**实验1 Python入门练习**

**姓名：宁浩 学号：2312190617 班级：软件2301 成绩：**

**1. 实验目的**

本次实验旨在动手安装Anaconda3的前提下，入门练习Python代码的开发，了解Python的各种运行方式，掌握Python3的基本数据结构与简单的控制语句(if …else、for循环等)，具备一定的数据计算与字符串处理的能力。相关知识点包括：

1) Python运行方式：

(a) 交互方式

(b) IDE环境直接运行（spyder或pycharm）

(c) Jupyter Notebook环境

(d) 命令行执行Python源文件

2) 数据结构：列表、元组、字符串、字典、集合。

3) 简单的控制结构：if…else条件语句以及条件表达式、for循环等。

根据实验练习题要求完成实验报告。注意，请在本报告中将自己的姓名、学号、班级书写正确。最后，请在规定时间内完成本次实验，并将报告的word文档以附件形式提交至学习通。(注意：不要删除题目信息，也不要将word文档转换成pdf版本)

**2. 实验练习题**

**(0)** 在1~100的正整数中，找出既能被2整除也能被3整除的所有数，并计算这些数的和值. (附答题样式，无需再次解答)

**源代码**：

s,ls=0,[]

for i in range(1,101):

if i%2==0 and i%3==0:

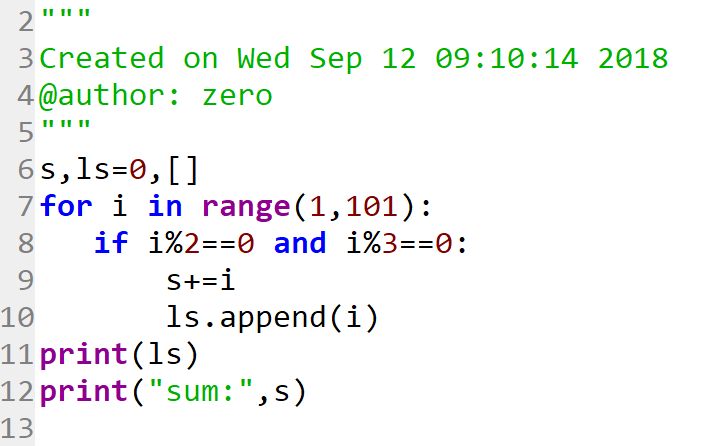
s+=i

ls.append(i)

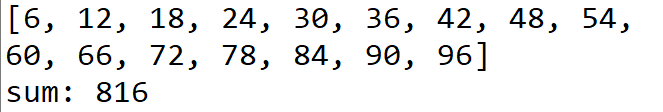
print(ls)

print("sum:",s)

**源代码截图**：



**运行结果截图：**



**(1)** 在某个Python交互式环境（cmd、Spyder或PyCharm中的IPython窗口）中展示下列三个任务的运行结果。注意需逐行敲入代码，并将运行过程的截图整体粘贴在题目下方，同时重点关注print函数的各种调用方式，以及import、help、math模块的用法，领会程序运行结果：

**(a) 任务1：**

>>> x=10

>>> y=20

>>> z=x+y

>>> print('X', '=', x)

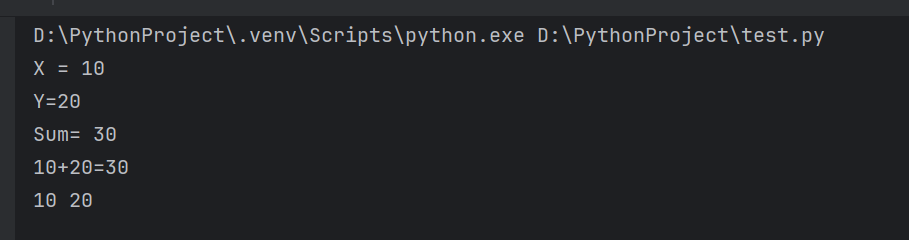
>>> print('Y', y, sep='=')

