## 查找

L.index(v [, begin[, end]]) | 返回对应元素的索引下标, begin为开始索引，end为结束索引,当 value 不存在时触发ValueError错误

L.count(x) | 用于统计某个元素在列表中出现的次数

L.pop([index]) | 删除索引对应的元素，如果不加索引，默认删除最后元素，同时返回删除元素的引用关系

## 修改

L.insert(index, obj) | 将某个元素插放到列表中指定的位置

L.extend(lst) | 向列表追加另一个列表,扩展列表用于在列表末尾一次性追加另一个序列中的多个值

L.remove(x) | 从列表中删除第一次出现在列表中的值

L.clear() | 清空列表,等同于 L[:] = []

L.sort(reverse=False) | 将列表中的元素进行排序，默认顺序按值的小到大的顺序排列

list.sort(cmp=None, key=None, reverse=False)

L.reverse() | 列表的反转，用来改变原列表的先后顺序

## 拷贝

L.copy() | 复制此列表（只复制一层，不会复制深层对象)

print(**'列表方法'**)

*# 查找方法:拥有返回值*

print(**'pop'**,list\_3.pop(1))

print(**'count'**,list\_3.count(8))

print(**'index'**,list\_3.index(8))

*#修改方法:返回值为None*

list\_3.insert(8,10) *#在索引8插入10*

print(**'insert'**,list\_3)

list\_3.sort()

print(**'sort'**,list\_3)

print(**'sort'**,list\_3.sort())

list\_3.reverse()

print(**'reverse'**,list\_3)

list\_3.remove(10)

print(**'remove'**,list\_3)

*# list\_3.clear()*

*# print('clear',list\_3.clear())*

*# 拷贝方法:拥有返回值*

print(**'copy'**,list\_3.copy())