

计算机二级 Python 精选 200 题

备注：视频解析都上传到（刷题班）或（全程班）或（冲刺班）或（寒暑假班）对应的课程，可以打开小黑课堂 APP，我的课程目录【最下面观看视频讲解】。

1、以下变量名中,不能作为变量名的是()。

- A、p
- B、Temp
- C、3p
- D、_fp

2.以下变量名中,符合 Python 语言变量命名规则的是()。

- A、33_keyword
- B、key@word33_
- C、break
- D、_33keyword

3、以下关于 Python 缩进的描述中,错误的是()。

- A、缩进表达了所属关系和代码块的所属范围
- B、缩进是可以嵌套的,从而形成多层缩进
- C、判断、循环、函数等都能够通过缩进包含一批代码
- D、Python 用严格的缩进表示程序的格式框架,所有代码都需要在行前至少加一个空格

4、下列关于 Python 缩进的描述中,错误的是()。

- A、Python 语言中采用严格的"缩进"来表明程序格式不可嵌套
- B、判断、循环、函数等语法形式能够通过缩进包含一批 Python 代码,进而表达对应的语义
- C、Python 单层缩进代码属于之前最邻近的一行非缩进代码,多层缩进代码根据缩进关系决定所属范围
- D、缩进指每一行代码前面的留白部分,用来表示代码之间的层次关系

5、以下关于 Python 语言技术特点的描述中,错误的是()。

- A、对于需要更高执行速度的功能,例如数值计算和动画,Python 语言可以调用 C 语言编写的底层代码
- B、Python 比大部分编程语言具有更高的软件开发产量和简洁性
- C、Python 语言是解释执行的,因此执行速度比编译型语言慢
- D、Python 是脚本语言,主要用作系统编程和 Web 访问的开发语言

6、以下不是 Python 语言特点的是()

- A、执行速度快
- B、语法简洁

- C、支持中文
- D、生态丰富

7、下列哪个语句在 Python 中是非法的? ()

- A、`x=y=z=1`
- B、`x=(y=z+1)`
- C、`x,y=y,x`
- D、`x+=y`

8、以下代码的输出结果是 ()。

```
x = 2 + 9 * ((3*12) - 8) // 10  
print(x)
```

- A、26
- B、27.2
- C、28.2
- D、27

9、表达式 `3**2*4//6%7` 的计算结果是()。

- A、3
- B、5
- C、4
- D、6

10、以下选项不是 Python 保留字的是()。

- A、For
- B、try
- C、False
- D、from

11、以下代码的输出结果是 ()。

```
print( 0.1 + 0.2 == 0.3 )
```

- A、True
- B、False
- C、-1
- D、0

12、以下关于Python循环结构的描述中，错误的是 ()。

- A、continue只结束本次循环
- B、遍历循环中的遍历结构可以是字符串、文件、组合数据类型和range()函数等
- C、Python通过for、while等保留字构建循环结构
- D、break用来结束当前当次语句，但不跳出当前的循环体

13、下列关于Python运算符的使用描述正确的是()。

- A、`a!=b`,比较a与b是否不相等
- B、`a+=b`,等同于`a=a+b`
- C、`a==b`,比较a与b是否相等

D、`a//b`等同于`a = a/b`

14、字符串`tstr='television'`，显示结果为`vi`的选项是（ ）。

- A、`print(tstr[-6:6])`
- B、`print(tstr[5:7])`
- C、`print(tstr[4:7])`
- D、`print(tstr[4:-2])`

15、设`str1='*@python@*'`，语句`print(str1[2:].strip('@'))`的执行结果是（ ）。

- A、`*@python@*`
- B、`python*`
- C、`python@*`
- D、`*python*`

16、以下代码的输出结果是（ ）。

```
x = 'R\0S\0T'
```

```
print(len(x))
```

- A、3
- B、5
- C、7
- D、6

17、以下关于Python字符串的描述中，错误的是（ ）。

- A、空字符串可以表示为`""`或`''`
- B、在Python字符串中，可以混合使用正整数和负整数进行索引和切片
- C、字符串`'my\\text.dat'`中第一个`\`表示转义符
- D、Python字符串采用`[N:M]`格式进行切片，获取字符串从索引`N`到`M`的子字符串（包含`N`和`M`）

18、Python语言提供三种基本的数字类型，它们是（ ）。

- A、整数类型、浮点数类型、复数类型
- B、整数类型、二进制类型、浮点数类型
- C、复数类型、二进制类型、浮点数类型
- D、整数类型、二进制类型、复数类型

