

量化金融 python 基础 5-Set

2023 年 6 月 14 日

1 Data Structure 数据结构 - Set 集合

Python set 是无序的不可变集合，且内部元素具有唯一性，即不可重复。

set 也是用 {} 表示, 但不像字典一样输入键值对 key-value pairs，而是输入单独的元素内容。

```
[1]: contract_set1 = {"c2309", "m2309", "lh2309", "pp2309", "i2309", "i2309"}
      contract_set1
```

```
[1]: {'c2309', 'i2309', 'lh2309', 'm2309', 'pp2309'}
```

由于 set 是无序的，所以不能用索引访问内部元素。

可以通过 for 循环打印集合内所有元素。

集合最大的用处是做去重，以及集合运算。

```
[2]: contract_set2 = {"m2309", "pp2309", "AP310", "CF309"}
      # 取交集

      # 方法 1
      print(contract_set1 & contract_set2)
      # 方法 2
      print(contract_set1.intersection(contract_set2))
```

```
{'m2309', 'pp2309'}
```

```
{'m2309', 'pp2309'}
```

```
[3]: # 并集
```

```
# 方法 1
```

```
print(contract_set1 | contract_set2)
# 方法 2
print(contract_set1.union(contract_set2))
```

```
{'m2309', 'c2309', 'CF309', 'pp2309', 'lh2309', 'i2309', 'AP310'}
{'m2309', 'c2309', 'CF309', 'pp2309', 'lh2309', 'i2309', 'AP310'}
```

```
[4]: # 差集
# 集合 1 有的, 集合 2 没有的元素

# 方法 1
print(contract_set1.difference(contract_set2))
# 方法 2
print(contract_set1 - contract_set2)
```

```
{'i2309', 'c2309', 'lh2309'}
{'i2309', 'c2309', 'lh2309'}
```

```
[5]: # 对称差分集合
# 取两个集合都有的, 但不在两者交集的元素。即, 仅存于任一集合的元素。

# 方法 1
print(contract_set1.symmetric_difference(contract_set2))
# 方法 2
print(contract_set1 ^ contract_set2)
```

```
{'c2309', 'CF309', 'lh2309', 'i2309', 'AP310'}
{'c2309', 'CF309', 'lh2309', 'i2309', 'AP310'}
```

有时候会看到类似于

`contract_set1 &= contract_set2`, 即 `set_a &= set_b`.

该含义为 `set_a = set_a & set_b`.

同理于 `set_a |= set_b`

`set_a ^= set_b`

```
[6]: # 子集判断
contract_set1_subset = {"c2309", "i2309"}
```

```
# contract_set1_subset 是否包含于 contract_set1, 是的话返回 True
contract_set1_subset.issubset(contract_set1)
```

[6]: True

```
[7]: # 超集判断
# contract_set1 是 contract_set1_subset 的超集, 这里返回 True
contract_set1.issuperset(contract_set1_subset)
```

[7]: True

```
[8]: # 两者没有交集返回 True
{"AP310", "CF309"}.isdisjoint(contract_set1)
```

[8]: True

注意:

由于集合的元素是不可变的, 如果想包含列表型的元素, 需要先转换成元组。

```
[9]: # 直接向集合内添加 list 会报错
list1 = ["plum", "avocado"]
contract_set1.add(list1)
```

```
-----
TypeError                                Traceback (most recent call last)
Cell In[9], line 3
      1 # 直接向集合内添加 list 会报错
      2 list1 = ["plum", "avocado"]
----> 3 contract_set1.add(list1)

TypeError: unhashable type: 'list'
```

```
[10]: contract_set1.add(tuple(list1))
contract_set1
```

[10]: {('plum', 'avocado'), 'c2309', 'i2309', 'lh2309', 'm2309', 'pp2309'}

```
[11]: # 移除集合内某个元素
contract_set1.remove("c2309")
contract_set1
```

```
[11]: {('plum', 'avocado'), 'i2309', 'lh2309', 'm2309', 'pp2309'}
```

```
[12]: # 清空集合
contract_set1.clear()
contract_set1
```

```
[12]: set()
```

2 Summary

- `set1 & set2`, `set1.intersection(set2)`
- `set1 | set2`, `set1.union(set2)`
- `set1 - set2`, `set1.difference(set2)`
- `set1 ^ set2`, `set1.symmetric_difference(set2)`
- `set1 &= set2`, `set1 |= set2`
- `set1.issubset(set2)`
- `set.issuperset(set2)`
- `set1.isdisjoint(set2)`
- `set.add()`
- `set.remove()`
- `set.clear()`