

10. 파이썬 자료형 - 집합(Set)

Made By. 김규일

- 집합의 특징은 순서가 없다는 점과 중복을 허용하지 않는다는 점이다.

```
>>> s2 = set("Hello")
>>> s2
{'e', 'H', 'l', 'o'}
```

이는 중복을 허용하지 않아 l이 1개 삭제가 되었고 순서가 없어서 출력 시 순서가 뒤죽박죽이라는 것이다.

비어 있는 집합은 s = set() 으로 만들 수 있다.

- 집합은 순서가 없기 때문에 인덱싱을 지원하지 않는다. (이 점은 딕셔너리와 똑같다.)
- 따라서 리스트나 튜플로 변환 후 사용할 수 있다.

```
>>> s1 = set([1,2,3])
>>> l1 = list(s1)
>>> l1
[1, 2, 3]
>>> l1[0]
1
>>> t1 = tuple(s1)
>>> t1
(1, 2, 3)
>>> t1[0]
1
```

교집합, 합집합, 차집합 구하기

1. 교집합

```
>>> s1 = set([1, 2, 3, 4, 5, 6])
>>> s2 = set([4, 5, 6, 7, 8, 9])

>>> s1 & s2
{4, 5, 6}
```

```
## 교집합은 & 기호를 사용

>>> s1.intersection(s2)
{4, 5, 6}

## 또는 intersection 함수를 사용
```

2. 합집합

```
>>> s1 = set([1, 2, 3, 4, 5, 6])
>>> s2 = set([4, 5, 6, 7, 8, 9])

>>> s1 | s2
{1, 2, 3, 4, 5, 6, 7, 8, 9}

## 합집합은 | 기호를 사용

>>> s1.union(s2)
{1, 2, 3, 4, 5, 6, 7, 8, 9}

## 또는 union 함수를 사용
```

3. 차집합

```
>>> s1 = set([1, 2, 3, 4, 5, 6])
>>> s2 = set([4, 5, 6, 7, 8, 9])

>>> s1 - s2
{1, 2, 3}
>>> s2 - s1
{8, 9, 7}

## 차집합은 - 기호를 사용

>>> s1.difference(s2)
{1, 2, 3}
>>> s2.difference(s1)
{8, 9, 7}

## 또는 difference 함수를 사용
```

집합 관련 함수들

- 값 1개 추가하기(add)

```
>>> s1 = set([1, 2, 3])
>>> s1.add(4)
>>> s1
{1, 2, 3, 4}
```

- 값 여러 개 추가하기(update)

```
>>> s1 = set([1, 2, 3])
>>> s1.update([4, 5, 6])
>>> s1
{1, 2, 3, 4, 5, 6}
```

- 특정 값 제거하기(remove)

```
>>> s1 = set([1, 2, 3])
>>> s1.remove(2)
>>> s1
{1, 3}
```