19、以下关于Python语言复数类型的描述中,错误的是()。

- A、复数可以进行四则运算
- B、实部不可以为0
- C、Python语言中可以使用`z.real`和`z.imag`分别获取它的实部和虚部
- D、复数类型与数学中复数的概念一致

20、以下描述中，错误的是（ ）。

A、如果 s 是一个序列，s =[1,"kate",True]，s[3]返回 True

B、如果 x 不是 s 的元素，x not in s 返回 True

C、如果x是s的元素，x in s返回True

D、如果s是一个序列，s =[1,"kate",True]，s[-1]返回True

21、s='1234567890'，以下表示'1234'的选项是（ ）。

A、s[-10:-5]

B、s[0:3]

C、s[0:4]

D、s[1:5]

22、表达式int('100/3')的执行结果是（ ）。

A、'100/3'

B、33.3

C、33

D、ValueError

23、以下语句的执行结果是（ ）。

```
str1='||北京大学|'
```

```
print(str1.strip('|').replace('北京','南京'))
```

A、南京大学||

B、南京大学

C、||南京大学||

D、||南京大学

24、在Python语言中，使用for..in..方式形成的循环不能遍历的类型是（ ）。

A、字典

B、列表

C、浮点数

D、字符串

25、以下代码的输出结果是（ ）。

```
for s in "PythonNCRE":
```

```
    if s=="N":
```

```
        break
```

```
    print(s,end="")
```

A、N

B、PythonCRE

C、Python

D、PythonNCRE

26、以下的描述中，不属于 Python 语言控制结构的是（ ）。

A、循环结构

B、分支结构

- C、跳转结构
- D、顺序结构

27、以下关于分支结构的描述中，错误的是（ ）。

- A、二分支结构有一种紧凑形式，使用保留字if和elif实现
- B、if 语句中条件部分可以使用任何能够产生 True 和 False 的语句和函数
- C、if语句中语句块执行与否依赖于条件判断
- D、多分支结构用于设置多个判断条件以及对应的多条执行路径

28、以下程序的输出结果是（ ）。

```
a = [3, 2, 1]
for i in a[::-1]:
    print(i, end=' ')
```

- A、1,2,3
- B、3 2 1
- C、1 2 3
- D、3,2,1

29、以下代码的输出结果是（ ）。

```
for s in "testatest":
    if s=="a" or s=="e":
        continue
    print(s, end="")
```

- A、tsttst
- B、testatest
- C、testtest
- D、tstatst

30、以下程序的输出结果是（ ）。

```
for i in "Nation":
    for k in range(2):
        if i == 'n':
            break
    print(i, end="")
```

- A、aattiioo
- B、NNaattiioo
- C、Naattiioon
- D、aattiioonn

31、以下关于分支和循环结构的描述中，错误的是（ ）。

- A、所有的for分支都可以用while循环改写
- B、while循环只能用来实现无限循环
- C、可以终止一个循环的保留字是break
- D、continue可以停止后续代码的执行，从循环的开头重新执行

32、以下关于"for <循环变量> in <循环结构>"的描述，错误的是（ ）。

- A、这个循环体语句中不能有break语句，会影响循环次数
- B、<循环结构>采用[1,2,3]和['1','2','3']的时候，循环的次数是一样的
- C、使用range(a,B)函数指定for循环的循环变量取值是从a到b-1
- D、for i in range(1,10,2)表示循环5次，i的值是从1到9的奇数

33、执行以下程序，输入"fish520"，输出结果是（ ）。

```
w = input()
```

```
for x in w:
```

```
    if '0' <= x <= '9':
```

```
        continue
```

```
    else:
```

```
        w.replace(x,"")
```

```
print(w)
```

- A、 520
- B、 fish520
- C、 fish
- D、 520fish

34、执行以下程序，输 ab，然后回车，结果是（ ）。

```
k = 10
```

```
while True:
```

```
    s = input("请输入q退出：")
```

```
    if s == 'a':
```

```
        k+=1
```

```
        continue
```

```
    else:
```

```
        k+=2
```

```
        break
```

```
print(k)
```

- A、 13
- B、 请输入q退出：
- C、 12

D、10

35、以下程序的输出结果是（ ）。

```
x = 10
```

```
y = 0
```

```
if (x > 5) or (x/y > 5):
```

```
    print('Right')
```

```
else:
```

```
    print('Wrong')
```

