# python题库

为了方便python学习组同学的训练，现在需要组建一个python题库对相应的python知识点进行训练。

# 1、输入输出

已知以下程序段，要想输出结果为 1,2,3，应该使用的表达式是：

x = [1,2,3]

z = []

for y in x:

z.append(str(y))

A. print(z)

B. print(“,”.join(x))

C. print(x)

D. print(“,”.join(z))

正确答案： D.

设 str = ‘python’，想把字符串的第一个字母大写，其他字母还是小写，正确的选项是：

A. print(str[0].upper()+str[1:])

B. print(str[1].upper()+str[-1:1])

C. print(str[0].upper()+str[1:-1])

D. print(str[1].upper()+str[2:])

正确答案： A.

给出如下代码：

TempStr = “Hello World”

以下选项中可以输出“World”子串的是

A. print(TempStr[–5: –1])

B. print(TempStr[–5:0])

C. print(TempStr[–4: –1])print(TempStr[–4: –1])

D. print(TempStr[–5:])

正确答案： D.

假设将单词保存在变量 word 中，使用一个字典类型 counts={}，统计单词出现的次数可采用以下代码：

A. counts[word] = count[word] + 1

B. counts[word] = 1

C. counts[word] = count.get(word,1) + 1

D. counts[word] = count.get(word,0) + 1

正确答案： D.

下面代码的执行结果是：

ls = [“2020”, “20.20”, “Python”]

ls.append(2020)

ls.append([2020, “2020”])print(ls)

A. [‘2020’, ‘20.20’, ‘Python’, 2020]

B. [‘2020’, ‘20.20’, ‘Python’, 2020, [2020, ‘2020’]]

C. [‘2020’, ‘20.20’, ‘Python’, 2020, [‘2020’]]

D. [‘2020’, ‘20.20’, ‘Python’, 2020, 2020, ‘2020’]

正确答案： B.

以下程序的输出结果是：

dat=[‘1’, ‘2’, ‘3’, ‘0’, ‘0’, ‘0’]

for item in dat:

if item == ‘0’:

dat.remove(item)

print(dat)

A. [‘1’, ‘2’, ‘3’]

B. [‘1’, ‘2’, ‘3’, ‘0’, ‘0’]

C. [‘1’, ‘2’, ‘3’, ‘0’, ‘0’, ‘0’]

D. [‘1’, ‘2’, ‘3’, ‘0’]

正确答案： D.

给出如下代码：

s = list(“巴老爷有八十八棵芭蕉树，来了八十八个把式要在巴老爷八十八棵芭蕉树下住。

老爷拔了八十八棵芭蕉树，不让八十八个把式在八十八棵芭蕉树下住。八十八个把式

烧了八十八棵芭蕉树，巴老爷在八十八棵树边哭。”)

以下选项中能输出字符“八”出现次数的是

A. print(s.index(“八”))

B. print(s.index(“八”),6)

C. print(s.index(“八”),6,len(s))

D. print(s.count(“八”))

正确答案： D.

以下关于 random 库的描述，正确的是：

A. 设定相同种子，每次调用随机函数生成的随机数不相同

B. 通过 from random import \* 引入 random 随机库的部分函数

C. uniform(0,1) 与 uniform(0.0,1.0) 的输出结果不同，前者输出随机整数，后者输出随机小数

D. randint(a,b) 是生成一个 [a,b] 之间的整数

正确答案： D.

以下关于 Python 函数对变量的作用，错误的是：

A. 简单数据类型在函数内部用global保留字声明后，函数退出后该变量保留

B. 全局变量指在函数之外定义的变量，在程序执行全过程有效

C. 简单数据类型变量仅在函数内部创建和使用，函数退出后变量被释放

D. 对于组合数据类型的全局变量，如果在函数内部没有被真实创建的同名变量，则函数内部不可以直接使用并修改全局变量的值

正确答案： D.

给出如下代码

import random as ran

listV = []

ran.seed(100)for i in range(10):

i = ran.randint(100,999)

listV.append(i)

以下选项中能输出随机列表元素最大值的是

A. print(listV.max())

B. print(listV.pop(i))

C. print(max(listV))

D. print(listV.reverse(i))

正确答案： C.

