Python程序设计

集合

刘安
苏州大学，计算机科学与技术学院

http://web.suda.edu.cn/anliu/

本节涉及到的知识点

- 集合的基本概念和主要用途
- 集合的构造方法
- 集合的常见方法
- 集合的应用
  - 去除重复元素
  - 计算Jaccard相似度

https://docs.python.org/3/library/stdtypes.html#set

集合的基本概念

- 一组具有唯一性、无序性和不变性的对象
  - 唯一性：对象不能重复
  - 不变性：对象必须是不可变类型（列表和字典不能作为集合的元素）
  - 无序性：1）对象在集合中没有固定的位置，不支持索引、切片等操作；2）对象放入集合的顺序不确定
- set：可变集合，能加入或者删除元素
- frozenset：不可变集合，构造后无法改变，可以作为字典的键或作为其它集合的元素

集合的主要用途

- 高效的成员测试
  - 通过运算符in，效率比列表快很多，和字典类似
- 实现数学集合的操作：并集、交集、差集、子集等
- 去除序列（比如列表、字符串）中的重复元素

集合的构造方法

- 放在{ }之间的用逗号分开的一组对象（只能创建非空集合，因为{}是一个空字典）

- 使用函数set([iterable])和frozenset([iterable])：使用iterable中的所有元素创建一个集合。如果没有指定iterable，创建空集合

```
>>> vowels = {'a', 'e', 'i', 'o', 'u'}
>>> vowels
{'i', 'o', 'u', 'a', 'e'}
>>>
>>> letters = set('alice')
>>> letters
{'l', 'i', 'c', 'a', 'e'}
>>>
>>> set()
set()
```

集合的常见方法

- len(s)：返回s中元素的个数
- x in S：如果x在S中，返回True，否则False
- add(x)：将对象x加入集合中
- discard(x)：如果x在集合中，删除它

```
>>> len(vowels)
5
>>> 'e' in vowels
True
>>> vowels.add('y')
>>> vowels
{'i', 'o', 'y', 'u', 'a', 'e'}
>>> vowels.discard('y')
>>> vowels
{'i', 'o', 'u', 'a', 'e'}
```

集合运算

- 集合运算（比如并、交、差、子集）有两种风格：函数形式和运算符形式，前者的参数可以是可迭代对象，后者的参数只能是集合

```
>>> S = set('abc')
>>> T = set('cd')
>>> S | T
{'c', 'a', 'b', 'd'}
>>> S & T
{'c'}
>>> S - T
{'b', 'a'}
>>> S ^ T
{'b', 'a', 'd'}
```

```
>>> S.union('cd')
{'b', 'd', 'c', 'a'}
>>> S.intersection('cd')
{'c'}
>>> S.difference('cd')
{'b', 'a'}
>>> S.symmetric_difference('cd')
{'b', 'a', 'd'}
>>> S.issubset('abcd')
True
```

去除集合（Collection）的重复元素

- 编写一个函数，接受一个列表，去除该列表中重复的元素，返回新的列表

```
1 def remove_duplication(L):
2     res = []
3     for e in L:
4         if e not in res:
5             res.append(e)
6     return res
7
8 def remove_duplication_v1(L):
9     return list(set(L))
```

```
>>> set([1, 2, 1, 3, 2, 4, 5])
{1, 2, 3, 4, 5}
>>> set('hello')
{'e', 'o', 'l', 'h'}
```

注意！
该方法只适用于 !!
包含不可变类型元素的列表

Jaccard相似度

- 集合S和T的Jaccard相似度定义为$|S \cap T| / |S \cup T|$
- 编写一个函数，接受两个集合，返回它们的Jaccard相似度

```
1 def jaccard_sim(S, T):
2     if S.union(T):
3         return (len(S.intersection(T)) /
4                 len(S.union(T)))
```

!! 注意！该方法只适用于包含不可变类型元素的列表