**一、列表list**

**1创建列表**

['abc', 123] 字面量创建列表

**2遍历**

len(x)

[1] 返回索引1的元素

max(x) 返回列表最大值

min(x) 返回列表最小值

.count(e) 返回元素出现的次数

.index(e) 返回元素对应索引

**3判断**

operator.eq(x,y) 判断两个列表是否相等

e in [...] 判断是列表的元素

**4操作**

逻辑操作

+ 返回合并后的列表

[index:index] 返回截取的子列表

[1:] 返回1到-1索引的子列表

[:3] 返回0到3索引的子列表

[1:3] 返回1到3索引的子列表

.extend(x) 合并另一列表到尾部

.copy() 返回列表的一份拷贝

元素操作

.append(e) 像列表增加一个元素

del list[1] 删除列表索引为1的元素

.insert(index, e) 将元素插入指定索引

map(func, x) 用func修改列表x的所有元素

.pop(<index=-1>) 移除一个元素，缺省最后一个元素

.remove(e) 移除匹配的第一个元素

.clear() 清除所有元素

整理操作

.reverse() 逆序列表

.sort() 排序列表

类型操作

str.join(x) 将列表元素用str连接

**二、元祖tuple(元素不能修改)**

**1元组创建**

('abc', 123) 字面量创建元组

**2遍历(见列表)**

**3判断(见列表)**

**4操作(建列表)**

类型操作

list(x) 将元祖转化为列表

**三、集合set(无序不重复)**

**1集合创建**

{1,2,3} 字面量创建集合

set(1,2,3) 构造器构造

**2遍历**

**3判断**

.isdisjoint(y) 判断与y没有交集

.issubset(y) 判断为y的子集

.issuperset(y) 判断为y的父集

**4操作**

逻辑操作

.difference(y) 返回与y的差集

.difference\_update(y) 与y的差集(修改原集合)

.discard(e) 移除指定元素

.intersection(y) 返回与y的交集

.intersection\_update(y) 与y的交集(修改原集合)

.symmetric\_difference(y) 返回与y的非交集

.symmetric\_difference\_update(y) 与y的非交集(修改原集合)

.union(y) 返回与y的并集

.update(y) 与y的并集(修改原集合)

元素操作

.add(e) 添加一个元素

.clear() 清空集合

.copy() 返回集合的一个拷贝

.pop() 随机移除元素

.remove(e) 移除指定元素

类型转换

set(list) 将列表转化为集合