以下程序的输出结果是：

for i in range(3):

for s in “abcd”:

if s==“c”:

break

print (s,end="")

A. abcabcabc

B. aaabbbccc

C. aaabbb

D. ababab

正确答案： D

s = s.lower()

for ch in ‘’,?.: ()’:

s = s.replace(ch," ")

words = s.split()

counts = {}

for word in words:

counts[word] = counts.get(word,0)+1

items = list(counts.items())

items.sort(key=lambda x:x[1],reverse = True)

fo = open(“wordnum.txt”,“w”,encoding =“utf-8”)

for i in range(10):

word,count = items[i]

fo.writelines( word + “:” + str(count) + “\n”)

fo.close()

A. 统计字符串 s 中所有单词的出现次数，将单词和次数写入 wordnum.txt 文件

B. 统计字符串 s 中所有字母的出现次数，将单词和次数写入wordnum.txt 文件

C. 统计输出字符串 s 中前10个字母的出现次数，将单词和次数写入 wordnum.txt 文件

D. 统计字符串 s 中前10个高频单词的出现次数，将单词和次数写入 wordnum.txt 文件

正确答案： D.

# 2、条件判断，控制结构

给定一个整数n，通过使用Python编程，求出1到n之间所有的数字，判断其是否为质数，打印出所有质数。

要求：

1. 使用for循环

2. 使用if-else条件判断

详细代码：

# 用Python编程求出1到n之间所有质数

n = int(input("请输入一个正整数："))

# 对输入的n进行判断，若n小于2，则直接输出"无质数"

if n < 2:

print("无质数")

# 使用for循环遍历1到n之间的每一个数字

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

# 使用if-else条件判断，判断当前数字是否为质数

is\_prime = True

for j in range(2, i):

if i % j == 0:

is\_prime = False

break

# 输出所有质数

if is\_prime:

print(I,end=’ ’)

输入1-26的数字出现对应索引的英文字母

如： 输入1，输出大写字母A

详细代码：

s = eval(input("请输入一个数字："))

ls = [0]

for i in range(65,91):

ls.append(chr(i))

print("输出大写字母：{}".format(ls[s]))

注意点 1. 1-A，2-B，所以ls一开始需要填充0位置

2. chr()中65-90对应A-Z

# 3、序列，list, str, dict

题目：

给定一个字典，字典的键为字符串，值为列表，列表的元素为字典。请编写一个函数，按照字典中每个列表元素的年龄，创建一个新的字典，新字典的键为每个列表元素的年龄，值为每个列表元素的其他属性。

例如：

‘A’: [{‘name’:’Jason’, ‘age’: 20},

{‘name’:’Brooke’, ‘age’: 30}],

‘B’: [{‘name’:’Nina’, ‘age’: 45},

{‘name’:’John’, ‘age’: 25}]}

输出：

{20: [{'name': 'Jason'}], 30: [{'name': 'Brooke'}], 45: [{'name': 'Nina'}], 25: [{'name': 'John'}]}

详细代码：

def create\_new\_dict(d):

new\_dict = {}

for key, lst in d.items():

for item in lst:

age = item['age']

del item['age']

if age not in new\_dict:

new\_dict[age] = [item]

else:

new\_dict[age].append(item)

return new\_dict

print(create\_new\_dict({'A': [{'name':'Jason', 'age': 20},

{'name':'Brooke', 'age': 30}],

'B': [{'name':'Nina', 'age': 45}, {'name':'John', 'age': 25}]}))

#输出结果：

# {20: [{'name': 'Jason'}], 30: [{'name': 'Brooke'}], 45: [{'name': 'Nina'}], 25: [{'name': 'John'}]}

题目：

给定一个字典，其键值为整数，值为字符串，以及一个列表，其元素为元组，编写一个函数，将字典中的值按照列表中的元组的第一个数字进行排序，元组的第二个数字表示目标值，结果为一个新的字典。

例如：

给定字典：

dic = {1: 'a', 2: 'b', 3: 'c', 4: 'd'}

列表：

lst = [(2, 'z'), (1, 'y'), (3, 'x')]

输出结果：

{2: 'z', 1: 'y', 3: 'x', 4: 'd'}

详细代码：

def sortDic(dic, lst):

# 创建一个长度为字典长度的列表

res = [''] \* len(dic)

