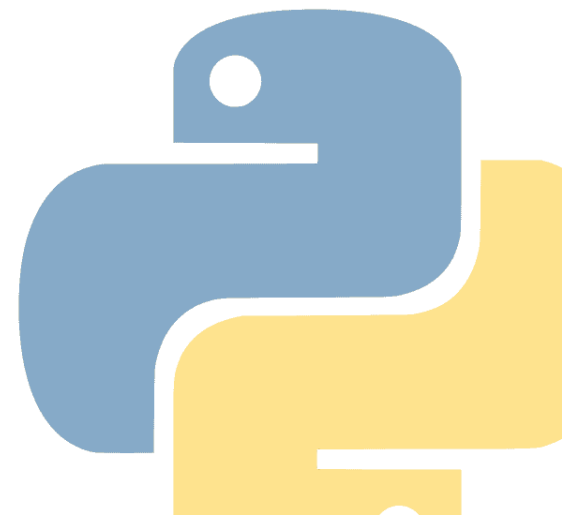


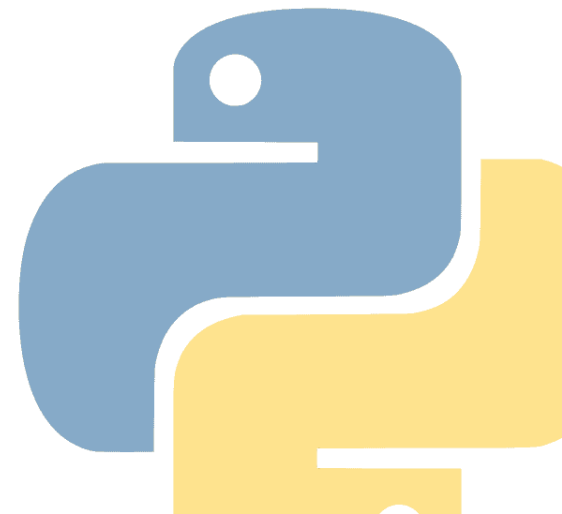
# 웹/파이썬 멘토링

<List & Loop Statement>

강유미 김진환



# List



## 리스트란?

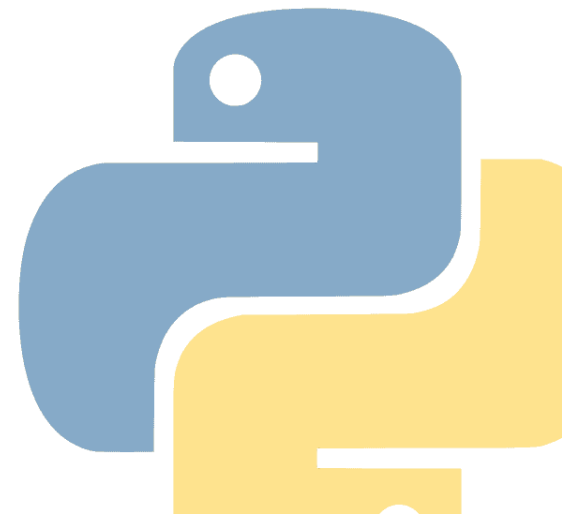
```
list_name = [0, 1, 2, 3, 4, 5, 6]
```

```
a = ['Hi', 'TGWinG', 'everyone']
```

```
b = [5, 7, 'Hi', 'TGwinG']
```

```
c = [5, 7, ['Hi', 'TGwinG']]
```

각 요소는 순서를 가진다 → index 사용



## 리스트 인덱싱

```
a = [1, 2, 3, 4]
```

```
print(a[0], a[1], a[2], a[3])
```

```
print(a[4])
```

```
print(a[0] + a[1])
```

```
print(a[-1])
```

3

4

1 2 3 4

```
Traceback (most recent call last):  
  File "main.py", line 3, in <module>  
    print(a[4])  
IndexError: list index out of range
```



## 리스트 슬라이싱

```
a = [1, 2, 3, 4]
```

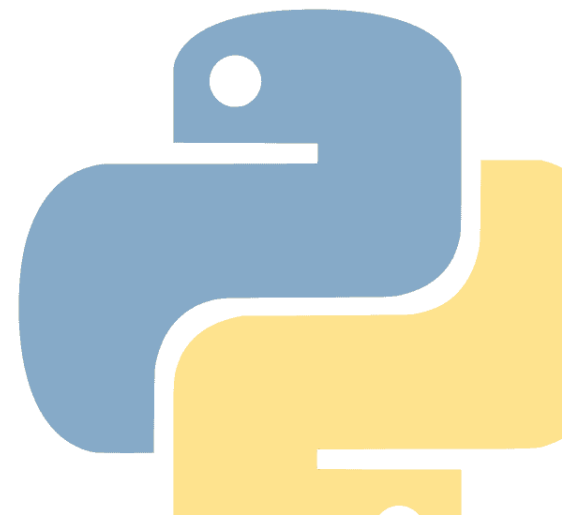
```
print(a[0:2])
```

```
[1, 2]
```

```
print(a[:2])
```

```
print(a[2:])
```

```
[3, 4]
```





