

파이썬 입문

한국폴리텍대학

2023.03.31

iterable



```
for key in data:  
    print(key)
```

```
for key in data.keys():  
    print(key)
```

```
for value in data.values():  
    print(value)
```

```
for key, value in data.items():  
    print(f'{key}, {value}')
```

형변환 (Casting)

```
list(tuple(list(data)))  
int(str(int(float(data))))  
set(data)  
tuple(data)  
data = dict(list[3][0])  
list(tuple[2][3])  
str(data)  
int(data)  
float(3)    # 3.0  
int(3.0)    # 3  
str(3)      # '3'  
hex(12)     # '0xa'  
oct(10)     # '0o12'  
bin(10)     # '0b1010'
```

단위 계단 함수 (Unit Step Function)

활성화 함수(activation function)


입력이 0보다 크면 1, 같거나 작으면 0

```
def unit_step_func (x):  
    if x >= 0:  
        return 1  
    else:  
        return 0
```

```
y = unit_step_func(2.5)  
print(y)
```


내포(內包) Comprehension (list, dict, set)

list comprehension



```
data1 = [i for i in range(10)]    --> [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
data2 = list(i for i in range(10)) --> [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```



```
data3 = [i for i in range(20) if i % 3 == 0] --> [0, 3, 6, 9, 12, 15, 18]
```



```
data4 = [i * j for j in range(5) for i in range(2)] --> [0, 0, 0, 1, 0, 2, 0, 3, 0, 4]
```

내포(內包) Comprehension (list, dict, set, tuple)

```
data1 = [i * j for j in range(2, 10) for i in range(1, 10)]    # 구구단
```

```
data1 = [i * j for j in range(2, 10)  
          for i in range(1, 10)]
```

```
data2 = [i for i in range(20) if i % 2 == 0 if i % 3 == 0]    # [0, 6, 12, 18]  
        if i % 2 == 0 and if i % 3 == 0                      # SyntaxError  
        if i % 2 == 0 or if i % 3 == 0                       # SyntaxError  
        if i % 2 == 0 or i % 3 == 0                          # [0, 2, 3, 4, 6 ... ]  
        if i % 2 == 0 and i % 3 == 0                         # [0, 6, 12, 18]
```

내포(內包) Comprehension (list, dict, set, tuple)

```
data1 = [[ i + j for i in range(2) ] for j in range(5)]          # 2차원 list
```

```
data2 = {str(i) : i for i in range(5)}      # dictionary comprehension
```

```
data3 = {i for i in range(5)}              # set comprehension
```

```
data4 = tuple(i for i in range(5))         # tuple comprehension
```

```
data4 = (i for i in range(5))              # generator expression
```

```
    print(next(data4, 100)) # 0
```

```
    print(next(data4, 100)) # 1
```

```
    print(next(data4, 100)) # 2
```

```
    print(type(data4)) # class 'generator'
```

클래스 super()

```
class person1:
    def __init__(self, name, job):
        self.name = name
        self.age = age
```

```
class person2:
    def __init__(self, addr):
        self.addr = addr
```

```
class student(person1, person2):
    def __init__(self, name, job, gender):
        super().__init__(name, job)
        #super(student, self).__init__(name, job)
        #super(person2, self).__init__(name, job)
        #person2.__init__(self, addr)
        #person1.__init__(self, name, job)
        self.gender = gender
```

```
# person1    <-- (python 3 가능)
              <-- (python 2, 3 모두 가능)
```


list

```
data = []
```

```
data.append(10)  
data.append(20)  
data.append(30)  
data.clear()
```

```
data[0] = 10    # error  
data[1] = 20    # error  
data[2] = 30    # error
```

```
data = [0, 0, 0]
```

```
data.append(10) # [0, 0, 0, 10]  
data.append(20) # [0, 0, 0, 10, 20]  
data.append(30) # [0, 0, 0, 10, 20, 30]  
data.clear()    # []
```

```
data[0] = 10    # [10, 0, 0]  
data[1] = 20    # [10, 20, 0]  
data[2] = 30    # [10, 20, 30]
```

2차원 합성곱 계산 (convolution)

1	2	3	0
0	1	2	3
3	0	1	2
2	3	0	1

입력 데이터



2	0	1
0	1	2
1	0	2

필터



15	16
6	15

사용법

pip install 패키지명

1. 파이썬 패키지를 설치하고 관리하는 패키지 매니저 (package manager)
2. python 3.4 (버전) 이후부터는 내장되어 있음. 따로 설치할 필요 없음
3. pip : 파이썬 2 (버전) / pip3 : 파이썬 3 (버전) 이상 / pip3.xx : 파이썬 3.xx (버전)

사용법

pip install matplotlib

어디에 설치되었는지 확인할 수 있음.
현재 사용중인 파이썬 위치와 같아야 함

```
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (2.4.7)
Requirement already satisfied: cycler>=0.10 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (4.38.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: packaging>=20.0 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (20.9)
Requirement already satisfied: pillow>=6.2.0 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (8.0.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (1.2.0)
Requirement already satisfied: numpy>=1.17 in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from matplotlib) (1.21.6)
Requirement already satisfied: six in c:\users\remy\appdata\local\programs\python\python37\lib\site-packages (from cycler>=0.10->matplotlib) (1.16.0)
WARNING: Ignoring invalid distribution -ensorflow-gpu (c:\users\remy\appdata\local\programs\python\python37\lib\site-packages)
Installing collected packages: matplotlib
WARNING: Ignoring invalid distribution -ensorflow-gpu (c:\users\remy\appdata\local\programs\python\python37\lib\site-packages)
Successfully installed matplotlib-3.5.3
WARNING: Ignoring invalid distribution -ensorflow-gpu (c:\users\remy\appdata\local\programs\python\python37\lib\site-packages)
WARNING: Ignoring invalid distribution -ensorflow-gpu (c:\users\remy\appdata\local\programs\python\python37\lib\site-packages)
WARNING: Ignoring invalid distribution -ensorflow-gpu (c:\users\remy\appdata\local\programs\python\python37\lib\site-packages)
PS C:\Users\remy\Desktop\workspace>
```