기초 PYTHON 프로그래밍

11. 집합(set), 사전(dict)

1. 집합 (set)
2. 집합 사용하기
3. 사전 (dict)
4. 사전 사용하기

서강대학교
SOGANG UNIVERSITY

1. 집합 (set)

서강대학교 SOGANG UNIVERSITY

- 집합은 중복된 데이터를 가질 수 없고 순서가 없는 데이터 구조이다.
- 인덱스 기호 ([ ]), +, * 를 사용할 수 없다.
- in, not in, len()은 사용할 수 있다.
- mutable 자료형이다.
- 집합 생성하기

```
>>> s = {1,2,5}     # set 생성
>>> print(s)
{1, 2, 5}
>>> type(s)
<class 'set'>
```

```
>>> S = set()   # 빈 집합 생성
>>> type(S)
<class 'set'>

>>> A = {}        # 빈 사전
>>> type(A)
<class 'dict'>
```

2. 집합 사용하기

서강대학교 SOGANG UNIVERSITY

메소드	설명
add (x)	집합에 원소 x를 추가한다.
clear()	공집합으로 만든다.
copy ()	집합을 복사한다.
discard (x)	집합에서 원소 x를 삭제한다. 없는 원소를 삭제하려고 할 때에도 에러를 발생하지 않는다. (remove와 비교)
pop()	집합에서 임의의 원소를 하나 가져온다. 어떤 원소를 가져올 지 알 수 없다. 집합에서 그 원소는 삭제된다.
remove (x)	집합에서 원소 x를 삭제한다. 없는 원소를 삭제하려고 하면 에러가 발생한다.

```
〉〉〉 dir(set)
['__and__', '__class__', …… '__xor__', 'add', 'clear', 'copy', 'difference',
'difference_update', 'discard', 'intersection', 'intersection_update', 'isdisjoint',
'issubset', 'issuperset', 'pop', 'remove', 'symmetric_difference',
'symmetric_difference_update', 'union', 'update']
```

2. 집합 사용하기

서강대학교 SOGANG UNIVERSITY

- 집합에 원소 추가하기

```
>>> A = {4,2,6,8,3}
>>> print(A)
{8, 2, 3, 4, 6}
>>> A.add(5)        # 집합 A에 데이터 5를 추가한다.
>>> print(A)
{2, 3, 4, 5, 6, 8}
>>> A.add(3)        # 이미 있는 데이터를 추가하면 변동이 없다.
>>> print(A)
{2, 3, 4, 5, 6, 8}
```

2. 집합 사용하기

서강대학교 SOGANG UNIVERSITY

- 집합에서 원소 삭제하기

```
>>> A = {4,6,2,5,3}
>>> A.discard(5)
>>> print(A)
{2, 3, 4, 6}
>>> A.discard(7)     # 집합 A에 없는 원소 삭제해도 에러 없다.
>>> print(A)
{2, 3, 4, 6}

>>> A.remove(4)     # remove도 집합에서 원소를 삭제한다.
>>> print(A)
{2, 3, 6}
>>> A.remove(7)     # 집합에 없는 원소를 remove하면 에러발생함.
```

2. 집합 사용하기

서강대학교 SOGANG UNIVERSITY

◆ 집합에서 원소 삭제하기 / 공집합 만들기

```
〉〉〉 A = {1,3,5,6,4}
〉〉〉 A.pop()  # pop 메소드는 집합에서 임의의 원소를 반환하고 삭제한다.
1
〉〉〉 print(A)
{3, 4, 5, 6}
〉〉〉 A.clear()  # 집합 A를 공집합으로 만든다.
〉〉〉 print(A)
set()
```

3. 사전 (dict)

서강대학교 SOGANG UNIVERSITY

- 사전은 집합의 일종이다 (순서 개념이 없다).
- +, * 를 사용할 수 없다. 인덱스 기호([])는 사용한다.
- in, not in, len()은 사용할 수 있다.
- 사전에는 (키:값)의 쌍으로 하나의 데이터가 저장된다.
- 사전 예
  - 1반 25명, 2반 30명, 3반 27명이라는 정보를 저장하고자 한다.

```
>>> number = {1:25, 2:30, 3:27}
>>> type(number)
<class 'dict'>
```

3. 사전 (dict)

서강대학교 SOGANG UNIVERSITY

- 사전의 키
  - mutable 자료형은 '키'가 될 수 없다. (리스트, 집합, 사전)
  - 정수, 실수, bool, 복소수, 문자열, 튜플은 '키'가 될 수 있다.
- 사전의 값
  - 모든 자료형이 사전의 '값'이 될 수 있다.
- 빈 사전 생성하기

```
>>> mydict = {}          # mydict = dict()
>>> type(mydict)
<class 'dict'>
```

3. 사전 (dict)

서강대학교 SOGANG UNIVERSITY

◆ 사전에 아이템 추가 / 수정하기

```
>>> colorpen = {'red':2, 'blue':3, 'yellow':1}
>>> colorpen['green'] = 3    # 추가하기
>>> print(colorpen)
{'green': 3, 'red': 2, 'yellow': 1, 'blue': 3}
>>> colorpen['red'] = 4        # 수정하기
>>> print(colorpen)
{'green': 3, 'red': 4, 'yellow': 1, 'blue': 3}
```