# Quiz

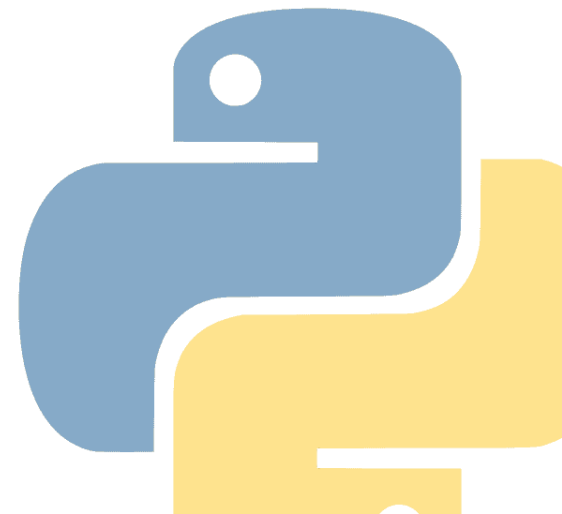
```
a = [1, 2, 3, 4]

print(a[2] + 'hi')
```

```
print(a[1:-1])
```

```
Traceback (most recent call last):
  File "main.py", line 3, in <module>
    print(a[2] + 'hi')
TypeError: unsupported operand type(s) f
or +=: 'int' and 'str'
```

```
[2, 3]
```



## 리스트 요소

파이썬에서 사용되는 모든 자료형 → 객체

```
list = [1, 3.14, 'hi', True, ['TGwing', 'everyone', 2], (3, 4),  
        {'coffee': 'Starbucks', 'dessert': 'Twosome'}, {'가위', '바위', '보'}  
        ]  
print(list)
```

```
[1, 3.14, 'hi', True, ['TGwing', 'everyone', 2], (3, 4), {'coffee': 'Starbucks', 'dessert': 'Twosome'}, {'가위', '보', '바위'}]
```



## 리스트 요소

```
a = [1, 2, 3, 4]
```

수정

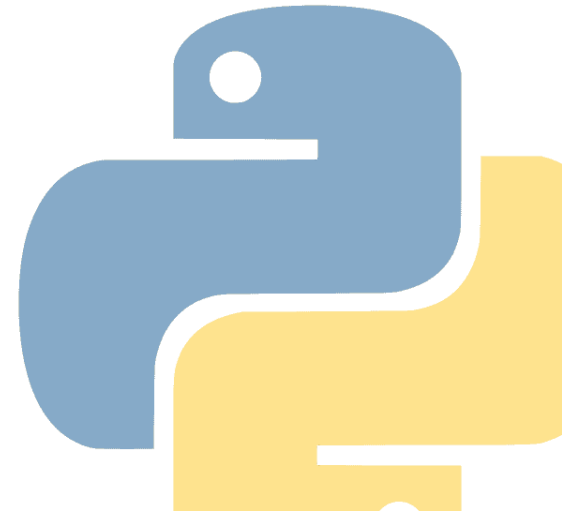
```
a[3] = 0  
print(a)
```

```
[1, 2, 3, 0]
```