A、报错：ZeroDivisionError

B、Wrong

C、Right

D、不报错，但不输出任何结果

36、以下关于函数优点的描述中，错误的是（ ）。

A、函数便于阅读

B、函数可以使程序更加模块化

C、函数可以减少代码重复

D、函数可以表现程序的复杂度

37、以下关于 Python 函数的描述中，错误的是（ ）。

A、可以定义函数接受可变数量的参数

B、定义函数时，某些参数可以赋予默认值

C、函数必须要有返回值

D、函数可以同时返回多个结果

38、使用Python的函数,需要的步骤不包括()。

A、函数测试

B、函数定义

C、函数调用

D、函数执行

39、以下程序的输出结果是()。

```
f= lambda x:5
```

```
f(3)
```

A、3

B、5

C、3 5

D、35

40、当用户输入5时，下面代码的输出结果是（ ）。

try:

n = input("请输入一个整数:")

def pow2(n):

return n**2

pow2(n)

except:

print("程序执行错误")

A、程序执行错误

B、25

C、程序没有任何输出

D、5

41、以下关于程序的异常处理的描述中,错误的是()。

A、在 try...except...else 结构中,如果 try 块的语句引发了异常则会执行 else 块中的代码

B、异常处理结构中的 finally 块中的代码仍然有可能出错从而再次引发异常

C、一般不建议在 try 中放太多代码, 而是应该只放入可能会引发异常的代码

D、在异常处理结构中,不论是否发生异常,finally 子句中的代码总是会执行的

42、以下关于 Python 语言 return 语句的描述中, 正确的是（ ）。

A、return只能返回一个值

B、函数必须有 return 语句

C、函数可以没有return语句

D、函数中最多只有一个return语句

43、以下关于Python全局变量和局部变量的描述中, 错误的是（ ）。

A、局部变量在函数内部创建和使用, 函数退出后变量被释放

B、全局变量一般指定义在函数之外的变量

C、使用global保留字声明后, 变量可以作为全局变量使用

D、当函数退出时, 局部变量依然存在, 下次函数调用可以继续使用

44、以下程序的输出结果是（ ）。

def myf(x = 2.0,y = 4.0):

global sp

sp += x * y

return(sp)

sp = 100


```
print(sp,myf(4,3))
```

- A、 112 100
- B、 100 112
- C、 100 100
- D、 112 112

45、以下程序的输出结果是（ ）。

```
def loc_glo( b = 2, a = 4):
```

```
    global z
```

```
    z += 3 * a +5 * b
```

```
    return z
```

```
z = 10
```

```
print(z, loc_glo (4,2))
```

- A、 10 36
- B、 32 32
- C、 36 36
- D、 10 32

46、以下不是 Python 组合数据类型的是（ ）。

- A、复数类型
- B、字符串类型
- C、集合类型
- D、字典类型

47、以下不能用于生成空字典的选项是（ ）。

- A、 dict()
- B、 dict()
- C、 {}
- D、 {}

48、列表变量ls 共包含 10 个元素，ls 索引的取值范围是（ ）。

- A、 -1到-9(含)的整数
- B、 0到10(含)的整数
- C、 1到10(含)的整数
- D、 0到9(含)的整数

49、以下代码的输出结果是（ ）。

```
S = 'Pame'
```

```
for i in range(len(S)):
```

```
    print(S[-i],end="")
```

- A、 ameP
- B、 emaP
- C、 Pema
- D、 Pame

50、以下程序的输出结果是（ ）。

```
lt=['绿茶','乌龙茶','红茶','白茶','黑茶']
```

```
ls=lt
```

```
ls.clear()
```

```
print(lt)
```

A、变量未定义的错误

B、[]

C、['绿茶','乌龙茶','红茶','白茶','黑茶']

D、'绿茶','乌龙茶','红茶','白茶','黑茶'

51、以下代码的输出结果是（ ）。

```
CList = list(range(5))
```

```
print(2 in CList)
```

A、0

B、False

C、True

D、-1

52、以下代码的输出结果是（ ）。

```
ls = ["car","truck"]
```

```
def funC(A):
```

```
    ls.append(A)
```

```
    return
```

```
funC("bus")
```

```
print(ls)
```

A、["bus"]

B、["car","truck"]

C、["car","truck","bus"]

D、[]

53、以下关于列表变量ls操作的描述中，错误的是（ ）。

A、ls.reverse(): 反转列表ls中所有元素

B、ls.append(x): 在ls最后增加一个元素

C、ls.copy(): 生成一个新列表，复制ls的所有元素

D、ls.clear(): 删除ls的最后一个元素

54、以下关于 Python 字典变量的定义中，错误的是（ ）。

- A、d = {1:[1,2], 3:[3,4]}
- B、d = {[1,2]:1, [3,4]:3}
- C、d = {(1,2):1, (3,4):3}
- D、d = {'张三':1, '李四':2}