>>> print('Sum=', z)

>>> print(x, y,sep='+', end='='); print(z)

>>> print(x,y)

执行过程截图：



**(b)任务2：**

>>> import time

>>> i=time.localtime(time.time())

>>> print(time.strftime('%Y-%m-%d %H:%M:%S', i))

>>>import datetime

>>>j=datetime.datetime.now()

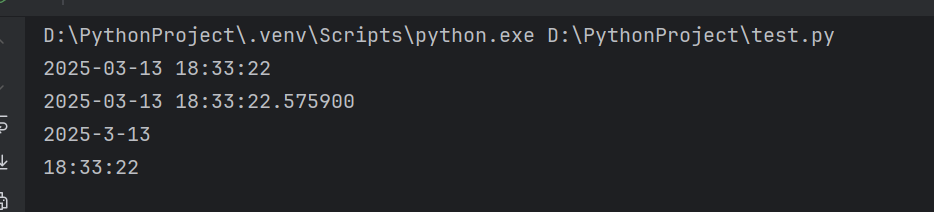
>>>j

>>>print(j)

>>>print(j.year,j.month,j.day,sep='-')

>>>print(j.hour,j.minute,j.second,sep=':')

执行过程截图：



**(c)任务3：**

>>> help(pow)

>>> pow(3,2)

>>> 3\*\*2

>>> import math

>>> help(math.pow)

>>> math.pow(3,2)

>>> help(math.log)

>>> help(math.exp)

>>> math.log(2,10)

>>> math.log(2,2)

>>> funs=[math.log,math.exp,math.sin,math.cos,math.tan]

>>> for i in range(1,6):

... print(funs[i-1](i))

...

执行过程截图：



**(2)** 字典。运行如下程序，然后截图粘贴运行结果，进而描述其功能。

s={'学号':['姓名','性别','年龄','婚否',('高数','英语','体育','软件')],

'1901':['张三','男',19,True,(90,92,98,96)],

'1902':['李四','女',16,False,(95,96,99,97)],

'1903':['王五','男',18,True,(97,91,95,98)]}

print(s)

for x in s.keys():print(x,s[x])

s['1904']=['孙六','女',20,False,None]

s['1905']=['赵七','女',22,True]

for x in s.keys():print(x,s[x])

s['1904']=['孙六','女',20,False,(99,99,99,99)]

for x in s.keys():print(x,s[x])

s.pop('1905')