3. 사전 (dict)

서강대학교 SOGANG UNIVERSITY

- 사전에 아이템 삭제하기 / in, not in, len()

```
>>> del colorpen['red']      # 삭제하기
>>> colorpen
{'yellow': 1, 'green': 4, 'blue': 3}
>>> len(colorpen)
3
>>> 'blue' in colorpen
True
>>> 'red' not in colorpen
True
```

4. 사전 사용하기

서강대학교 SOGANG UNIVERSITY

```
〉〉〉 dir(dict)
['__class__', '__contains__', …… '__subclasshook__', 'clear',
'copy', 'fromkeys', 'get', 'items', 'keys', 'pop', 'popitem',
'setdefault', 'update', 'values']
```

메소드	설명
clear ()	사전의 내용을 모두 지운다.
copy ()	사전을 복사한다.
items()	사전에 있는 모든 데이터의 (키,값)을 모두 반환한다.
keys()	사전에서 키만 반환한다.
values()	사전에서 값만 반환한다.
update(D)	사전 D를 추가한다.

4. 사전 사용하기

서강대학교 SOGANG UNIVERSITY

- clear() 메소드

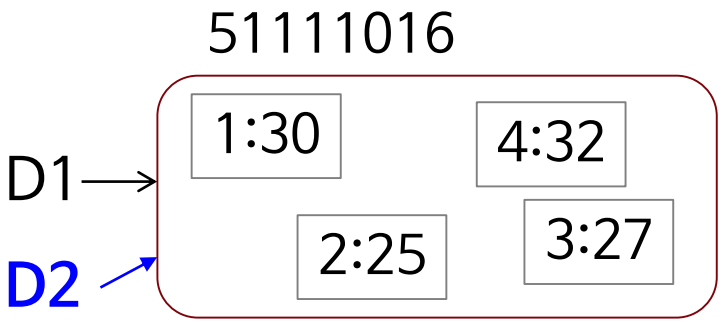
```
>>> D = {1:30, 2:25, 3:27}
>>> print(D)
{1: 30, 2: 25, 3: 27}
>>> D.clear()
>>> print(D)
{}
```

4. 사전 사용하기

서강대학교 SOGANG UNIVERSITY

- copy() 메소드

```
>>> D1 = {1:30, 2:25, 3:27}
>>> id(D1)
51111016
>>> D2 = D1
>>> id(D2)
51111016
>>> D2[4] = 32
>>> print(D1)
{1: 30, 2: 25, 3: 27, 4: 32}
>>> print(D2)
{1: 30, 2: 25, 3: 27, 4: 32}
```




```
>>> D1 = {1:30, 2:25, 3:27}
>>> D2 = D1.copy()
>>> D2[4] = 32
>>> print(D1)
{1: 30, 2: 25, 3: 27}
>>> print(D2)
{1: 30, 2: 25, 3: 27, 4: 32}
```

4. 사전 사용하기

서강대학교 SOGANG UNIVERSITY

- update(D) 메소드

```
>>> Voca = {1:'one', 2:'two', 3:'three'}
>>> Voca2 = {100:'hundred', 1000:'thousand'}
>>> Voca.update(Voca2)
>>> print(Voca)
{1: 'one', 2: 'two', 3: 'three', 1000: 'thousand', 100: 'hundred'}
>>> print(Voca2)
{1000: 'thousand', 100: 'hundred'}
```

4. 사전 사용하기

서강대학교 SOGANG UNIVERSITY

- items() - 사전에서 (키,값)을 쌍으로 반환한다.

사전 count는 A 영어 학원의 각 강좌 등록생 수를 저장하고 있다.

```
>>> count = {'toefl':40, 'toeic':50, 'sat':25}
>>> A = count.items()
>>> print(A)
dict_items([('sat', 25), ('toefl', 40), ('toeic', 50)])

>>> B = list(count.items())
>>> print(B)
[('sat', 25), ('toefl', 40), ('toeic', 50)]
```

4. 사전 사용하기

서강대학교 SOGANG UNIVERSITY

- keys() - 사전에서 키만을 반환한다.

사전 count는 A 영어 학원의 각 강좌 등록생 수를 저장하고 있다.

```
>>> count = {'toefl':40, 'toeic':50, 'sat':25}
>>> A = count.keys()
>>> print(A)
dict_keys(['sat', 'toefl', 'toeic'])

>>> B = list(count.keys())
>>> print(B)
['sat', 'toefl', 'toeic']
```

4. 사전 사용하기

서강대학교 SOGANG UNIVERSITY

- values() - 사전에서 값만을 반환한다.

사전 count는 A 영어 학원의 각 강좌 등록생 수를 저장하고 있다.

```
>>> A = count.values()
>>> print(A)
dict_values([25, 40, 50])

>>> B = list(count.values())
>>> print(B)
[25, 40, 50]
```