제4강 Lists

학습 목차

- 리스트 정의
- 리스트 멤버 액세스
- 슬라이스
- 다중 배열
- 멤버 삭제와 삽입
- 리스트 iteration
- 멤버 판별
- 기타 기능
- 튜플

리스트 (List)

- 순서가 있는 값들의 집합
- 파이썬의 내장 자료구조

```
[1,2,3]

['cat', 'bat', 'rat', 'elephant']

['hello', 3.1415, True, None, 42]
```

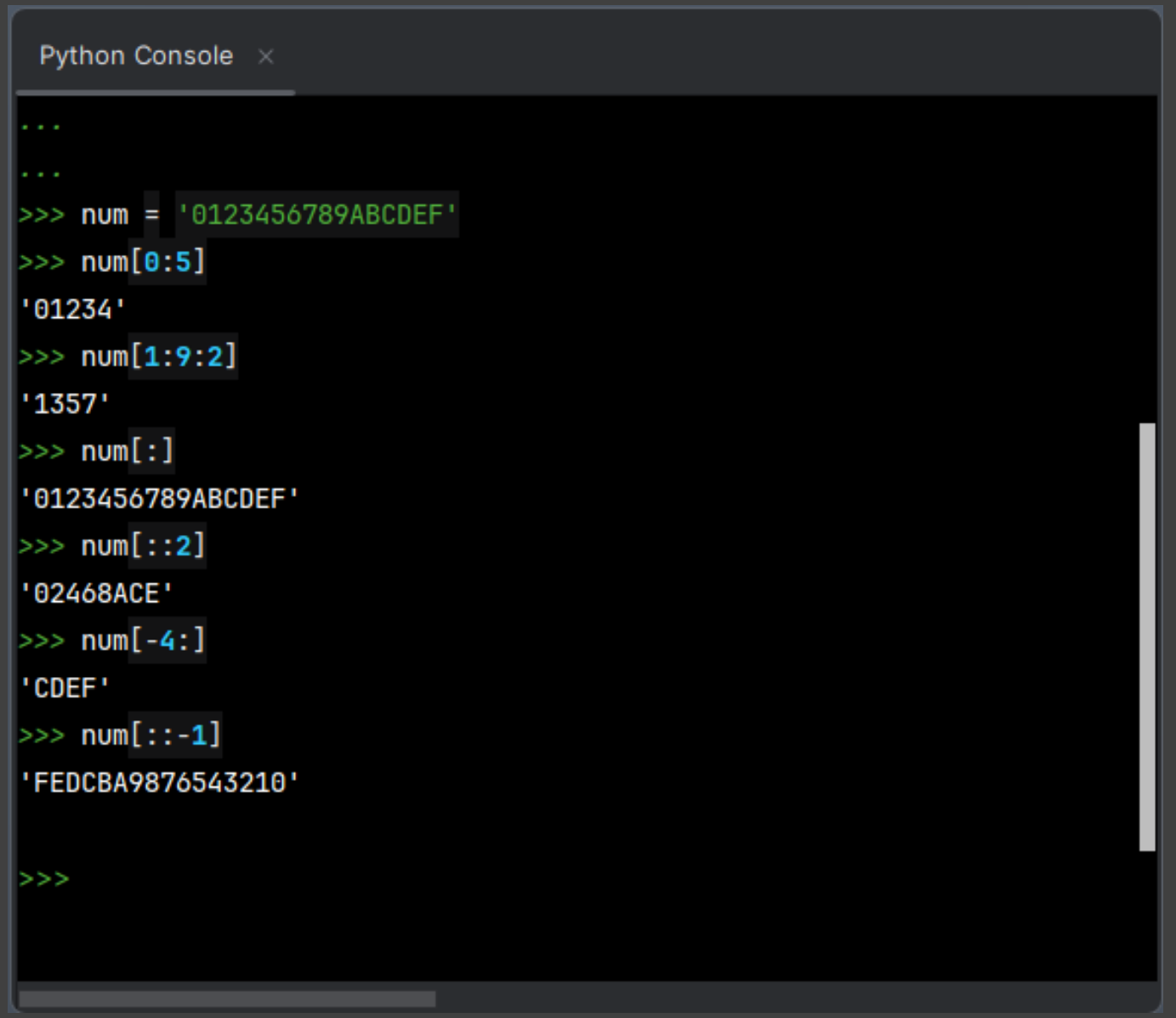
리스트 멤버 액세스

Python Console

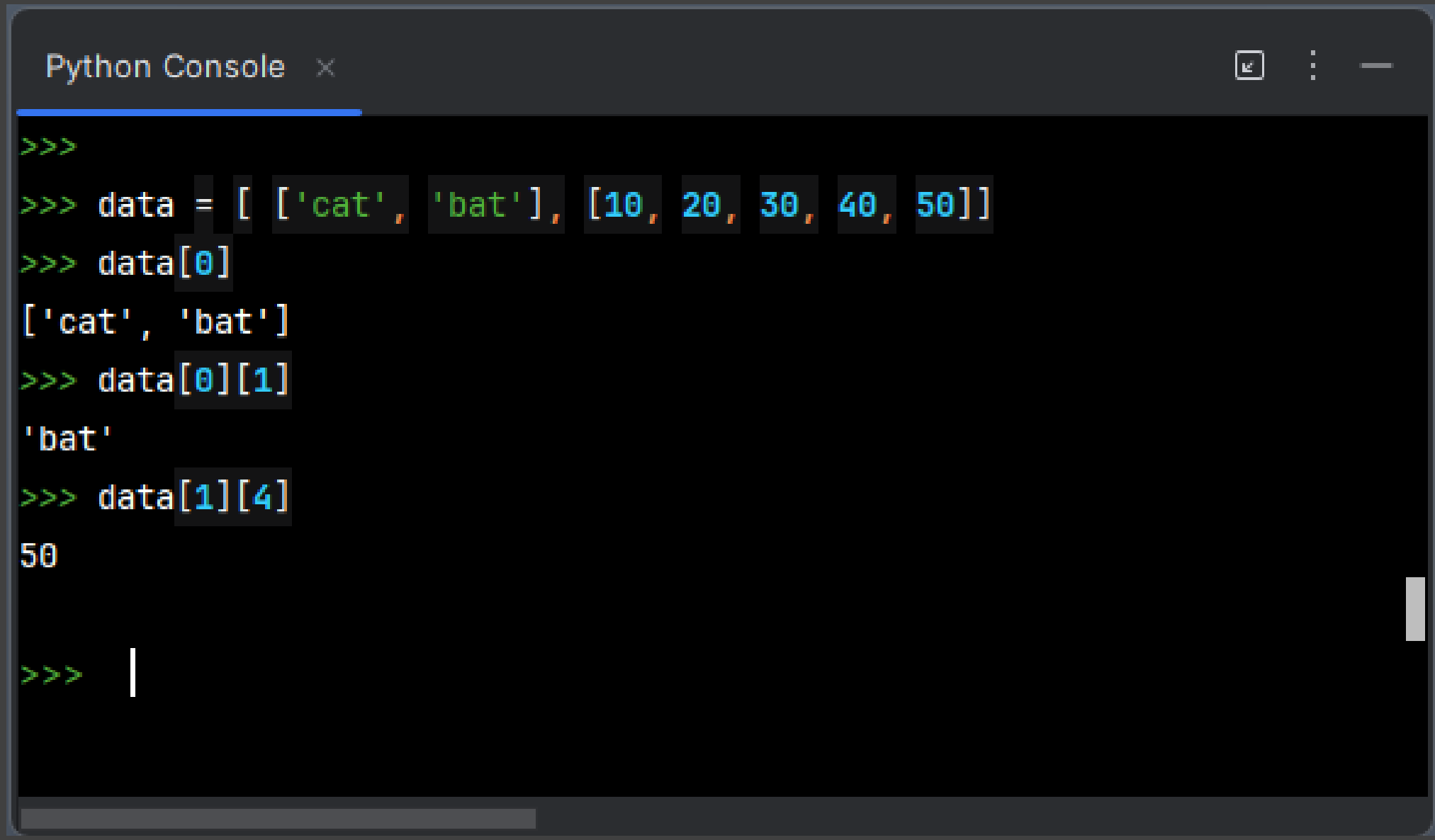
```
>>> black_pink = ['jisu', 'jeni', 'rose', 'risa']
>>> black_pink[0]
'jisu'
>>> black_pink[3]
'risa'
>>> black_pink[4]
Traceback (most recent call last):
  File "C:\Program Files\JetBrains\PyCharm Community Edition 2023.2.4\pl
    coro = func()
           ^^^^^^
  File "<input>", line 1, in <module>
IndexError: list index out of range
>>> black_pink[-1]
'risa'
>>> black_pink[-2]
'rose'


>>>
```