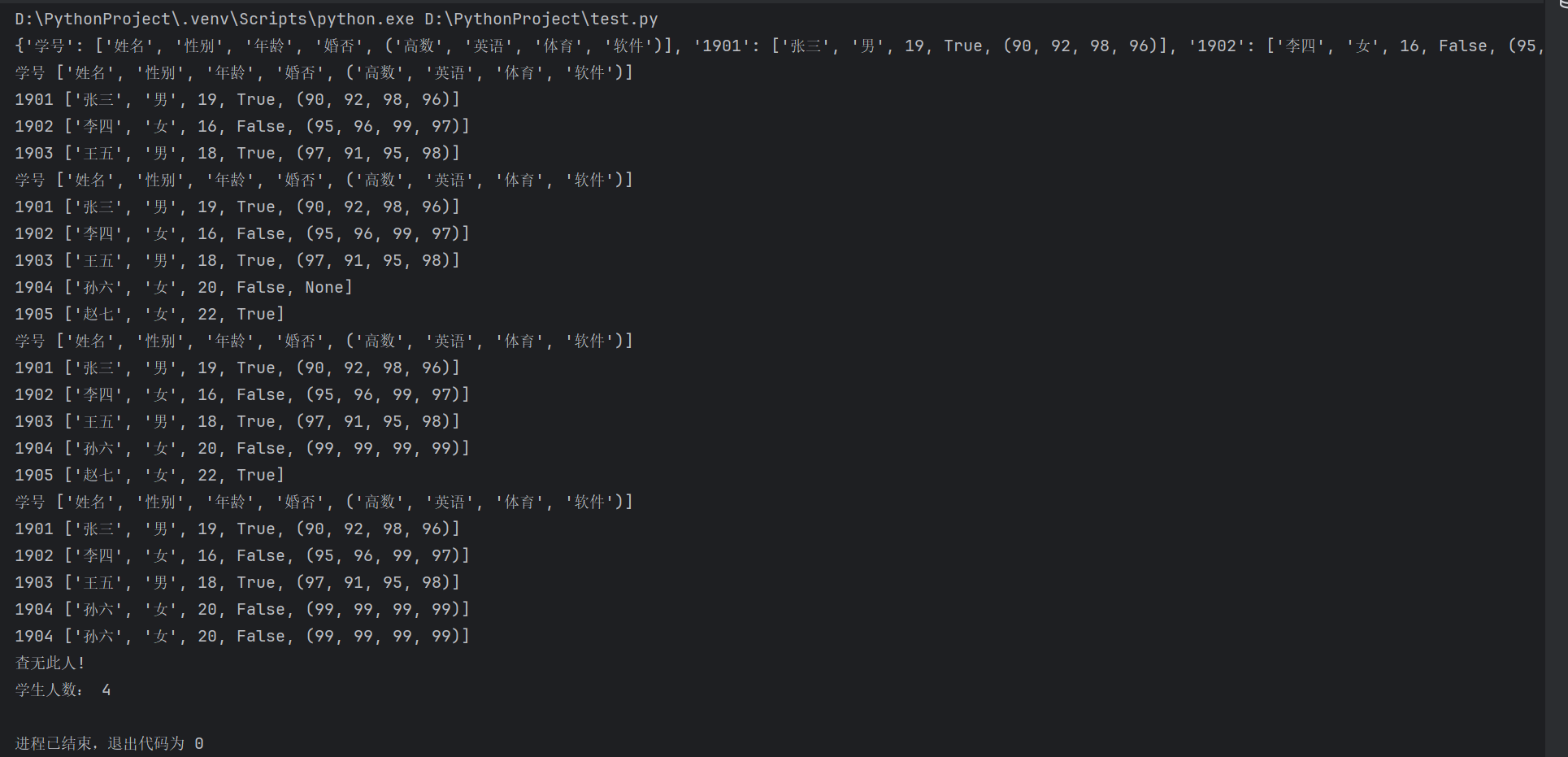
for x in s.keys():print(x,s[x])

if '1904' in s.keys():print('1904',s['1904'])

if '1905' not in s.keys():print('查无此人!')

print('学生人数：',len(s)-1)

