

Random.choice()

Random.sample()

Random.randint()

16. Object

함수와 데이터를 묶어 놓은 것. python에서는 객체가 많이 사용된다.