1在Python中定义函数时不需要声明函数参数的类型。 T

2 函数一定要有返回值。 F

3 使用函数的一个重要原因是为了减少代码重复。 T

4 有函数如下，如果a = 1, b = 2，然后调用foo(a,b)，在函数外a,b的值依然为1,2

**def** **foo**(a,b):

a, b = 3,4 **T**

**5** Pyhon定义函数时，最后必须要有Return关键字，否则会出错。 F

6 调用带有默认值参数的函数时，不能为默认值参数传递任何值，必须使用函数定义时设置的默认值。 F

7 在Python 中，函数需要有且仅有一个返回值。 F

8 eval函数的作用是将传入的字符串转化为数值。 F

9 程序中如果定义了函数，则函数内部使用的变量名不能与程序其它地方的变量名同名。 F

10 执行如下代码后，ls内的值是[0,1,3]

ls = [1,0,3]

**sorted**(ls) F

11 如果定义两个函数时不小心使用了相同的名称，则第二个函数会覆盖第一个函数。T

12 定义一个函数后，不能在该函数内部再定义新的函数。 F

13 某程序定义了一个函数fx()，若要让变量y获得函数返回值，则可以使用代码y=fx()或y=fx。 F

14 程序设计中，函数可以先调用再定义，也可以先定义再调用。 F

15 每个函数定义或调用时都必须要有一个名字。 F

16 主程序中定义的变量不能函数内部使用，只能通过参数方式传递给函数内部变量进行使用。 F

17 定义某函数fx(x,\*p,\*\*q)中，\*p是指参数个数不固定的参数，\*\*q是指参数为字典格式的参数。 T

18 用户在主程序中定义了两个以上的函数时，函数之间不能相互调用对方，只能由主程序调用。 F

19 使用函数的一个重要原因是为了减少代码重复。 T

20 在Python 中，函数需要有且仅有一个返回值。 F

21 程序中如果定义了函数，则函数内部使用的变量名不能与程序其它地方的变量名同名。 F

22 调用函数时，如果函数格式是：Funx(a=1,b=2)，则该函数的参数a和b是关键参数的调用方式。 F

23 Python定义的函数如果有参数，则调用函数时，实参和形参必须保持一致。 F

24 调用带有默认值参数的函数时，不能为默认值参数传递任何值，必须使用函数定义时设置的默认值。 F

25 Pyhon定义函数时，最后必须要有Return关键字，否则会出错。 F

26 在定义函数时，某个参数名字前面带有两个\\*符号表示可变长度参数，可以接收任意多个关键参数并将其存放于一个字典之中。 T

27 函数都有返回值，如果函数没有return语句，则会返回一个特殊值：None。 T

28 在Python中定义函数时不需要声明函数参数的类型。 T

11Python内置函数( )可以返回列表、元组、字典、集合、字符串以及range对象中元素个数。

A.object()

B.type()

C.sum()

D.len()

12, 函数-程序代码为

def cube(x):

answer = x \* x \* x

**return** answer

answer=4

print(answer,cube(3))

其结果为（ ）

A.27 27

B.4 4

C.4 3

D.4 27

13 运行表达式eval("5/2+5%2+5//2")的结果是（ ）。

A.7.5

B.5/2+5%2+5//2

C.5.5

D.5

下面对函数调用时的关键字参数描述错误的是（ ）。

A.调用的关键字参数有具体的名字和赋值如:fx(x=2)

B.关键字参数的值无法通过实参形式传递

C.多个关键字参数之间的顺序可以交换

D.调用函数时，多个参数可以只对其中某个变量采用关键值参数格式赋值

下面代码运行结果是（ ）。

def judge(para,\*\*para2):

print(type(para2),end='')

print(para2)

judge(1,a=2,b=3,c=4,d=5)

A.<class 'int'> a=2,b=3,c=4,d=5

B.<class 'dict'> a=2,b=3,c=4,d=5

C.<class 'int'> {'a': 2, 'b': 3, 'c': 4, 'd': 5}

D.<class 'dict'> {'a': 2, 'b': 3, 'c': 4, 'd': 5}

Python函数内，若要定义一个变量为全局变量，应对该变量使用关键字（ ）。

A.all

B.local

C.keywords

D.global

下面选项中描述的函数参数特点一定是字典的是（ ）。

(1分)

A.f(1,2,c=3)

B.f(c=2,a=1)

C.def use(\*\*t):

D.def (\*t):

下面函数定义错误的是（ ）。

A.def fux(x=1,y=1):

B.def fux(x=1,y):

C.def fux(x,y):

D.def fux(x,y=1):

Python内置函数( )可以返回列表、元组、字典、集合、字符串以及range对象中元素个数。

A.sum()

B.type()

C.len()

D.object()

字符串的‘转义’是指某些字母与特殊符号放一起，具有字母以外的系统特定功能，下面具有“转义”的特殊符号是（ ）。

A." \ \ \ "

B."//"

C." \ "

D."/"

14 编写函数实现以下功能： 输入三角形的三条边，先判断是否可以构成三角形，满足条件则输出三角形的面积，否则就报错； 运行及输出要求请参考样例。 （提示：1.三角形需要满足，均大于0且任意两条边之和大于第三边，2.面积s=sqrt（h\*(h-a)\*(h-b)\*(h-c)），其中h=(a+b+c)/2)。

import math

 #def sanjiao(a,b,c):

if ((a+b)>c) and ((a+c)>b) and ((c+b)>a):

h=(a+b+c)/2

 #s=math.sqrt(h\*(h-a)\*(h-b)\*(h-c))

else:

print("Error a,b,c values!")

return s

x=eval(input())

y=eval(input())

z=eval(input())

area=sanjiao(x,y,z)

print(area)

请根据提示完成下方程序功能，程序中用到的map()和filter()函数均接收参数为：一个函数和一个序列，并依次将序列的数据逐个应用到函数中进行处理； 请在空白处完善下方代码使其能输出下面信息：

依次统计：1报名人数大于40人 的课程信息： 2.只输出需要查看的课程名字： 3.按照课程的（人数）增序排序：

如：

count>40: [('Python', 70), ('Java', 65), ('Math', 96)]

the course name: ['Python', 'Java', 'C++', 'Math', 'Photo', 'Physical']

order by student number: [('C++', 20), ('Photo', 30), ('Physical', 38), ('Java', 65), ('Python', 70), ('Math', 96)]

请在下方空白处填写正确代码

word = {'Python':70, 'Java':65, 'C++':20, 'Math':96, 'Photo':30,'Physical':38}

word\_list=



1分

filtered\_list = list(filter(lambda



2分

, word\_list))

print('count>40:', filtered\_list)

mapped\_list = list(map(lambda



1分

, word\_list))

print('the course name:', mapped\_list)

print('order by student number:', sorted(word\_list, key=lambda



2分

阅读程序或#提示信息，然后请在空格内填入正确代码使其运行后能得到下列结果：

module pi= 3.141592653589793

module e= 2.718281828459045

pi= 3.14

e= 2.718

module pi= 3.14

module e= 2.718281828459045

程序如下：

def my\_func():



1分

pi =



1分

print('pi=', pi)

e =



1分

print('e=', e)

pi = 3.141592653589793

e = 2.718281828459045

print('module pi=', pi) #输出全局变量的值

print('module e=', e) #输出全局变量的值



1分

print('module pi=', pi)

print('module e=', e)

(部分正确)