运行结果截图：



代码功能说明：

利用字典存储学生的信息，可以直接通过赋值方式进行添加信息，通过pop进行删除信息，同时也能输入信息，查询信息。

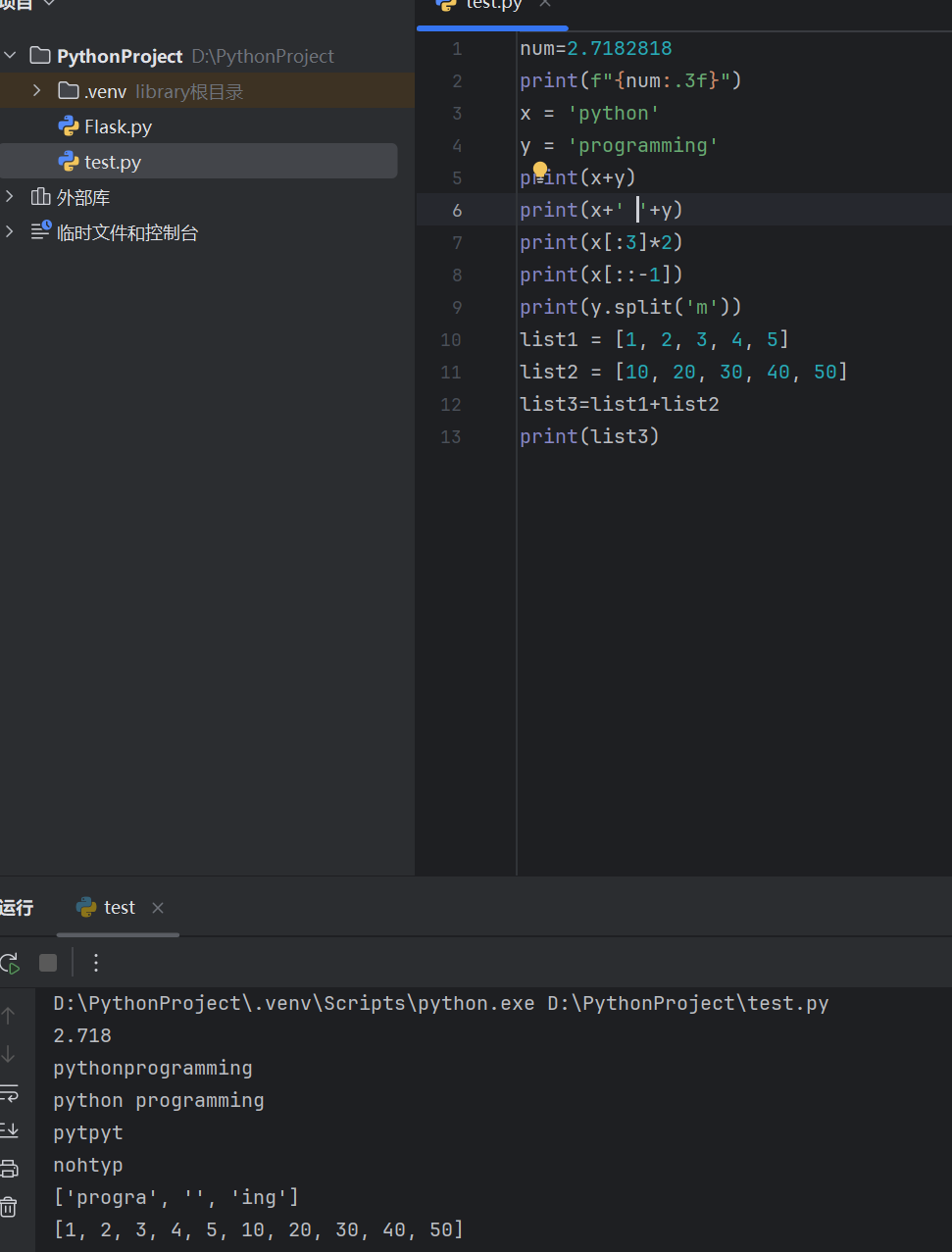
**(3)** 请在Jupyer Notebook环境中实现以下任务（将运行过程的截图粘贴在题目下方）：

(a) 变量 num = 2.7182818，将num保留小数点后3位并打印输出。

(b) 变量 x = 'python', y = 'programming'，打印输出：x + y；x + ' ' + y；x[:3] \* 2；x[::-1]；y.split('m')的结果。

(c) 给定两个列表 list1 = [1, 2, 3, 4, 5] 和 list2 = [10, 20, 30, 40, 50]，要求将list1和list2合并为一个新列表，并打印结果。

执行过程截图：



**(4)** 编写一个Python源代码，要求依次输出以下内容：

Hi, Python!

Welcome …

Happy You!

要求以两种方式执行该代码源文件，并截图粘贴相关运行结果：

**(a)**在开发环境Spyder或PyCharm中直接运行；

**(b)**在cmd环境以命令行方式执行。

**(a)**

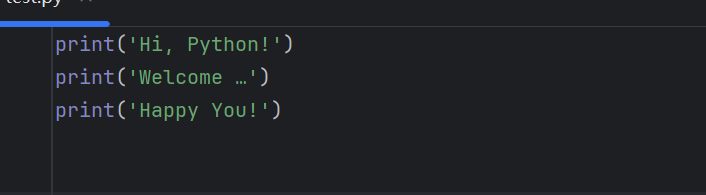
源代码：

print('Hi, Python!')