슬라이스 – 규칙에 따른 여러 개의 멤버들을 추출

[start : stop : step]



다중 배열


Python Console
>>>
>>> data = [['cat', 'bat'], [10, 20, 30, 40, 50]]
>>> data[0]
['cat', 'bat']
>>> data[0][1]
'bat'
>>> data[1][4]
50
>>>

변경, 삽입, 삭제

```
>>> black_pink = ['jisu', 'jeni', 'rose', 'risa']
>>> len(black_pink)
4
>>> black_pink[0] = 'daehyun'
>>> black_pink
['daehyun', 'jeni', 'rose', 'risa']
>>> black_pink.append('JYP')
>>> black_pink
['daehyun', 'jeni', 'rose', 'risa', 'JYP']
>>> black_pink = black_pink + ['YG']
>>> black_pink
['daehyun', 'jeni', 'rose', 'risa', 'JYP', 'YG']
>>> twice = ['momo', 'sana', 'zwi', 'nayun', 'dahyun']
>>> unite = black_pink + twice
>>> unite
['daehyun', 'jeni', 'rose', 'risa', 'JYP', 'YG', 'momo', 'sana', 'zwi', 'nayun', 'dahyun']
>>> unite.remove('daehyun')
>>> unite
['jeni', 'rose', 'risa', 'JYP', 'YG', 'momo', 'sana', 'zwi', 'nayun', 'dahyun']
>>> del unite[0]
>>> unite
['rose', 'risa', 'JYP', 'YG', 'momo', 'sana', 'zwi', 'nayun', 'dahyun']
>>> unite.insert(2, 'IU')
>>> unite
['rose', 'risa', 'IU', 'JYP', 'YG', 'momo', 'sana', 'zwi', 'nayun', 'dahyun']
```

리스트 멤버 iteration

```
black_pink = ['jisu', 'jeni', 'rose', 'risa’]

for member in black_pink:
    print(member)

for i in range(len(black_pink)):
    print(f'{i+1}th member is {black_pink[i]}’)

for i, member in enumerate(black_pink):
    print(f'{i+1}th member is {member}')
```

리스트 멤버 여부 판별

Python Console

```
>>>
>>> 'jisu' in black_pink
True
>>> 'rose' not in black_pink
False
>>> 'daehyun' not in black_pink
True


>>>
```

