

一、基本概念

1. 元组：方括号法[]，指明类型法list()

不可变：(1, 99, 5)；

2. 列表：圆括号法()，指明类型法tuple()

可变，可变容器

› 索引

`alist[n]或atuple[n]`

- 可以用赋值语句给列表中的任何一个位置重新赋值
- 但元组属于不可变类型，索引只能获取对应位置中的数据值，不可重新赋值

› 切片

`alist[start : end : step]`

`atuple[start : end : step]`

3. [x:y:s]

[x:y:s] 意思是从X到Y，步长是S，`print(str([::-1]))`的意思是从后往前一个个打印。

01. `print(str([0::-1]))`的意思是从第一个开始从后往前打印，因为第一个之前没有了，所以只有一个a,如果是+1就是从头开始全部打印。

02. Python中[: n]、[m :]、[-1]、[:-1]、[::-1]、[2::-1]和[1:]的含义：

03.

[m :] 代表列表中的第m+1项到最后一项

[: n] 代表列表中的第一项到第n项

例子：

```
a=[1,2,3,4,5]
```

```
print(a)
```

```
[ 1 2 3 4 5 ]
```

04.

```
print(a[-1])
```

 取最后一个元素

结果：[5]

```
print(a[:-1])
```

 除了最后一个取全部

结果：[1 2 3 4]

```
print(a[::-1])
```

 取从后向前（相反）的元素

结果：[5 4 3 2 1]

```
print(a[2::-1])
```

 取从下标为2的元素翻转读取

结果：[3 2 1]

`print(a[1:])` 取第二个到最后一个元素

结果: [2 3 4 5]

4注意: 数组内地元素是用方括号[], 而不是圆括号();

二、操作方法

列表:

方法名称	使用例子	说明
append	<code>alist.append(item)</code>	列表末尾添加元素
insert	<code>alist.insert(i,item)</code>	列表中i位置插入元素
pop	<code>alist.pop()</code>	删除最后一个元素, 并返回其值
pop	<code>alist.pop(i)</code>	删除第i个元素, 并返回其值
sort	<code>alist.sort()</code>	将表中元素排序
reverse	<code>alist.reverse()</code>	将表中元素反向排列
del	<code>del alist[i]</code>	删除第i个元素
index	<code>alist.index(item)</code>	找到item的首次出现位置
count	<code>alist.count(item)</code>	返回item在列表中出现的次数
remove	<code>alist.remove(item)</code>	将item的首次出现删除

列表和元组的操作：查找和计算

› 查找

- in操作: 判断某个元素是否存在于列表/元组中
- index操作: 指定的数据在列表/元组的哪个位置
- count操作: 指定的数据在列表/元组中出现过几次

› 计算

sum函数: 将列表中所有的数据元素累加

min/max函数: 返回列表中最小/最大的数据元素