

知识点

1. 创建集合

2. 添加集合元素

3. 删除集合元素

4. 集合运算

4.集合

集合（**set**）是0个或多个对象引用的无序组合，这些对象所引用的对象都是可哈希运算的。集合是可变的，因此可以很容易地添加或移除数据项，但由于其中的项是无序的，因此，没有索引位置的概念。

在集合中，任何元素都没有重复，这是集合的一个非常重要的特点。

set和**dict**类似，是一组**key**的集合，但不存储**value**，且在**set**中没有重复的**key**。

4.集合

4.1创建集合

可以通过调用集合的构造函数来创建一个集合。和前文许多其他数据结构不同，创建集合没有快捷方式。因此，要创建集合，必须使用**set**构造函数。**set**构造函数至多有一个参数。如果没有参数，**set**会创建空集。如果有一个参数，那么参数必须是可迭代的，例如字符串或列表，可迭代对象的元素将生成集合的成员。

4.集合

4.1创建集合

- 没有参数，**set**会创建空集。

```
>>> nullSet=set()
```

```
>>> nullSet
```

```
set()
```

- 提供一个**str**作为输入集合，创建一个**set**。

```
>>> a_set=set('abcd')
```

```
>>> a_set
```

```
{'a', 'c', 'b', 'd'}
```

- 提供一个**list**作为输入集合，创建一个**set**。

```
>>> s = set([1, 2, 3])
```

```
>>> s
```

```
{1, 2, 3}
```

- 重复元素在**set**中自动被过滤。

```
>>> s = set([1, 1, 2, 2, 3, 3])
```

```
>>> s
```

```
{1, 2, 3}
```

4.集合

4.2添加元素

通过`add(key)`方法可以添加元素到`set`中，可以重复添加，但不会有效果。例如：

```
>>> s.add(4)
>>> s
{1, 2, 3, 4}
>>> s.add(4)
>>> s
{1, 2, 3, 4}
```

4.集合

4.3删除元素

- 通过`remove(key)`方法可以删除元素。例如：
 - `discard(key)`也可删除元素，不同的是，如果删除的元素不在集合中，`remove`会报错，`discard`不会报错。例如：
 - `clear()`删除集合的所有元素（使它成为空集）。
-

4.集合

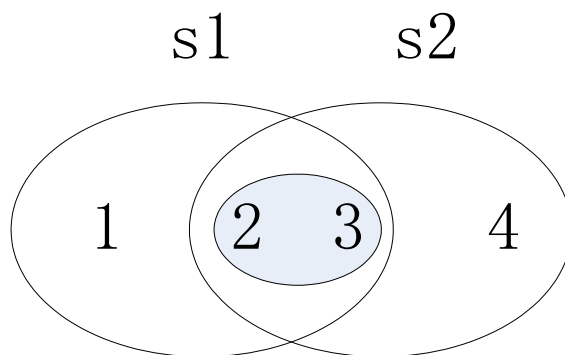
4.4典型的集合运算符

- **len():** 和所有集合类型一样，**len**函数可以确定集合中的元素数量。
 - **in:** 判断某元素是否在集合中。**in**运算符根据元素是否在集合中返回布尔值**True**或**False**。
 - **for:** 和所有集合类型一样，**for**语句能遍历集合中的元素。
-

4.集合

4.5典型的数学集合运算

交集

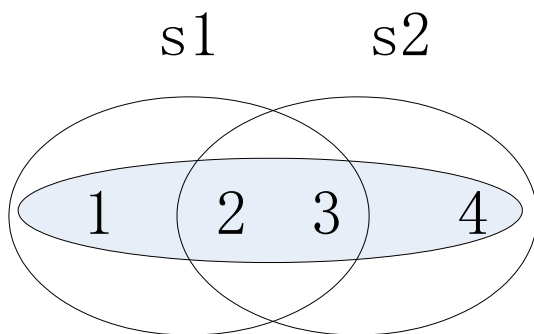


```
>>> s1 = set([1, 2, 3])  
>>> s2 = set([2, 3, 4])  
>>> s1.intersection(s2)  
{2, 3}  
>>> s1 & s2  
{2, 3}
```


4.集合

4.5典型的数学集合运算

并集

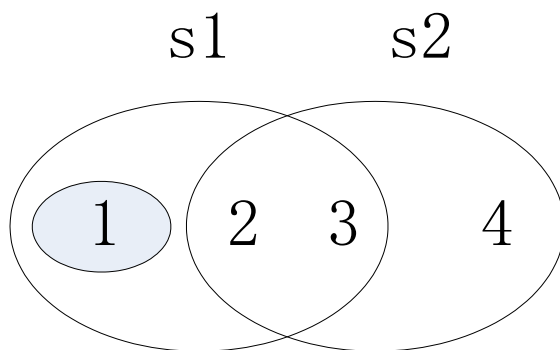


```
>>> s1.union(s2)
{1, 2, 3, 4}
>>> s1 | s2
{1, 2, 3, 4}
```

4.集合

4.5典型的数学集合运算

差集

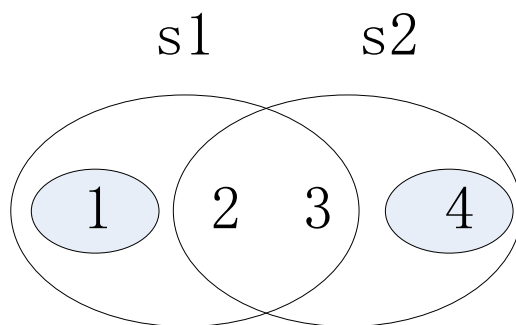


```
>>> s1.difference(s2)
{1}
>>> s2.difference(s1)
{4}
>>> s1-s2
{1}
>>> s2-s1
{4}
```

4.集合

4.5典型的数学集合运算

对称差

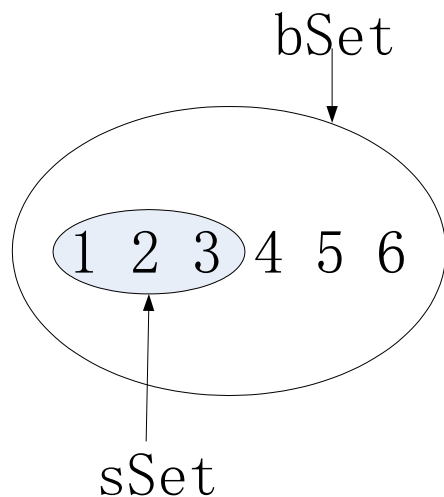


```
>>> s1.symmetric_difference(s2)
{1, 4}
>>> s2.symmetric_difference(s1)
{1, 4}
```

4.集合

4.5典型的数学集合运算

子集和超集



```
>>> sSet=set([1,2,3])
>>> bSet=set([1,2,3,4,5,6])
>>> sSet.issubset(bSet)
True
>>> bSet.issubset(sSet)
False
>>> sSet.issuperset(bSet)
False
>>> bSet.issuperset(sSet)
True
>>> sSet.issubset(sSet)
True
>>> sSet.issuperset(sSet)
True
```

代码练习题

请创建如下两个集合，并输出它们的交集。

```
s1 = {'a', 'b', 'c'}
```

```
s2 = {'b', 'd', 'e'}
```

答案：

```
s = set('abcd')
```

或者

```
s = set(['a', 'b', 'c', 'd'])
```

代码练习题

请创建如下两个集合，并输出s1与s2的差集。

```
s1 = {'a', 'b', 'c'}
```

```
s2 = {'b', 'd', 'e'}
```

答案：

```
s1.difference(s2)
```

或者

```
s1-s2
```