55、下面代码的输出结果是()。

```
letter=['A','B','C','D','D','D']
```

```
for i in letter:
```

```
    if i == 'D':
```

```
        letter.remove(i)
```

```
print(letter)
```

- A、['A','B','C']
- B、['A','B','C','D','D']
- C、['A','B','C','D','D','D']
- D、['A','B','C','D']

56、以下代码的输出结果是 ()。

```
ls = []
```

```
for m in 'AB':
```

```
    for n in 'CD':
```

```
        ls.append(m+n)
```

```
print(ls)
```

- A、ABCD
- B、AABBCCDD
- C、ACADBCBD
- D、['AC', 'AD', 'BC', 'BD']

57、以下代码的输出结果是 ()。

```
def fibRate(n):
```

```
    if n <= 0:
```

```
        return -1
```

```
    elif n == 1:
```

```
        return -1
```

```
    elif n == 2:
```

```
        return 1
```

```
    else:
```

```
        L = [1, 1]
```

```
        for i in range(2,n):
```

```
L.append(L[-1]+L[-2])
return L[-2]/L[-1]
```

```
print(fibRate(5))
```

- A、 -1 B、 0.625
C、 0.6 D、 0.5

58、下面的d是一个字典变量，能够输出数字5的语句是（ ）。

```
d = {'food':{'cake':1,'egg':5},'cake':2,'egg':3}
```

- A、 `print(d['egg'])`
B、 `print(d['food']['egg'])`
C、 `print(d['food'][-1])`
D、 `print(d['cake'][1])`

59、d = {"大海": "蓝色", "天空": "灰色", "大地": "黑色"}

```
print(d["大地"], d.get("大地", "黄色"))
```

- A、黑色 黑色
B、黑色 灰色
C、黑色 黄色
D、黑色 蓝色

60、以下程序的输出结果是（ ）。

```
l1=['aa',[2,3,3.0]]
```

```
print(l1.index(2))
```

- A、 2
B、 3.0
C、 3
D、 ValueError

61、为以下程序填空，使得输出结果是{40:'yuwen',20:'yingyu',30:'shuxue'}的选项是（ ）。

```
tb = {'yingyu':20, 'shuxue':30, 'yuwen':40}
```

$$\text{stb} = \{\}$$

```
for it in tb.items():
```

```
print(it)
```

```
print(stb)
```

- A、 `stb[it[1]]=tb[it[0]]`
 B、 `stb[it[1]]=stb[it[0]]`
 C、 `stb[it[1]]=tb[it[1]]`
 D、 `stb[it[1]]=it[0]`

62、以下程序的输出结果是（ ）。

```
ls = [12,44,23,46]
for i in ls:
    if i == '44':
        print('found it! i = ',i)
        break
else:
    print('not found it ...')
```

- A、 not found it...
- B、 found it!i=44
- C、 found it!i=44
not found it...
- D、 found it!i='44'
not found it...

63、给以下程序填空，能输出{0:[90,'Aele'],1:[87,'Bob'],2:[93,'lala']}结果的选项是（ ）。

```
x = [90,87,93]
y = ("Aele", "Bob","lala")
z = {}
for i in range(len(x)):
```

```
    _____
    print(z)
```

- A、 z[i]=[x[i],y[i]]
- B、 z[i]=x[i],y[i]
- C、 z[i]=list(zip(x,y))
- D、 z[i]=x,y

64、字典d={'Python':123,'C':123,'C++':123},len(d)的结果是（ ）。

- A、 3
- B、 6
- C、 9
- D、 12

65、对于序列s，能够返回序列s中第i到第j以k为步长的子序列的表达是（ ）。

- A、 s[i:j:k]
- B、 s[i,j,k]
- C、 s[i;j;k]
- D、 s(i,j,k)

66、以下程序的输出结果是()。

```
img1 = [12,34,56]
img2 = [1,2,3,4]
def disp1(img):
    print(img)
img1 = img2
img1.append([5,6])
disp1(img2)
```

- A、 [12,34,56]
- B、 [1,2,3,4,5,6]
- C、 [1,2,3,4]
- D、 [1,2,3,4,[5,6]]

67、执行以下程序后，要输出ab44cd46结果，该执行的选项是 ()。

```
ls = ['ab','44','cd','46']
ls.reverse()
```

- A、 print(''.join(ls[-1::]))
- B、 print(''.join(ls))
- C、 print(''.join(ls[1:]))
- D、 print(''.join(ls[-1::-1]))

