# 实验4 组 合 数 据

### 一、实验目的

1．熟悉组合数据的类型。

2．掌握列表、元组、字典、集合等组合数据的创建、访问方法。

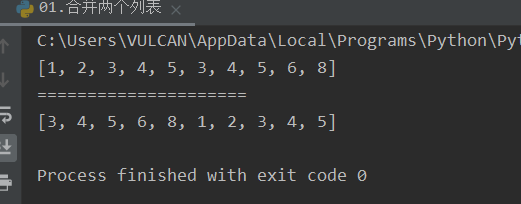
3．掌握组合数据推导式的使用方法。

4．熟悉组合数据的常见应用。

### 二、实验内容

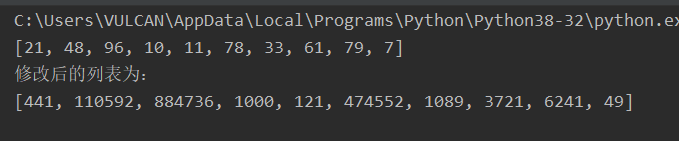
1．使用两种方法将两个列表中的数据合并。

# 1．使用两种方法将两个列表中的数据合并。  
li1 = [1, 2, 3, 4, 5]  
li2 = [3, 4, 5, 6, 8]  
# 方法一  
li3 = li1 + li2  
print(li3)  
print("=====================")  
li2.extend(li1)  
print(li2)



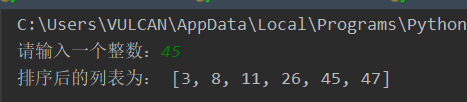
1. 列表list中包含10个1～100之间的随机整数，将列表list中的奇数变成它的平方，偶数变成它的立方。编程实现上述功能。

# 列表list中包含10个1～100之间的随机整数，将列表list中的奇数变成它的平方，偶数变成它的立方。编程实现上述功能。  
  
import random  
  
li = random.sample(range(1, 101), 10)  
li1 = []  
print(li)  
for i in li:  
 if i % 2 != 0:  
 i = i \*\* 2  
 li1.append(i)  
 else:  
 i = i \*\* 3  
 li1.append(i)  
print("修改后的列表为：")  
print(li1)



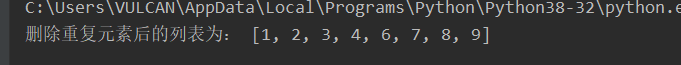
1. 列表list = [3, 8, 11, 26, 47]，从键盘输入一个新的元素，将该元素插入列表list中，保持列表list1中的元素有序。编程实现上述功能。

# 列表list = [3, 8, 11, 26, 47]，从键盘输入一个新的元素，将该元素插入列表list中，保持列表list1中的元素有序。编程实现上述功能。  
li = [3, 8, 11, 26, 47]  
a = int(input("请输入一个整数："))  
li.append(a)  
li.sort()  
print("排序后的列表为：", li)



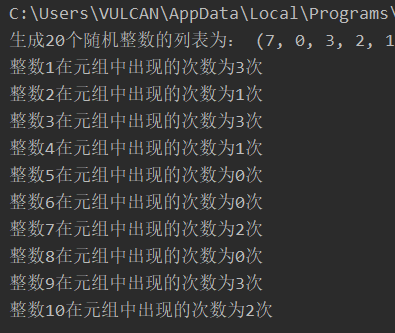
1. 编写一个程序，删除列表中的重复元素。

list = [1, 2, 4, 2, 3, 6, 7, 7, 8, 9]  
list.sort()  
list1 = []  
for i in list:  
 if i not in list1:  
 list1.append(i)  
print("删除重复元素后的列表为：", list1)



5．元组tuple中包含20个1～10之间的随机整数，统计每个整数在元组tuple中出现的次数。

import random  
  
list = []  
tuple()  
for i in range(20):  
 item = random.randint(0, 11)  
 list.append(item)  
tu = tuple(list)  
print('生成20个随机整数的列表为：', tu)  
for i in range(1, 11):  
 count = tu.count(i)  
 print("整数" + str(i) + "在元组中出现的次数为" + str(count) + "次")