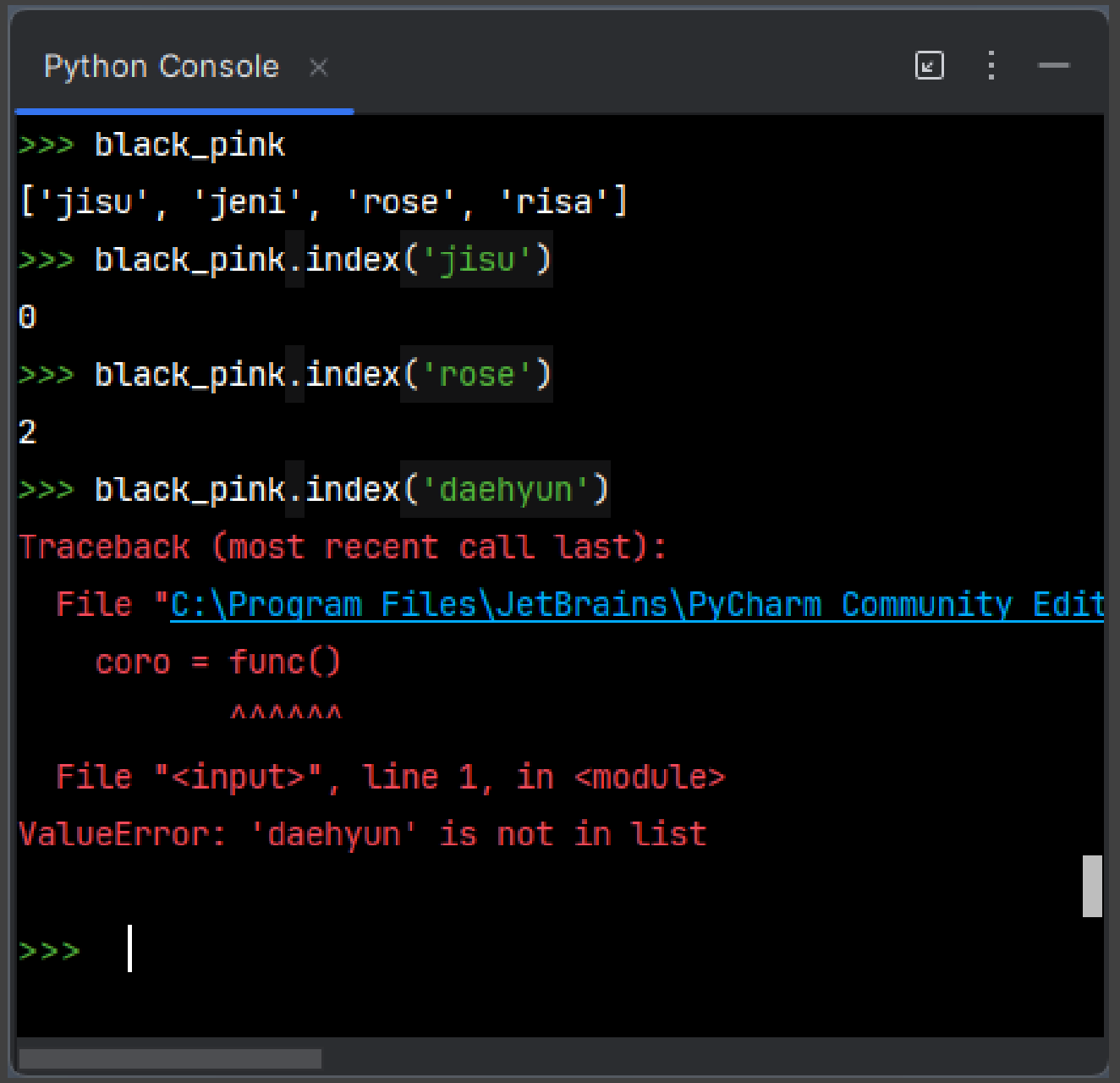
리스트를 활용한 다중 대입

```
cat = ['fat', 'black', 'loud’]

size, color, character = cat
```

Augmented Assignment Operators

Augmented assignment statement	Equivalent assignment statement
`spam = spam + 1`	`spam += 1`
`spam = spam - 1`	`spam -= 1`
`spam = spam * 1`	`spam *= 1`
`spam = spam / 1`	`spam /= 1`
`spam = spam % 1`	`spam %= 1`

index


Python Console
>>> black_pink
['jisu', 'jeni', 'rose', 'risa']
>>> black_pink.index('jisu')
0
>>> black_pink.index('rose')
2
>>> black_pink.index('daehyun')
Traceback (most recent call last):
File "C:\Program Files\JetBrains\PyCharm Community Edit
coro = func()
^^^^^^
File "<input>", line 1, in <module>
ValueError: 'daehyun' is not in list
>>>