68、以下程序的输出结果是 ()。

```
ss = [2,3,6,9,7,1]
for i in ss:
    print(max(ss),end = ',')
    ss.remove(max(ss))
```

- A、 9,7,6
- B、 9,7,6,
- C、 9,7,6,3,2,1,
- D、 9,7,6,3,2,1

69、以下语句的执行结果是 ()。

```
dd={'a':90,'b':87,'c':93}
print(max(dd),min(dd))
```

- A、 c a
- B、 93 90
- C、 c b

D、 93 87

70、以下语句的执行结果是（ ）。

```
dd = {'chinese':200, 'pe':187, 'Gana':930}
```

```
print(dd.keys())
```

A、 200,187,930

B、 'chinese': 'pe': 'Gana'

C、 dict_keys(['chinese', 'pe', 'Gana'])

D、 ['chinese', 'pe', 'Gana']

71、以下程序的执行结果是（ ）。

```
x = [90,87,93]
```

```
y = ("Aele", "Bob", "lala")
```

```
z = {}
```

```
for i in range(len(x)):
```

```
    z[x[i]] = y[i]
```

```
print(z)
```

A、 {'Aele':90,'Bob':87,'lala':93}

B、 {'90': 'Aele', '87': 'Bob', '93': 'lala'}

C、 {'Aele': '90', 'Bob': '87', 'lala': '93'}

D、 {90: 'Aele', 87: 'Bob', 93: 'lala'}

72、以下语句的执行结果是()。

```
dic = {90:'Aele',87:'Bob',93:'lala'}
```

```
print(dic.pop(90,-1),len(dic))
```

A、 Aele 3

B、 -1 3

C、 Aele 2

D、 -1 2

73、以下语句的执行结果是（ ）。

```
ls = [120,'120',int('120'),12*10]
```

```
print(ls.index(120),ls.count(120))
```

A、 0 3

B、 0 2

C、 2 4

D、 1 3

74、下面语句的执行结果是（ ）。

```
tstr = 'Baidu Translation is an online translation service'  
print(len(tstr.split('a')))
```

- A、 9
- B、 6
- C、 7
- D、 8

75、不能返回列表数据类型的选项是（ ）。

- A、 range()
- B、 s.split()
- C、 dict.items()
- D、 s.lcut()

76、下列关于Python内置函数的使用描述错误的是()。

- A、 int(x)是将x转换为一个整数
- B、 print("6+5")的输出是11
- C、 dic.clear()的结果是清空字典dic
- D、 使用append()函数可以向列表添加元素

77、在Python语言中，写文件的操作是（ ）。

- A、 write
- B、 writeall
- C、 seek
- D、 writetext

78、S 和 T 是两个集合,对 S&T 的描述正确的是()。

- A、 S 和 T 的交运算,包括同时在集合 S 和 T 中的元素
- B、 S 和 T 的并运算,包括在集合 S 和 T 中的所有元素
- C、 S 和 T 的差运算,包括在集合 S 但不在 T 中的元素
- D、 S 和 T 的补运算,包括几个 S 和 T 中的非相同元素

79、以下对 Python 文件处理的描述中，错误的是（ ）。

- A、 Python通过解释器内置的open()函数打开一个文件
- B、 Python能够以文本和二进制两种方式处理文件
- C、 当文件以文本方式打开时，读写按照字节流方式
- D、 文件使用结束后可以用close()方法关闭，释放文件的使用授权

80、以下关于Python二维数据的描述中，错误的是（ ）。

- A、一种通用的二维数据存储形式是CSV格式
- B、二维数据由多条一维数据构成，可以看作一维数据的组合形式
- C、表格数据属于二维数据，由整数索引的数据构成
- D、CSV格式每行表示一个一维数据，用英文半角逗号分隔

81、以下关于文件的描述中，错误的是（ ）。

- A、文件打开后，可以用seek()控制对文件内容的读写位置
- B、采用readlines()可以读入文件中的全部文本，返回一个列表
- C、使用open()打开文件时，必须要用r或w指定打开方式，不能省略
- D、如果没有采用close()关闭文件，Python程序退出时文件将被自动关闭

82、以下关于Python二维数据的描述中，错误的是（ ）。

- A、从CSV文件获得数据内容后，可以用replace()来去掉每行最后的换行符
- B、CSV文件的每一行是一维数据，可以用列表、元组和字典表示
- C、若一个列表变量里的元素都是字符串类型，则可以用join()合成字符串
- D、列表中保存的二维数据，可以通过循环用write()写入CSV文件

