

## Python核心数据类型-列表

### 关于列表的重要属性

1. 列表是任意对象的有序集合，列表中可以包含任何种类的对象，数字 字符串 或者是其他列表，列表所包含的每一项都保持了从左到右的位置顺序
2. 列表可以通过偏移来读取其中的元素，也可以通过分片读取列表中的某一部分元素
3. 可变长度， 异构 任意嵌套，列表可以增长或者缩短(长度可变)，并且可以包含任何类型的对象(异构)，因为列表能够包含其他复杂的对象，又支持任意嵌套，所以可以在列表中嵌套多层列表
4. 可变类型，列表支持在原处修改，

### 以下函数应用列表

1. `len()` ##返回列表的长度

2. `list()` # 将 `str` 元组 集合变成列表

`list` 函数 将元组 字符串 集合变成列表

```
In [74]: list('spam')
```

```
Out[74]: ['s', 'p', 'a', 'm']
```

```
In [75]: list(set([1, 2, 3]))
```

```
Out[75]: [1, 2, 3]
```

```
In [76]: list((4, 5, 6))
```

```
Out[76]: [4, 5, 6]
```

列表的基本操作:

由于列表是序列，它支持很多与字符串相同的操作 如 + 与 \*

```
In [14]: x = [1, 2, 3]
```

```
In [15]: y = [2, 3, 4]
```

```
In [16]: x + y
```

```
Out[16]: [1, 2, 3, 2, 3, 4]
```

```
In [17]: x * 3
```

```
Out[17]: [1, 2, 3, 1, 2, 3, 1, 2, 3]
```

索引 分片 矩阵 原处修改

```
In [20]: x
```

```
Out[20]: [1, 2, 3]
```

```
In [21]: x[1]
```

```
Out[21]: 2
```

```
In [22]: x[1:2]
```

```
Out[22]: [2]
```

```
In [23]: x = [[1,2,3,4], ['a', 'b', 'c']]
```

```
In [24]: x[0][1]
```

```
Out[24]: 2
```

```
In [26]: x[1][1] = 'apple'
```

```
In [27]: x
```

```
Out[27]: [[1, 2, 3, 4], ['a', 'apple', 'c']]
```

## 列表的常用方法

### 1. append

```
In [21]: a
```

```
Out[21]: [1, 2, 4, 5]
```

```
In [22]: a.append('c')
```

```
In [23]: a.append('d')
```

```
In [24]: a
```

```
Out[24]: [1, 2, 4, 5, 'c', 'd']
```

2. extend() #在列表末尾一次性追加另一个序列中的多个值 列表变为追加后新序列后的列表

```
In [51]: a
```

```
Out[51]: ['a', 'd', 'c', 4, 2, 'e']
```

```
In [52]: b = [11, 12, 13]
```

```
In [54]: b
```

```
Out[54]: [11, 12, 13]
```

```
In [55]: a.extend(b)
```

```
In [56]: a
```

```
Out[56]: ['a', 'd', 'c', 4, 2, 'e', 11, 12, 13]
```

### 3.insert #将对象插入列表

```
In [47]: a.insert(index, value)    #用法
```

```
In [45]: a.insert(0, 'a')
```

```
In [46]: a
```

```
Out[46]: ['a', 'd', 'c', 4, 2]
```

### 4. remove

remove #删除列表中的指定的元素

pop #删除列表中的最后一个元素

```
In [33]: a
```

```
Out[33]: ['d', 'c', 5, 4, 2, 1]
```

```
In [34]: a.remove(5)
```

```
In [35]: a
Out[35]: ['d', 'c', 4, 2, 1]
```

```
5.pop()
In [36]: a.pop()
Out[36]: 1
In [37]: a
Out[37]: ['d', 'c', 4, 2]
```

6. count 统计某个元素在列表中出现的次数

```
In [38]: a
Out[38]: ['d', 'c', 4, 2]
In [39]: a.count(2)
Out[39]: 1
```

7.index 从列表中找出某个值第一个匹配项的索引位置

```
In [44]: a.index(4)
Out[44]: 2
```

8. sort

对列表中的元素进行正向 反向排序

```
In [26]: a
Out[26]: [1, 2, 4, 5, 'c', 'd']
In [27]: a.sort()
In [28]: a
Out[28]: [1, 2, 4, 5, 'c', 'd']
```

9.reverse

```
In [29]: a.reverse()
In [30]: a
Out[30]: ['d', 'c', 5, 4, 2, 1]
```