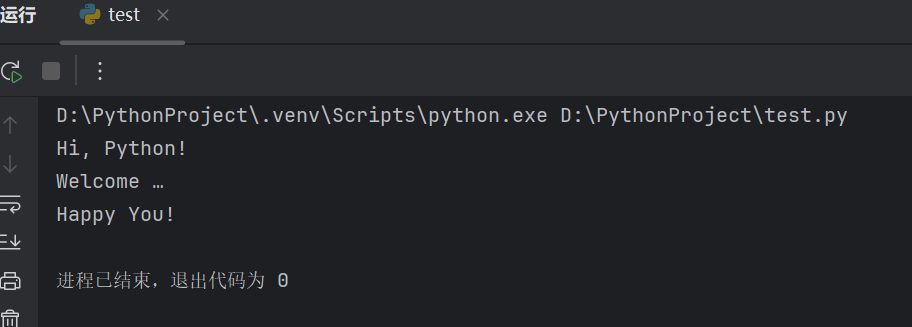
print('Welcome …')

print('Happy You!')

源代码截图：

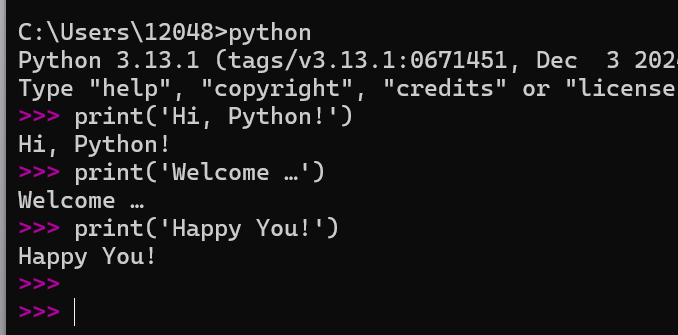


运行结果截图：



**(b)**

执行过程截图：



**(5)** 现有字符串s=”helllo12345aa\*&@\*”，分别统计英文字母、数字以及其他字符的个数，将统计结果打印输出。

源代码：

s='helllo12345aa\*&@\*'

num1,num2,num3=0,0,0

for i in s:

if 'a'<=i<='z'or 'A'<=i<='Z':

num1+=1

elif '0'<=i<='9':

num2+=1

else:

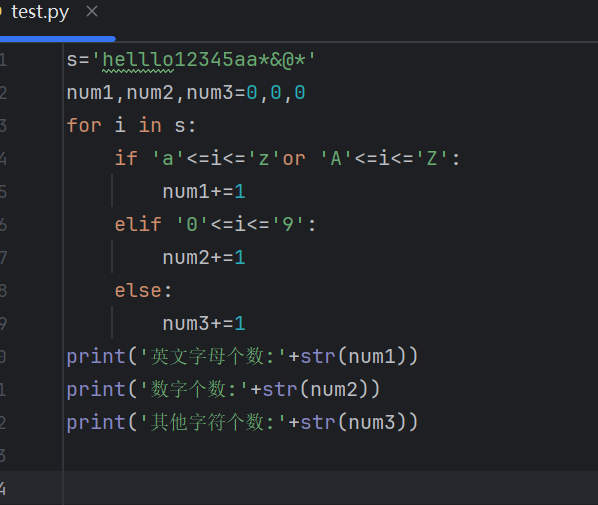
num3+=1

print('英文字母个数:'+str(num1))