83、假设city.csv文件内容如下（ ）。

巴哈马,巴林,孟加拉国,巴巴多斯

白俄罗斯,比利时,伯利兹

以下代码的输出结果是：

```
f = open("city.csv", "r")
```

```
ls = f.read().split(",")
```

```
f.close()
```

```
print(ls)
```

- A、['巴哈马', '巴林', '孟加拉国', '巴巴多斯', '白俄罗斯', '比利时', '伯利兹']
- B、['巴哈马', '巴林', '孟加拉国', '巴巴多斯', '白俄罗斯', '比利时', '伯利兹']
- C、['巴哈马', '巴林', '孟加拉国', '巴巴多斯\n白俄罗斯', '比利时', '伯利兹']
- D、['巴哈马', '巴林', '孟加拉国', '巴巴多斯', '\n', '白俄罗斯', '比利时', '伯利兹']

84、以下不属于Python的pip工具命令的选项是（ ）。

- A、install
- B、-V
- C、show
- D、download pip

85、用Pyinstaller工具打包Python源文件时-F参数的含义是（ ）。

- A、指定所需要的第三方库路径
- B、在dist文件夹中只生成独立的打包文件
- C、指定生成打包文件的目录
- D、删除生成的临时文件

86、表格类型数据的组织维度最可能是（ ）。

- A、二维数据
- B、一维数据
- C、多维数据
- D、高维数据

87、f=open()可以打开一个文件，关于f的描述错误的是（ ）。

- A、f是一种特殊的Python变量，执行print(f)时会报错
- B、f是文件对象引用，在程序中表示文件
- C、f.read()可以一次性读入文件全部信息
- D、执行m=f后，m和f同时表示所打开文件

88、以下语句的执行结果是（ ）。

```
with open('A. txt','w+') as fo:
```

```
    fo.write('hello')
```

```
    print(fo.read())
```

- A、出现读写异常
- B、读文件的函数从文件里读到字符串'hello'
- C、写到文件a.txt里一个'hello'字符串，显示在屏幕上一个'hello'字符串
- D、写到文件a.txt里一个'hello'字符串

89、以下关于数据维度的描述，不正确的是（ ）。

- A、二维数据处理等同于二维列表的操作
- B、要处理二维数据的每个元素，需要用循环结构处理
- C、二维数据用CSV文件格式存储，得用二重循环嵌套实现
- D、CSV文件里是用英文半角逗号来分隔数据元素

90、以下选项不属于Python的pip工具的功能是（ ）。

- A、打包python源代码
- B、安装第三方python库
- C、卸载已经安装了的第三方库
- D、对第三方库进行基本的维护

91、下面哪一种导入方式是错误的()。

- A、import numpy
- B、import ndarray from numpy
- C、from numpy import *
- D、import numpy as np

92、下面属于Python语言的标准库是()。

- A、turtle
- B、jieba
- C、scipy
- D、flask

93、不能返回列表数据类型的选项是 ()。

- A、list(range())
- B、s.split()
- C、dict.items()
- D、jieba.lcut()

94、以下不属于 Python 的内置函数的选项是 ()。

- A、abs()
- B、sum()
- C、input()
- D、get()

95、在 Python 语言中，包含矩阵运算的第三方库是 ()。

- A、wxPython
- B、PyQt5
- C、wordcloud
- D、numpy

96、在 Python 语言中，能够处理图像的第三方库是 ()。

- A、PIL
- B、pyserial
- C、requests
- D、pyinstaller

97、在Python语言中，属于网络爬虫的第三方库的是 ()。

- A、PIL

- B、pyserial
- C、requests
- D、pyinstaller

98、在 Python 语言中，属于网络爬虫领域的第三方库是（ ）。

- A、PyQt5
- B、numpy
- C、openpyxl
- D、scrapy

99、在Python语言中，不属于机器学习领域第三方库的是（ ）。

- A、MXNet
- B、Tensorflow
- C、PyTorch
- D、Arcade

100、关于 BeautifulSoup4 的描述，正确的选项是（ ）。

- A、用于处理三维可视化
- B、用于绘制图形
- C、这个库能遍历处理HTML网页上的语法元素
- D、用于解析和处理 HTML 和 XML

101、关于Python代码编写原则的描述，错误的选项是（ ）。

- A、保持代码的简洁
- B、保持代码的可读性、减少嵌套
- C、变量的定义和引用范围要清晰
- D、尽量使用函数和类封装代码

102、关于Python赋值语句的描述，错误的选项是（ ）。

- A、对于a=100语句，无论变量a是什么类型，该赋值语句运行一定正确
- B、a,b = b,a可以交换a和b的值
- C、使用符号"="表达赋值关系
- D、赋值语句要求赋值两侧的数据类型一致

