

# 메서드

- 데이터 구조를 활용하기 위해서는 메서드를 활용
  - 메서드는 클래스 내부에 정의한 함수, 사실상 함수와 동일
  - 쉽게 설명하자면 객체의 기능
  - 객체.메서드() 형태 (ex : 교수.말하기())
- 메서드 구조는 예를 들어 s.replace(old, new[,count]) 로 되어있음
  - 대괄호 []안에 들어있는 인자는 선택적 인자로, 필수로 입력해야되는 내용은 아님

## 문자열 메서드

문법	설명
s.find(x)	x의 첫 번째 위치를 반환. 없으면, -1을 반환
s.index(x)	x의 첫 번째 위치를 반환. 없으면, 오류 발생
s.isalpha()	알파벳 문자 여부 *단순 알파벳이 아닌 유니코드 상 Letter (한국어도 포함)
s.isupper()	대문자 여부
s.islower()	소문자 여부
s.istitle()	타이틀 형식 여부

문법	설명
s.replace(old, new[,count])	바꿀 대상 글자를 새로운 글자로 바꿔서 반환
s.strip([chars])	공백이나 특정 문자를 제거
s.split(sep=None, maxsplit=-1)	공백이나 특정 문자를 기준으로 분리
'separator'.join([iterable])	구분자로 iterable을 합침
s.capitalize()	가장 첫 번째 글자를 대문자로 변경
s.title()	문자열 내 띄어쓰기 기준으로 각 단어의 첫글자는 대문자로, 나머지는 소문자로 변환
s.upper()	모두 대문자로 변경
s.lower()	모두 소문자로 변경
s.swapcase()	대↔소문자 서로 변경

## List 메서드

문법	설명
L.append(x)	리스트 마지막에 항목 x를 추가
L.insert(i, x)	리스트 인덱스 i에 항목 x를 삽입
L.remove(x)	리스트 가장 왼쪽에 있는 항목(첫 번째) x를 제거 항목이 존재하지 않을 경우, ValueError
L.pop()	리스트 가장 오른쪽에 있는 항목(마지막)을 반환 후 제거
L.pop(i)	리스트의 인덱스 i에 있는 항목을 반환 후 제거
L.extend(m)	순회형 m의 모든 항목들의 리스트 끝에 추가 (+와 같은 기능)
L.index(x, start, end)	리스트에 있는 항목 중 가장 왼쪽에 있는 항목 x의 인덱스를 반환
L.reverse()	리스트를 거꾸로 정렬
L.sort()	리스트를 정렬 (매개변수 이용가능)
L.count(x)	리스트에서 항목 x가 몇 개 존재하는지 갯수를 반환

## Set 메서드

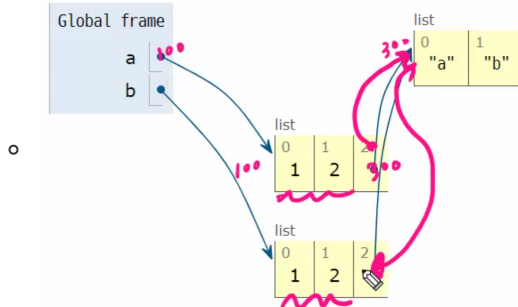
문법	설명
s.copy()	셋의 얇은 복사본을 반환
s.add(x)	항목 x가 셋 s에 없다면 추가
s.pop()	셋 s에서 랜덤하게 항목을 반환하고, 해당 항목을 제거 set이 비어 있을 경우, KeyError
s.remove(s)	항목 x를 셋 s에서 삭제 항목이 존재하지 않을 경우, KeyError
s.discard(x)	항목 x가 셋 s에 있는 경우, 항목 x를 셋s에서 삭제
s.update(t)	셋 t에 있는 모든 항목 중 셋 s에 없는 항목을 추가
s.clear()	모든 항목을 제거
s.isdisjoint(t)	셋 s가 셋 t의 서로 같은 항목을 하나라도 갖고 있지 않은 경우, True반환
s.issubset(t)	셋 s가 셋 t의 하위 셋인 경우, True반환
s.issuperset(t)	셋 s가 셋 t의 상위 셋인 경우, True반환

## 딕셔너리 메서드

문법	설명
d.clear()	모든 항목을 제거
d.copy()	딕셔너리 d의 얇은 복사본을 반환
d.keys()	딕셔너리 d의 모든 키를 담은 뷰를 반환
d.values()	딕셔너리 d의 모든 값을 담은 뷰를 반환
d.items()	딕셔너리 d의 모든 키-값의 쌍을 담은 뷰를 반환
d.get(k)	키 k의 값을 반환하는데, 키 k가 딕셔너리 d에 없을 경우 None을 반환
d.get(k, v)	키 k의 값을 반환하는데, 키 k가 딕셔너리 d에 없을 경우 v를 반환
d.pop(k)	키 k의 값을 반환하고 키 k인 항목을 딕셔너리 d에서 삭제하는데, 키 k가 딕셔너리 d에 없을 경우 KeyError를 발생
d.pop(k, v)	키 k의 값을 반환하고 키 k인 항목을 딕셔너리 d에서 삭제하는데, 키 k가 딕셔너리 d에 없을 경우 v를 반환
d.update([other])	딕셔너리 d의 값을 매핑하여 업데이트

## 얇은 복사와 깊은 복사

- 리스트는 얇은 복사로, 대입연산자로 변수에 다른 리스트 변수를 대입해도 주소만을 가리킬 뿐이기에 결국 새로운 변수가 생기는 것이 아니라 동기화가 되는 것임
- [:] 슬라이싱으로 리스트 변수를 새로운 변수에 복사를 할 수 있음. 하지만 여전히 '얇은 복사'임
  - 리스트 속 리스트, 즉 이중 리스트는 여전히 새로운 값으로 복사되는 것이 아니라 주소를 가리킴
  - 즉 맨 바깥 리스트만 진짜 복사가 되는 것임



- 진짜 "깊은 복사"는

```
import copy
a = [1, 2, 3]
b = copy.deepcopy(a)
```

- 이렇게 copy를 import 하여야만 한다.