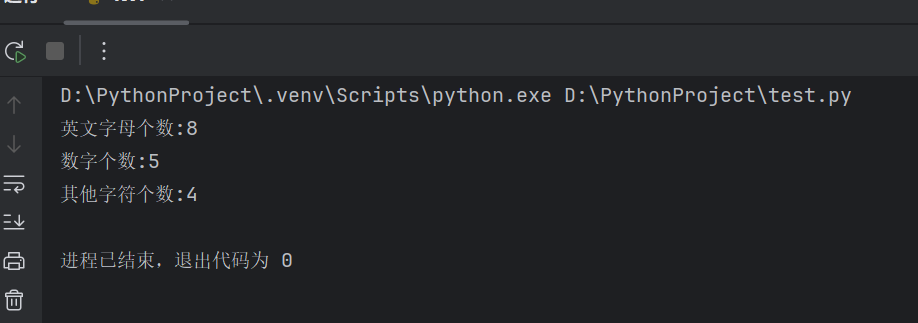
print('数字个数:'+str(num2))

print('其他字符个数:'+str(num3))

源代码截图：



运行结果截图：



**(6)** 在1~30的正整数中（包括30），找出能被2整除，但不能被4整除的所有数，并打印输出这些数以及这些数的乘积。

源代码：

list1=[]

for i in range(1,31):

if i % 2 == 0 and not i % 4 == 0:

list1.append(i)

mut=1

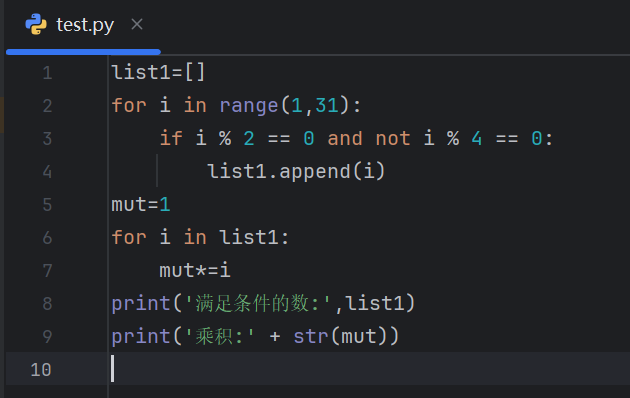
for i in list1:

mut\*=i

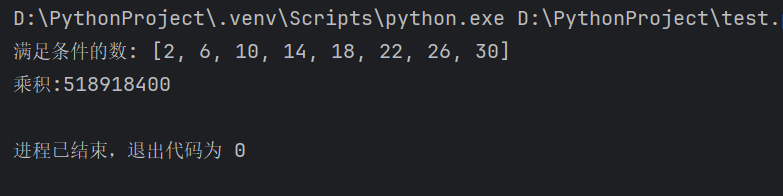
print('满足条件的数:',list1)

print('乘积:' + str(mut))

源代码截图：



运行结果截图：



**(7)** 若一个数能表示成某个整数的[平方](https://baike.baidu.com/item/%E5%B9%B3%E6%96%B9/619690" \t "_blank)形式，则称这个数为完全平方数. 请找出1000000以内的正整数x，使得x+100和x+168都是完全平方数。

源代码：

import math

list1=[]

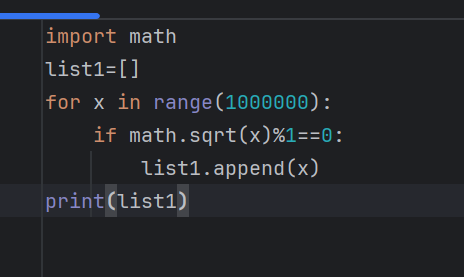
for x in range(1000000):

if math.sqrt(x)%1==0:

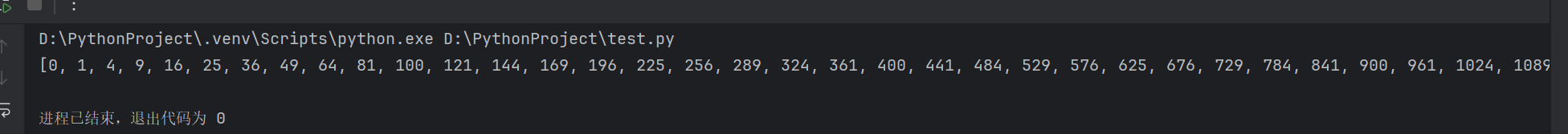
list1.append(x)

print(list1)