103、函数名称定义错误的选项是（ ）。

- A、turtle()
- B、_func()
- C、fun*1()

D、max()

104、将下列表达式作为eval()函数的参数，执行结果错误的选项是（ ）。

A、"input()"

B、"print()"

C、"1+2"

D、1+2

105、关于类型转换的描述，错误的选项是（ ）。

A、int(1.23)能将浮点数1.23转换为整数

B、int('1.23')能将字符串转换为整数

C、str(1+2j)能将复数1+2j转换为字符串类型

D、int(1+2j)不能将复数1+2j转换为整数类型，执行出错

106、以下代码的执行结果是（ ）。

```
a = 2.71828182459
```

```
if isinstance(a ,int):
```

```
    print("{} is int. ".format( round(a, 2)))
```

```
else:
```

```
    print("{} is not int.".format (round(a,2)))
```

A、2.72 is not int.

B、2.72 is int.

C、执行错误

D、2.71828182459 is not int.

107、关于Python分支结构的描述，错误的选项是（ ）。

A、二分支结构可以用if else语句表示

B、多分支结构可以用if elif else语句表示

C、单分支结构可以用if语句表示

D、多分支结构可以用switch语句表示

108、以下代码的执行结果是（ ）。

```
a=eval("12+3")
```

```
if type(a) == type(123):
```

```
    print("整数类型")
```

```
elif type(a) == type("123"):
```

```
    print("字符串类型")
```

```
else:
```

```
print("其他类型")
```

- A、字符串类型
- B、其他类型
- C、代码执行错误提示
- D、整数类型

109、以下代码的执行结果是（ ）。

```
a=1000
```

```
while (a>123):
```

```
    for i in range(10):
```

```
        a=a-a//123
```

```
        if a<200:
```

```
            break
```

```
print(a)
```

- A、1000
- B、123
- C、199
- D、执行错误

110、关于异常处理的描述，错误的选项是（ ）。

- A、异常处理中try是必须的，except并不必须
- B、try、except、else、finally都可以用于异常处理
- C、except后面可以增加异常类型，进而区分不同异常进行处理
- D、try和except都是异常处理保留字

111、以下代码的执行结果是（ ）。

```
start = 2
```

```
n = 0
```

```
while start <=10:
```

```
    start = start +2
```

```
    n=n+ 1
```

```
print(n)
```

- A、4
- B、6
- C、5
- D、0

112、关于函数作用的描述，错误的选项是（ ）。

- A、复用代码
- B、降低编程复杂度
- C、提高代码执行速度
- D、增加代码的可读性

113、关于return语句的描述，正确的选项是（ ）。

- A、函数必须有一个return语句
- B、函数可以没有return语句
- C、函数最多只有一个return语句
- D、return语句只能返回一个值

114、以下代码的执行结果是（ ）。

```
def func (n):  
    n=n +2  
    return n
```

a= 10

func (a)

print(a)

- A、 10
- B、 12
- C、 0
- D、 11

115、关于以下代码的描述，错误的选项是（ ）。

```
def fact(n):  
    s = 1  
    for i in range(1, n+1):  
        s*=i  
    return s
```

- A、代码中n是可选参数
- B、s是局部变量
- C、s不能在函数外使用
- D、range()函数是Python内置函数

116、不属于组合数据类型的选项是（ ）。

- A、集合类型
- B、元组类型
- C、字典类型

D、数组类型

117、有一组获奖学生名单，从第1名到第6名的学生姓名为：张良，韩梅梅，李亮，王红，张晓晓，李梅。要求名单一旦创建，数据不能修改，不能打乱先后次序。适合存储该组数据的数据类型是（ ）。

- A、集合类型
- B、列表类型
- C、元组类型
- D、字典类型

118、关于Python列表操作的描述，错误的选项是（ ）。

- A、ls是一个列表，ls.insert(j,x)可以在列表ls第j个位置插入一个元素x
- B、ls是一个列表，ls.append(x)可以在列表ls第1个位置插入一个元素x
- C、ls是一个列表，ls.remove(x)将列表ls中出现的第1个元素x删除
- D、ls是一个列表，ls.pop(i)可以返回列表ls中第i个元素，并删除列表中的该元素

