

# 操作列表

## 目录

- 操作列表
  - 目录
  - 1、遍历整个列表
  - 2、注意缩进
  - 3、循环错误
  - 4、创建数值列表
    - 1、range()使用
    - 2、range生成列表
    - 3、对数字列表进行简单的统计
    - 4、列表解析
  - 5、使用列表的一部分
    - 1、切片
    - 2、遍历切片
    - 3、使用切片复制列表
  - 5、元组

---

## 1、遍历整个列表

```
names = ['ff', 'ffffff', 'sdss']

for name in names:
    print(name)
```

## 2、注意缩进

Python根据缩进来判断代码行与前一个代码行的关系。

```
names = ['ff', 'ffffff', 'sdss']

for name in names:
    print(name)
    print(name+' kk')
print("hello")
```

hello 只执行一次，因为其已经脱离循环

## 3、循环错误

遗漏了冒号

```
for name in names: #注意冒号
    print(name)
```

## 4、创建数值列表

需要存储一组数字的原因有很多：

- 在游戏中，需要跟踪每个角色的位置，还可能跟踪玩家的几个最高得分。
- 在数据可视化中，处理的几乎都是由数字（如温度、距离、人口数量、经度和纬度等）组成的集合。

列表非常适合用于存储数字集合，而Python提供了很多工具，可帮助你高效地处理数字列表。

### 1、range()使用

使用range()生成从1开始到9结束，最后一个不会输出

```
for i in range(1,10):
    print(i)
```

产生 +N 的数字序列

```
N = 2
for i in range(1, 10, N):
    print(i)
```

## 2、range生成列表

将range生成的数字序列转为列表

```
list = list(range(1,10))  
print(list)
```

## 3、对数字列表进行简单的统计

```
array = range(1,10)  
print(min(array))  
print(max(array))  
print(sum(array))
```

## 4、列表解析

```
# 平常使用的求平方数  
for i in range(1, 10):  
    print(i**3)  
#使用列表解析  
sqr = [i**3 for i in range(1, 10)]  
print(sqr)
```

---

## 5、使用列表的一部分

### 1、切片

```
list = list(range(1, 10))
print(list)
# 不赋值代表从0开始
print(list[:3])
# 第二个数到第七个数
print(list[2:8])

# 负数代表从末尾开始的第几个数
print(list[-5:-3])
```

## 2、遍历切片

```
list = list(range(1, 10))
for value in list[:4]:
    print(value)
```

## 3、使用切片复制列表

如果希望两个列表独立使用的话，不可以直接使用赋值语句，因为赋值语句会将两个变量指向同一个地址，并不会为接收赋值的变量创建新的地址，所以两个列表还会使用同一个列表

```
list = list(range(1, 10))
list2 = list

list.append(11)

print(list)
print(list2)
```

可以使用切片来解决

```
list = list(range(1, 10))  
list2 = list[:]  
  
list.append(11)  
  
print(list)  
print(list2)
```

---

## 5、元组

```
dimensions = (20, 522)  
print(dimensions[0])  
# 元组不支持修改元组内的元素  
# dimensions[0] = 30  
  
# 只能通过重新赋值改变值  
dimensions = (30, 522)  
print(dimensions[0])
```