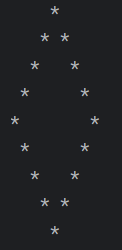
源代码截图：



运行结果截图：



**(8)** 请编写python程序，输入一个正整数n，打印出n行的空心菱形，例如当n = 5时，输出如下：



源代码：

n=int(input())

print('请输入整数n:')

for i in range(n):

for j in range(n-i-1):

print(' ',end='')

print('\*',end='')

for j in range(2\*i-1):

print(' ',end='')

if i>0:

print('\*')

else:

print()

for i in range(n-2,-1,-1):

for j in range(n-i-1):

print(' ', end='')

print('\*', end='')

for j in range(2 \* i - 1):

print(' ', end='')

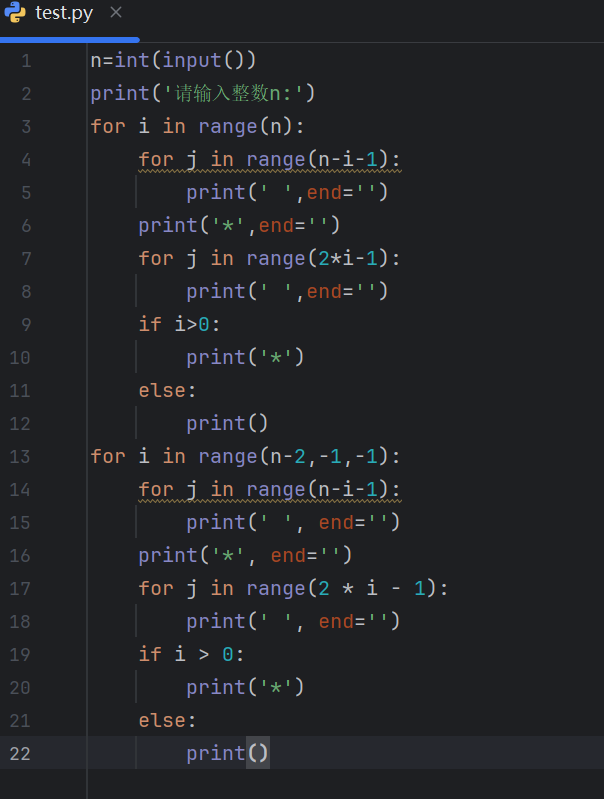
if i > 0:

print('\*')

else:

print()

源代码截图：



运行结果截图：



**(9)** 输入一个正整数 （），当自然数依次取时，算式有多少个不同的值？（注：为取整函数，表示不超过的最大自然数，即的整数部分。）

源代码：

import math

print('请输入一个正整数N:')

N=int(input())

list1=[]

for n in range(1,N+1):

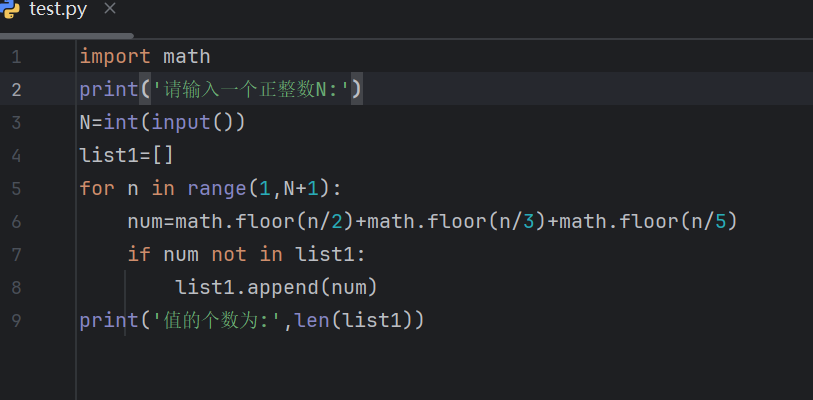
num=math.floor(n/2)+math.floor(n/3)+math.floor(n/5)

if num not in list1:

list1.append(num)

print('值的个数为:',len(list1))