삭제

```
del a[1]  
print(a)
```

```
[1, 3, 4]
```





## 리스트 메소드

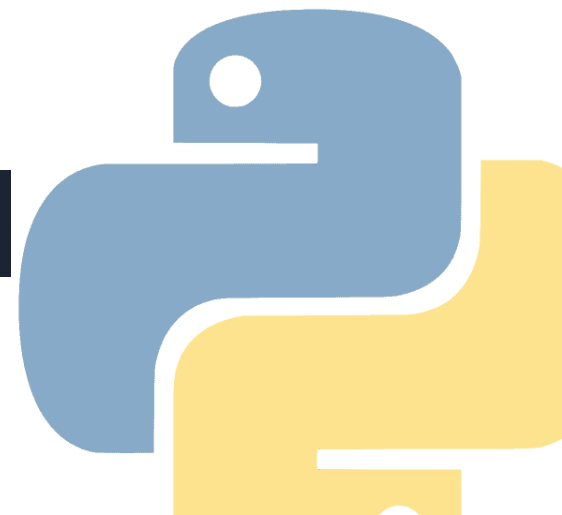
파이썬에서 제공해주는 객체의 함수

**append(값) : 리스트에 요소 추가**

```
a = [1, 2, 3, 4]

a.append(5)
print(a)
```

```
[1, 2, 3, 4, 5]
```



## 리스트 메소드

**insert(위치, 값) : 해당 위치에 요소 삽입**

```
a = [1, 2, 3]
a.insert(0,4)
```

```
[4, 1, 2, 3]
```

**remove(값) : 리스트내 첫번째로 나오는 해당 값 제거**

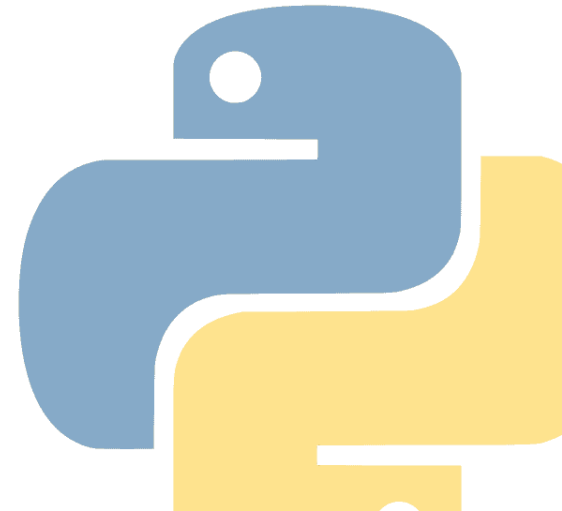
```
a = [1,2,3,1,2,3]
a.remove(3)
```

```
[1, 2, 1, 2, 3]
```

**pop(위치) : 해당 위치의 요소 제거(default = -1)**

```
a = [1,2,3]
a.pop()
```

```
[1, 2]
```



## 리스트 메소드

len(리스트), index(값), count(값), reverse(), sort() ...

```
>>> a = [1, 2, 3]
>>> len(a)
3
>>> a.index(3)
2
```

```
>>> a = [1, 2, 3, 1]
>>> a.count(1)
2
```

```
>>> a = ['a', 'c', 'b']
>>> a.reverse()
>>> a
['b', 'c', 'a']
```

```
>>> a = [1, 4, 3, 2]
>>> a.sort()
>>> a
[1, 2, 3, 4]
>>> a.sort(reverse = 1)
>>> a
[4, 3, 2, 1]
```

<b>max</b> (리스트)	리스트에 있는 데이터 중 가장 큰 데이터를 반환합니다.
<b>min</b> (리스트)	리스트에 있는 데이터 중 가장 작은 데이터를 반환합니다.
<b>sum</b> (리스트)	리스트에 있는 데이터의 합을 반환합니다.

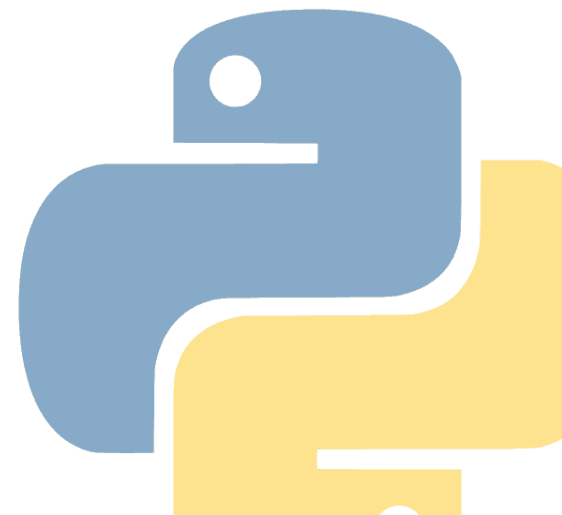
## 리스트 연산

**+**      **리스트 더하기(+)**

```
>>> a = [1, 2, 3]
>>> b = [4, 5, 6]
>>> a + b
[1, 2, 3, 4, 5, 6]
```

**\***      **리스트 반복하기(\*)**

```
>>> a = [1, 2, 3, "abc"]
>>> a * 3
[1, 2, 3, "abc", 1, 2, 3, "abc", 1, 2, 3, "abc"]
```



# Quiz

```
>> a = [1, 3, 2]
```

```
>>           
```

```
>> a  
[1, 3, 2, 4]
```

```
>>           
```

```
>> a  
[1, 2, 3, 4]
```

```
>>           
```

```
>> a  
[4, 3, 2, 1]
```

```
>>           
```

```
>> a  
[4, 3, 2]
```

```
a[3] = 4
```

```
a.append(4)
```

```
a.remove(1)
```

```
len(a)
```

```
a.count(1)
```

```
a.sort()
```

```
a = a * 4
```

```
del a[3]
```

