

L7-3 | 忍无可忍

# 模块5：列表排序 & 模块6：你能有多快



ANALYTICAL  
INTELLECTUALLY CURIOUS  
CREATIVE



## 项目内容

### 🔪 项目内容

列表排序

sorted()命令

sum()命令

### 单词卡

sum:总和

sort:排序

player:运动员

record:记录





## 排序



### 概念讲解

- 排序就是将一组杂乱无章的数据按照一定规律组织起来。

例如，将原始人们按照身高从低到高排列，这就是排序。



如何使用代码排序呢？



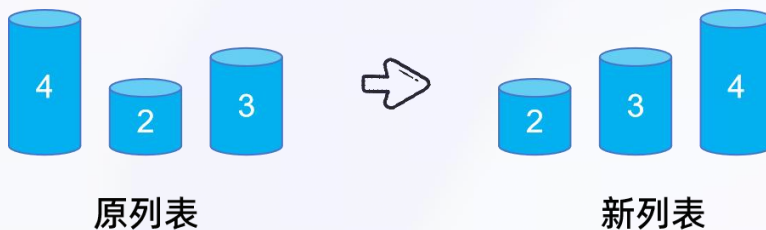
## 列表排序



### 概念讲解

每次排序时，在原列表中找到最小（大）的元素，将其从原列表中移除，添加到新列表的末尾。不断重复上述步骤，直到排序完成。

那么我们怎么排序才能让下面的列表变成有顺序的新列表呢？

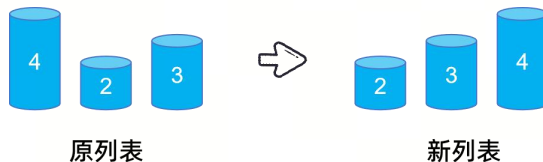




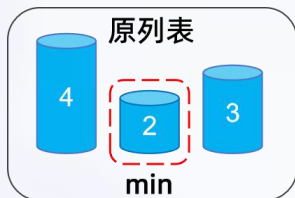
## 列表排序

### 示例讲解

将原列表按从小到大的顺序排列  
以此为例讲解过程

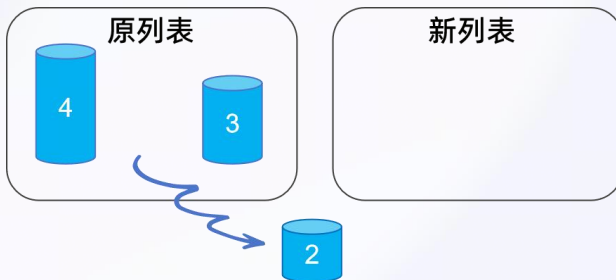


#### ① 从列表中找到最小的元素2



新列表

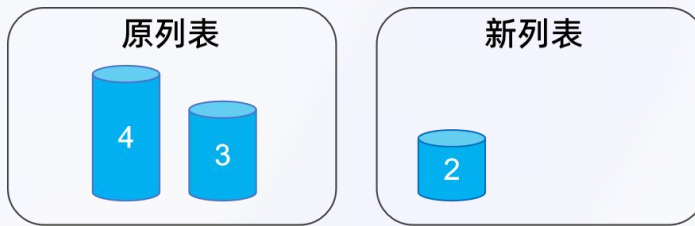
#### ② 将元素2从原列表移除



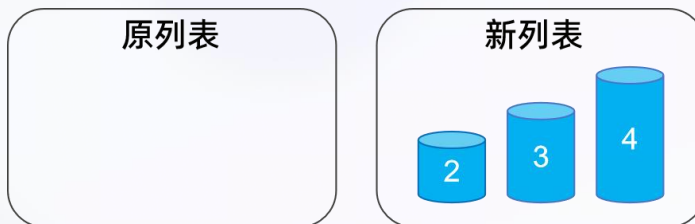


## 列表排序

### ③ 让元素2排在新列表末尾



### ④ 一共有3个元素，所以重复步骤①②③ 3次，得到按从小到大排列的新列表，排序完毕





## 列表排序

### 总结

- 从列表中找到最小的元素
- 将最小的元素从原列表移除
- 让最小的元素排在新列表末尾
- 列表中有几个元素，重复上述步骤几次

### </ 代码示例 />

```
for i in range(3):
```

```
    h = min(height) ———> 找到列表height的最小值
```

```
    height.remove(h) ———> 将最小值从列表height中删除
```

```
    height2.append(h) ———> 将最小值添加到列表height2中
```





## sorted()命令

### 概念讲解



sorted()命令可以得到按从小到大排序后的新列表。



用法：sorted(列表名)



注意：sorted()命令不会改变原来的列表。

### </ 代码示例 />

```
l = [1, 13, 5, 12, 7]
```

```
newl = sorted(l)
```

```
print(newl)
```

```
print(l)
```

```
>>> [1, 5, 7, 12, 13]
```

```
>>> [1, 13, 5, 12, 7]
```





## sum()命令

### 概念讲解



sum()命令可以对列表中存储的数字求和。



用法: **sum**(列表名)

### </ 代码示例 />



```
nums = [12, 6, 2]
```

```
s = sum(nums)
```

```
print(s)
```

```
>>> 20
```



## 科学探索



### 幸存者偏差

- 当我们获取信息的渠道，仅仅来自于幸存者，没有考虑未能幸存者或者未被观察到的数据时，很可能忽略掉被筛选掉的关键信息，从而得到错误的结论。

“幸存者偏差”在我们的生活中十分常见，例如：



老师在课堂上点名：没来的同学请举手。这样上课的出勤率是100%，但实际上缺勤的同学根本不会举手。



记者在高铁上询问乘客是否买到车票，大家都说买到了，但实际上没有买到车票的人根本不在高铁上。



## 思考笔记