6．元组grade = (68, 87, 83, 91, 93, 79, 68, 86, 66, 78)，按学号从小到大的顺序保存学生的成绩，实现如下功能：

(1)输出grade中的第2个元素。

(2)输出grade中的第3～7个元素。

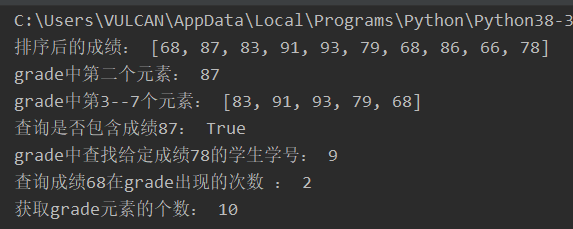
(3)使用in查询grade中是否包含成绩87。

(4)调用index()函数在grade中查找给定成绩为78的学生学号。

(5)调用count()函数查询成绩68在grade中的出现次数。

(6)使用len()函数获取grade中的元素个数。

grade=(68,87,83,91,93,79,68,86,66,78)  
list=list(grade)  
print("排序后的成绩：",list)  
print("grade中第二个元素：",list[1])  
print("grade中第3--7个元素：",list[2:7])  
print("查询是否包含成绩87：",87 in list)  
print("grade中查找给定成绩78的学生学号：",list.index(78))  
print("查询成绩68在grade出现的次数 ：",list.count(68))  
print("获取grade元素的个数：",len(list))



7．set1 = {2, 5, 9, 1, 3}，set2 = {3, 6, 8, 2, 5}，调用集合操作符或函数完成以下功能：

(1)向set1中添加一个新的元素7。

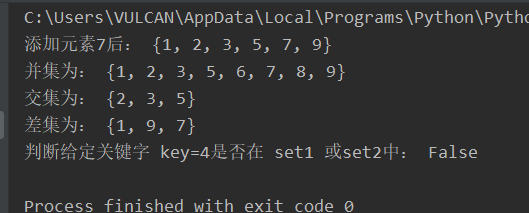
(2)求set1和set2的并集。

(3)求set1和set2的交集。

(4)求set1和set2的差集。

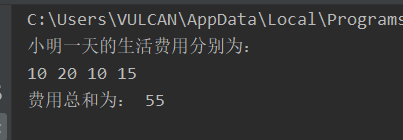
(5)判断给定关键字key = 4是否在set1或set2中。

set1 = {2, 5, 9, 1, 3}  
set2 = {3, 6, 8, 2, 5}  
set1.add(7)  
print("添加元素7后：", set1)  
set4 = set1.union(set2)  
print("并集为：", set4)  
set3 = set1.intersection(set2)  
print("交集为：", set3)  
set3 = set1.difference(set2)  
print("差集为：", set3)  
print("判断给定关键字 key=4是否在 set1 或set2中：", 4 in set4)



8．计算小明一天的生活费用，包括“早餐”费用、“中餐”费用、“晚餐”费用、“其他费用”等。给这些费用设定合理的值，并计算费用总和。

list = [10, 20, 10, 15]  
print("小明一天的生活费用分别为：")  
for i in list:  
 print(i, end=' ')  
print()  
print("费用总和为：", sum(list))



9．将某班学生的《Python程序设计》这门课程的成绩保存在字典中，学号为键(key)，分数为值(value)。实现如下功能：

(1)向字典中添加学生成绩。

(2)修改字典中指定学生成绩。

(3)删除指定学生成绩。

(4)查询指定学生成绩。

(5)统计学生成绩，如最高分、最低分、平均分等。

grade\_id=[1,2,3,4,5]  
grade\_list=[90,75,80,54,97]

grade=dict(zip(grade\_id,grade\_list))  
# {"01":80,"02":85,"03":70,"04":84,"05":97}  
grade[6]=83  
print("添加后的成绩为：",grade)  
grade[6]=87  
print("修改后的成绩为：",grade[6])  
grade.pop(6)  
print("删除学号为06的成绩：",grade)  
print("访问学号为03的成绩：",grade[3])  
print("最高分为：",max(grade\_list))  
print("最低分为：",min(grade\_list))  
len=len(grade\_list)  
sum=sum(grade\_list)  
average=sum/len  
print("平均分为：",average)