sort

Python Console

```
>>>
>>> data = [2, 5, 3.14, 1, -7]
>>> data.sort()
>>> data
[-7, 1, 2, 3.14, 5]
>>> data = ['ants', 'cats', 'dogs', 'elephants']
>>> data.sort()
>>> data
['ants', 'cats', 'dogs', 'elephants']
>>> data.sort(reverse=True)
>>> data = [1, 3, 2, 4, 'cat']
>>> data.sort()
Traceback (most recent call last):
  File "C:\Program Files\JetBrains\PyCharm Community Edition 2023.2.4
    coro = func()
           ^^^^^^
  File "<input>", line 1, in <module>
TypeError: '<' not supported between instances of 'str' and 'int'
```

사용자 정의 크기 비교

Python Console

```
>>> data = ['top', 'four', 'start', 'a']
>>> data.sort(key=lambda x:len(x))
>>> data
['a', 'top', 'four', 'start']
>>> data.sort(key=lambda x:len(x), reverse=True)
>>> data
['start', 'four', 'top', 'a']

>>>
```

퀴즈

Python Console

```
>>>
>>> data = [1, 3, 2, 4, 'cat']
>>>
>>> data.sort(                                                    )
>>>
>>> data
['cat', 1, 2, 3, 4]
>>>
```

copy()

```
>>> spam = ['A', 'B', 'C', 'D']
>>> cheese = spam
>>> cheese[1] = 77
>>> spam
['A', 77, 'C', 'D']
>>> cheese
['A', 77, 'C', 'D']
```

```
>>> import copy
>>> spam = ['A', 'B', 'C', 'D']
>>> cheese = copy.copy(spam)
>>> cheese[1] = 77
>>> spam
['A', 'B', 'C', 'D']
>>> cheese
['A', 77, 'C', 'D']
```

deepcopy()

Python Console

```
>>> data1 = [ ['jisu', 'rose'], ['23', '24']]
>>> data2 = data1.copy()
>>> data2[0][1] = 'jeni'
>>> data1
[['jisu', 'jeni'], ['23', '24']]
>>> id(data1[0])
2344738215104
>>> id(data2[0])
2344738215104

>>>
```

Python Console

```
>>> data1 = [ ['jisu', 'rose'], ['23', '24']]
>>> data2 = copy.deepcopy(data1)
>>> data2[0][1] = 'jeni'
>>> data1
[['jisu', 'rose'], ['23', '24']]
>>> id(data1[0])
2344738214400
>>> id(data2[0])
2344738213888

>>>
```

List Comprehension

- 리스트 데이타값들을 규칙에 의해서 자동 filling하는 것.

```
>>> values = [i for i in range(10)]
>>> values
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
>>> odd_values = [n for n in range(20) if n % 2 == 1]
>>> odd_values
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19]
>>> [(x, y) for x in [1, 2, 3] for y in ['a', 'b', 'c']]
[(1, 'a'), (1, 'b'), (1, 'c'), (2, 'a'), (2, 'b'), (2, 'c'), (3, 'a'), (3, 'b'), (3, 'c')]
>>> vec = [[1,2,3], [4,5,6], [7,8,9]]
>>> [num for elem in vec for num in elem]
[1, 2, 3, 4, 5, 6, 7, 8, 9]
>>> import random
>>> values = [random.randint(0, 5) for i in range(20)]
>>> values
[4, 2, 5, 4, 5, 5, 1, 5, 5, 0, 1, 4, 2, 0, 3, 4, 3, 1, 1, 1]
>>> [v if v != 0 else 'ZERO' for v in values]
[4, 2, 5, 4, 5, 5, 1, 5, 5, 'ZERO', 1, 4, 2, 'ZERO', 3, 4, 3, 1, 1, 1]
```

퀴즈

Python Console

```
>>>
>>> data = [1, 3, 2, 4, 'cat']
>>>
>>> data.sort(key=lambda x: x if type(x) == int else -99999999999999999999999999999)
>>>
>>> data
['cat', 1, 2, 3, 4]
>>>
```