# 按照元组中的第一个数字，从字典中获取值，并赋值给结果列表

for tup in lst:

num = tup[0]

res[num - 1] = tup[1]

# 遍历字典，如果结果列表中的值为空，就从字典中获取值，并赋值给结果列表

for key, value in dic.items():

if res[key - 1] == '':

res[key - 1] = value

# 将结果列表转换为字典，并返回

return dict(zip(dic.keys(), res))

dic = {1: 'a', 2: 'b', 3: 'c', 4: 'd'}

lst = [(2, 'z'), (1, 'y'), (3, 'x')]

print(sortDic(dic, lst)) # {2: 'z', 1: 'y', 3: 'x', 4: 'd'}

# 4、numpy, pandas, matplotlib, 可以使用一个数据文件，要求学生进行分析和绘图，综合考察对这三个库的使用。

请你根据以下数据绘制中国历届奥运奖牌的折线图，要求使用matplotlib，将金银铜三种奖牌分别用三种不同的颜色表示，必须画在同一张图中。

金 15, 5, 16, 16, 28, 32, 51, 38, 26, 38

银 8, 11, 22, 22, 16, 17, 21, 27, 18, 32

铜 9, 12, 16, 12, 15, 14, 28, 23, 26, 18

详细代码：

import matplotlib

import matplotlib.pyplot as plt

# 设置字体为微软雅黑，解决中文显示问题

matplotlib.rc("font", family='Microsoft YaHei')

# 1.准备数据

gold\_medal = [15, 5, 16, 16, 28, 32, 51, 38, 26, 38]

silver\_medal = [8, 11, 22, 22, 16, 17, 21, 27, 18, 32]

bronze\_medal = [9, 12, 16, 12, 15, 14, 28, 23, 26, 18]

# 索引列表

year = [1984, 1988, 1992, 1996, 2000, 2004, 2008, 2012, 2016, 2021]

"""

设置刻度：

- xticks()：设置x轴刻度

- yticks()：设置y轴刻度

"""

# 设置x轴刻度

plt.xticks(year)

# 设置线条样式

line\_gold, = plt.plot(year, gold\_medal, '-\*', label='gold\_medal')

line\_silver, = plt.plot(year, silver\_medal, '-o', label='silver\_medal')

line\_bronze, = plt.plot(year, bronze\_medal, '-^', label='bronze\_medal')

# 设置图例

plt.legend(handles=[line\_gold, line\_silver, line\_bronze], loc='best')

# 设置标题及字体大小

plt.title('中国历届奥运会奖牌情况', fontsize=20)

plt.xlabel('年份', fontsize=14)

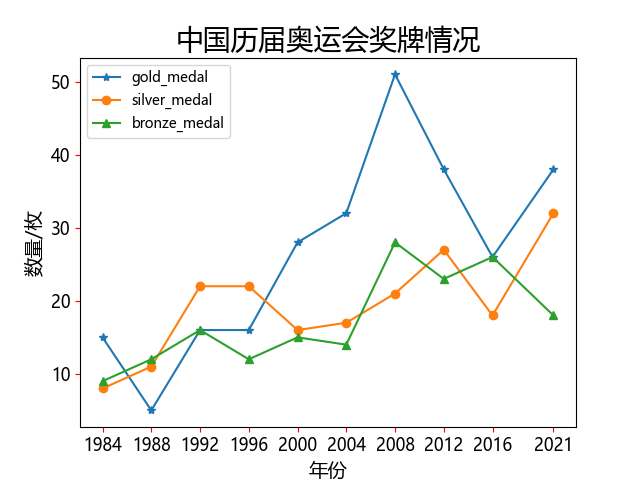
plt.ylabel('数量/枚', fontsize=14)

# 设置坐标轴刻度

plt.tick\_params(axis='both', labelsize=12, color='red')

# 显示图表

plt.show()