源代码截图：



运行结果截图：



**(10)** 编写python源代码，求一个3\*3矩阵主对角线元素之和。

矩阵为：[[1,2,3],[4,5,6],[7,8,9]]

源代码：

list1=[[1,2,3],[4,5,6],[7,8,9]]

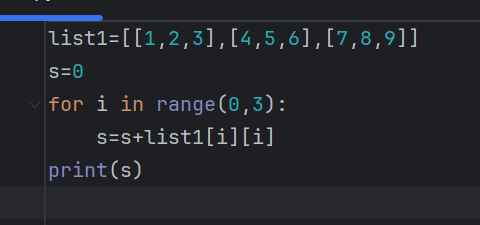
s=0

for i in range(0,3):

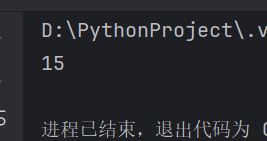
s=s+list1[i][i]

print(s)

源代码截图：



运行结果截图：



**(11)** 第行给出一个正整数，第行给出个非负整数，以空格分隔。请统计这个非负整数中奇数和偶数分别有多少个？

源代码：

print("输入一个正整数N")

N=int(input())

print("给出N个非负整数，以空格分隔")

List= input().split(' ')

sum1,sum2 = 0,0

for i in range(N):

if int(List[i])%2==0:

sum2+=1

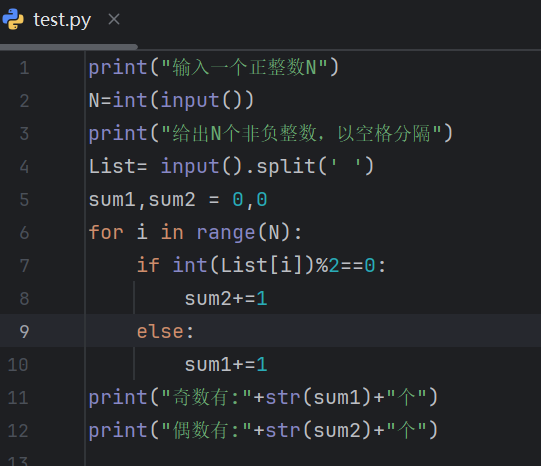
else:

sum1+=1

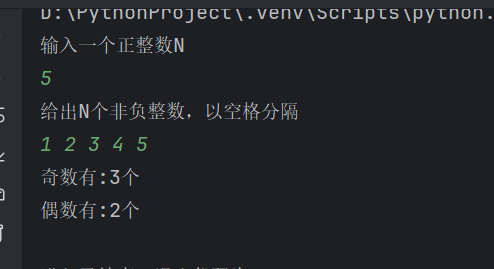
print("奇数有:"+str(sum1)+"个")

print("偶数有:"+str(sum2)+"个")

源代码截图：



运行结果截图：



**(12)** 创建一个字典student\_scores，包含以下学生的成绩：

'Alice': 85

'Bob': 90

'Charlie': 78

'David': 92

'Eve': 88

请打印输出所有学生的姓名和成绩；找出成绩最高的学生，并打印输出其姓名和成绩。

源代码：

student\_scores={'Alice':85,'Bob':90,'Charlie':78,'David':92,'Eve':88}

max\_scores,name=0,''

for i in student\_scores.keys():

print(i+':'+str(student\_scores[i]))

if student\_scores[i]>max\_scores:

max\_scores=student\_scores[i]

name=i

print('成绩最高的学生是:' + name +',他的成绩为:' + str(max\_scores))

源代码截图：



运行结果截图：

