**实验4：Python函数与代码复用**

**一、实验目的：**

（1）掌握函数的定义和基本使用方法；

（2）掌握函数的位置参数、默认值参数、关键参数、可变参数等参数类型的使用；

（3）掌握利用lambda表达式建立匿名函数的方法；

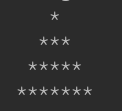
**二、实验仪器及实验环境**

（1）硬件：PC机；

（2）软件：Anaconda Jupyter Notebook，Spyder

**三、实验内容：**

**1、编写一个函数，接收一个表示行数的参数n（比如4），输出下面三种图形。**



\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

---------------------

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

提示：

（1）图形行数为n

**（2）每行字符包含两部分：空格和\*，注意查找个数和行号的关系规律。**

图1参考代码：

def print\_star(n):

for i in range(n):

print(" "\*(n-i),"\*"\*(2\*i+1))

print\_star(4)

尝试写出图2和图3的代码

**2、编写一个函数，计算n的阶乘，并依次输出1~20的阶乘。**

参考代码：

def facci(n):

s=1

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

s = s\*i

return s

for i in range(1,21):

print(facci(i),end=" ")

**3.定义两个函数，分别用于将小写字母转换为大写和将大写字母转换为小写。**

def to\_lower(s):

st=list(s)

for i in range(len(st)):

if st[i].islower():

st[i]=st[i].upper()

return "".join(st)

def to\_upper(s):

st=list(s)

for i in range(len(st)):

if st[i].isupper():

st[i]=st[i].lower()

return "".join(st)

s="chINA"

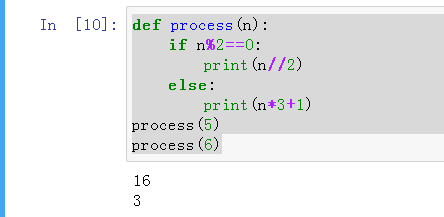
s=to\_lower(s)

print(s)

s=to\_upper(s)

print(s)

**4、编写一个名为process的函数，它接收一个参数n，如果n为偶数则process函数输出n//2，如果n为奇数，则process打印出n\*3+1。然后编写函数调用语句，测试当n=5和6时，函数的输出情况，以检查函数的功能是否正确。**



**5、编写一个函数calfilter，该函数可以接收任意个数的参数，返回一个元组，元组的第一个值为所有参数的平均值（保留两位小数），第二个值是元素个数，从第三个开始便是大于平均值的所有数。**

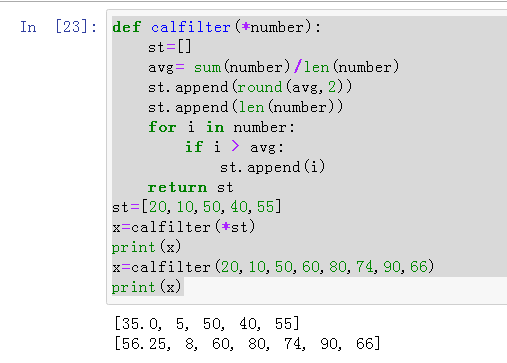
提示：round函数可以对浮点型数据按照小数点后位数做四舍五入。格式为：

round( x [, n=0] )

其中：x是要进行四舍五入的浮点数

n表示保留的小数点后位数，默认值为0。

运行截图：



**6、将实验3中统计词频题目生成的字典转换为列表后进行排序，输出频率最高的5个单词。**

提示：

（1）统计出的各单词的频率数据可以存储到一个字典中，也可以把每个单词和频率数据组成一个元组，形成一个元组列表来存储所有单词的出现次数。

（2）如果想查找频率最高的单词，则需要对词频结果排序。如果采用字典来存储数据时，由于字典是不能排序的，如果需要排序，则必须将字典的items转换为列表，转换后列表中的每一个元素都是一个元组，格式为：（”after”,2）。

Sorted函数的功能是从可迭代对象中的项目返回新的排序列表。语法格式为：

sorted(iterable[, key][, reverse])

其中：

iterable是需要进行排序的迭代对象

key用于指定一个函数参数，用于从每个列表元素中提取比较键：比如key=str.lower，则可以实现将字符串都转换为小写后进行比较，默认值为 None表示直接比较元素。

reverse 是一个布尔值。如果设置为 True，那么列表元素将按照每个比较反转进行排序。

本题目中对词频项目依据频率进行排序，则需要根据每个元组对象的第二项来进行比较。因此需要编写一个函数，返回元组第二项作为比较键。

def k(x):

return x[1]

然后排序时使用这个函数作为key参数值，即：

stdic=sorted(stdic,key=k,reverse=True)

将词频列表按照单词出现次数降序排列，这样输出前5项就是出现次数最多的单词及其词频了。

strstory='''Once upon a time, there lived a monkey in the woods. The monkey climbed up the tree and looked down at the river everyday.

One day, the monkey saw fishermen throw a net over the river. He watcbbhed how the fishermen caught fish.

Sometime after, the fishermen left the net and went to have lunch. Then, the monkey climbed down from the tree and went to the riverside. The monkey was planning to throw the net like the fishermen did.

The monkey was confident since he was good at imitating. Yet, the net wound around the monkey when he touched it. Tied up in the net, the monkey fell into the water.

The monkey regretted his behavior, but it was too late. "I should have learned how to use the net before touching it. After believing my prowess and showing off, I am in a bad fix." The monkey who did not behave carefully drowned in the water.

'''

for i in ".,'\"?\n":

strstory=strstory.replace(i," ")

strstory = strstory.lower()

words=strstory.split()

#print(words)

words2=set(words)

dic={}

for i in words2:

dic[i]=words.count(i)

stdic=list(dic.items())

def k(x):

return x[1]

stdic=sorted(stdic,key=k,reverse=True) #与使用key=lambda x:x[1]效果相同

print(stdic[0:5])

输出结果为：



**7、利用lambda表达式实现上题中k函数的功能。**

提示：lambda表达式的格式为：

lambda 参数列表：表达式

stdic=sorted(st,key=lambda x:x[1],reverse=True)

print(stdic[0:5])

**五、实验结果与分析：**

（1）验证实现的程序功能与题目要求是否一致。

（2）查找程序出错的语句代码，调试分析出错的原因并改进语句。