老王的血压有些高，医生让家属给老王测血压。老王的女儿记录了一段时间的血压测量值，在文件 xueyajilu.txt 中，内容示例如下：

2018/7/2 6:00,140,82,136,90,69

2018/7/2 15:28,154,88,155,85,63

2018/7/3 6:30,131,82,139,74,61

2018/7/3 16:49,145,84,139,85,73

2018/7/4 5:03,152,87,131,85,63

文件内各部分含义如下：

测量时间,左臂高压,左臂低压,右臂高压,右臂低压,心率

参照代码模板完善代码，实现下述功能，可以修改给定代码。

（1）使用字典和列表类型进行数据分析，获取老王的

1.左臂和右臂的血压平均值

2.左臂和右臂的高压最高值、低压最高值

3.左臂和右臂的高/低压差平均值

4.心率的平均值

给出左臂和右臂血压情况的对比表，输出到屏幕上，请注意每行三列对齐。

示例格式如下（不代表最终答案）：

对比项 左臂 右臂

高压最大值 163 155

低压最大值 93 90

压差平均值 61 57

高压平均值 146 140

低压平均值 85 82

（2）上述显示的五个项目，如果左臂有大于 50% 的项目高于右臂，则输出“结论：左臂血压偏高”；如果等于 50% 的项目高于右臂，则输出“结论：左臂血压与右臂血压相当”；如果小于 50% 的项目高于右臂，则输出“结论：右臂血压偏高”。

示例如下：

结论：左臂血压偏高, 心率的平均值为66

(注意：此处为格式示例，实际数据与此不同）

详细代码：

fi = open(r'C:\Users\lenovo\Desktop\xueya.txt','r',encoding='utf-8')

jl = [[],[],[],[],[]] # 1:zb\_h, zb\_l,yb\_h,yb\_l

zyc = []

yyc = []

xl = []

for l in fi:

lls=l.replace('\n','').strip().split(',')

for i in range(1,5):

jl[i].append(eval(lls[i]))

zyc.append(eval(lls[1])-eval(lls[2]))

yyc.append(eval(lls[3])-eval(lls[4]))

xl.append(eval(lls[5]))

fi.close()

cnt = len(xl)

res = []

res.append(list(("高压最大值",max(jl[1]),max(jl[3]))))

res.append(list(("低压最大值",max(jl[2]),max(jl[4]))))

res.append(list(("压差平均值",sum(zyc)//cnt,sum(yyc)//cnt)))

res.append(list(("高压平均值",sum(jl[1])//cnt,sum(jl[3])//cnt)))

res.append(list(("低压平均值",sum(jl[2])//cnt,sum(jl[4])//cnt)))

res.append(list(("心率平均值",sum(xl)//cnt,0)))

zbg = 0

ybg = 0

print('{:<10}{:<10}{:<10}'.format("对比项", "左臂", "右臂"))

for r in range(len(res)-1):

print('{:<10}{:<10}{:<10}'.format(res[r][0],res[r][1],res[r][2]))

if res[r][1]> res[r][2]:

zbg += 1

else:

ybg += 1

if zbg > ybg: print('结论：左臂血压偏高',end ='')

elif zbg == ybg: print('结论：左臂血压与右臂血压相当',end ='')

else: print('结论：右臂血压偏高',end ='')

print(', 心率的平均值为{}'.format(res[5][1]))

要点：

1.这道题的关键问题是解决对行数据按照列的方式来处理

2.参考答案给出的是一般思路，把文件里读出来的数据分别按列的方式，拷贝到一维列表中去，对一维列表进行各种max，min，sum，len操作

4.数据处理的逻辑并不复杂，这里不赘述。

5.其次要解决的问题，是当列数有些多的时候，程序看起来很冗余，需要做一些优化合并；

6.答案里给出的方案是前四个计算公式类似的，放到一个二维列表里，加个循环结构，就可以把四条语句缩成一条。

7.第三个问题，显示输出要求像表格一样输出，又要比较高效率，就引入了一个二维列表，把要输出的行头和内容，变成列表，再加到二维列表中去

8.最后的显示就变得很容易，一个循环就搞定了。

9.最后的问题，是要对输出的二维表格数据作统计、比对、下结论。比较好的方法是输出的时候，顺便做计数；

10.显示输出结论的时候，遇到一个小问题，要在做比对后的不同的结论后面，追加一条相同的心率的平均值，还不能换行；解决的方法就是用end=‘’，然后在最后加一句print心率的结果。当然可以在每条结论的后面，都重复的写一句心率的结果，虽然最后的结果是一样的，但代码有冗余。