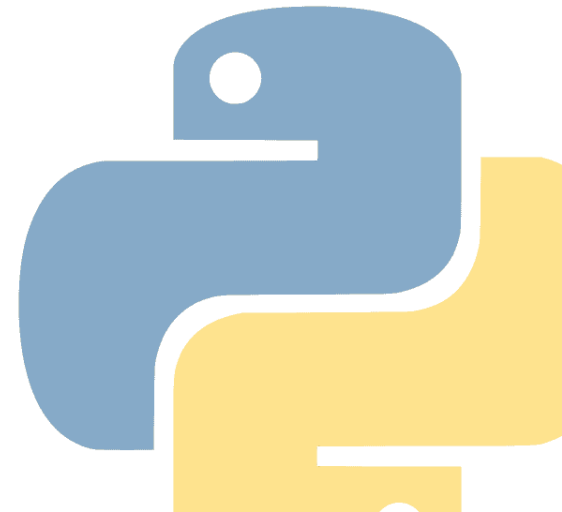
```
a.insert(3, 4)
```

```
a.pop()
```

```
a.index(4)
```

```
a.reverse()
```

```
a = a + [4]
```



## 다중 리스트 – 2차원 리스트

리스트의 요소가 리스트인 경우

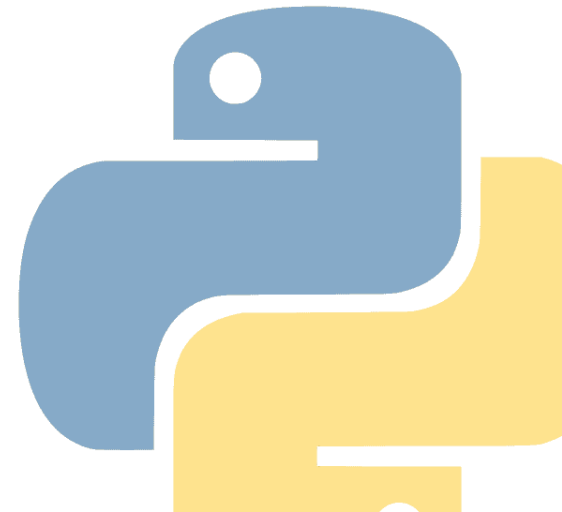
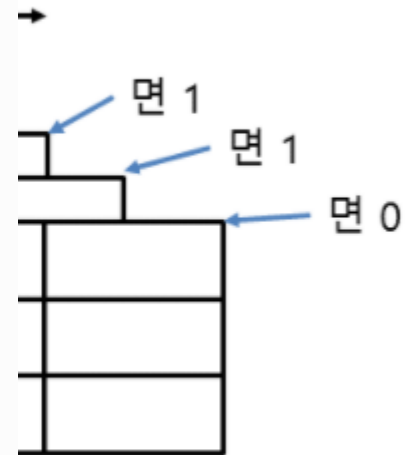
```
a = [[10, 20],  
      [500, 600, 700],  
      [9],  
      [30, 40],  
      [8],  
      [800, 900, 1000]]
```

가능!

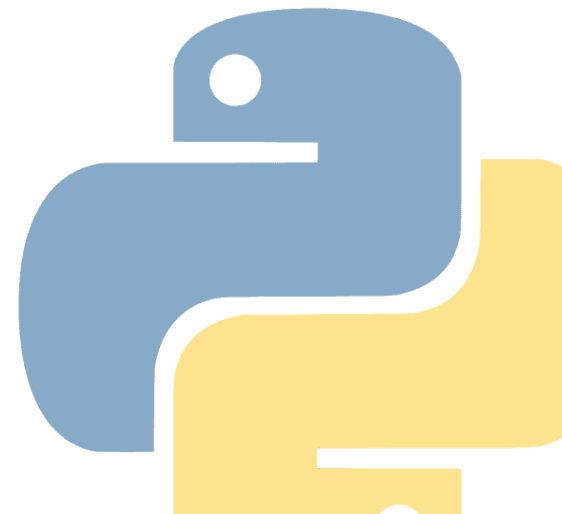


## 다중 리스트 - 3차원 리스트

```
a = [[ [ 0, 1, 2, 3],
        [ 4, 5, 6, 7],
        [ 8, 9, 10, 11]],
      [[12, 13, 14, 15],
        [16, 17, 18, 19],
        [20, 21, 22, 23]],
      [[24, 25, 26, 27],
        [28, 29, 30, 31],
        [32, 33, 34, 35]]]
```



# Loop Statement







참고자료 : <https://wikidocs.net/14>  
(점프 투 파이썬)