119、以下代码的执行结果是（ ）。

```
s=[1,"Alice",True]
print(s[1] and s[-1])
```

- A、True
- B、"Alice"
- C、1
- D、Alice

120、为了使以下代码的执行结果是：中国,韩国,日本,巴基斯坦,蒙古

```
country = ['中国','韩国','日本','巴基斯坦','蒙古']
```

```
n=1
```

```
for c in country:
```

```
    if n < len(country):
```

```
        print(_____)
```

```
    else:
```

```
        print(c)
```

```
    n=n+1
```

在横线处要补充的代码是（ ）。

- A、c
- B、c,end=","
- C、c,","

D、c,sep=","

121、以下代码可能的执行结果是（ ）。

```
list1 = ['河南','四川','河北','辽宁']
list2 = ['郑州','成都','石家庄','长春','济南']
d={}
for i in range(len(list1)):
    d[list1[i]] = list2[i]
print(d)
```

- A、{'河北':'石家庄','河南':'郑州','辽宁':'长春','四川':'成都'}
- B、[('河南','郑州'),('四川','成都'),('河北','石家庄'),('辽宁','长春')]
- C、(['河南','四川','河北','辽宁'])
- D、['河南','四川','河北','辽宁']

122、执行后返回列表类型的函数是（ ）。

- A、replace()
- B、strip()
- C、lower()
- D、split()

123、执行以下代码后，a.txt文件里的内容是（ ）。

```
fo=open("a.txt","w")
x=['天','地','人']
fo.writelines(','.join(x))
fo.close()
```

- A、天,地,人
- B、'天','地','人',
- C、天,地,人,
- D、'天','地','人'

124、关于Python数据维度的描述，错误的选项是（ ）。

- A、一维数据可以用列表表示，也可以用集合表示
- B、高维数据可以用字典表示
- C、二维数据由多个一维数据构成，可以看作一维数据的组合形式
- D、列表可以表示一维数据，各个元素的类型必须是相同的

125、设置图形绘制速度的turtle库函数是（ ）。

- A、goto()

- B、speedup()
- C、speed()
- D、getturtle()

126、关于以下代码执行结果的描述，正确的选项是（ ）。

```
import random
a = random.randint(1,100)
while not a:
    if a <= 50 and a >= 40:
        a = random.random()
        break
print(a)
```

- A、1到100之间的整数
- B、1到100之间的整数或浮点数
- C、0.0到1.0之间的浮点数
- D、40到50之间的整数或浮点数

127、属于Python数据分析方向的第三方库是（ ）。

- A、pandas
- B、moviepy
- C、Django
- D、pefile

128、属于Python机器学习方向的第三方库是（ ）。

- A、scikit-learn
- B、SQLAlchemy
- C、requests
- D、Click

129、当用户输入5时，下面代码的输出结果是（ ）。

```
try:
    n = input("请输入一个整数:")
    def pow2(n):
        return n**2
    pow2(n)
except :
    print("程序执行错误")
```

- A、程序没有任何输出

- B、25
- C、程序执行错误
- D、5

130、以下保留字不属于分支或循环逻辑的是（ ）。

- A、while
- B、for
- C、in
- D、elif

131、关于Python语言发展的描述，最次要的因素是（ ）。

- A、解释器采用Unicode编码，支持中文处理
- B、语言开源开放，能够整合C语言代码
- C、各领域应用广泛，通用性强
- D、语法简洁、精炼，语言设计质量高

132、不是Python保留字的选项是（ ）。

- A、define
- B、break
- C、await
- D、finally

133、关于变量名称的定义，错误的选项是（ ）。

- A、false
- B、a123
- C、pass
- D、And

134、关于input()函数的描述，错误的选项是（ ）。

- A、用户输入的信息全部被当作一个字符串处理
- B、用户可以输入多行信息，并将被当作一个字符串处理
- C、input()参数用于提示用户，不影响用户输入的内容
- D、input()参数只能是字符串类型

135、关于字符串类型转换的描述，错误的选项是（ ）。

- A、str(123)的结果是"123"
- B、str(1.23)的结果是'1.23'
- C、str(1+2j)的结果是'(1+2j)'

D、`str(1+2)`对表达式`1+2`进行字符串转换，结果是`'1+2'`

136、关于数字运算符的描述，错误的选项是（ ）。

- A、运算符`%`实现取余操作，参与运算的可以是浮点数，结果可能是浮点数
- B、运算符`/`实现除法操作，如果参与运算的都是整数，结果只能是整数
- C、运算符`*`实现乘法操作，参与运算的可以是任意数字类型
- D、运算符`**`实现幂运算操作，幂数可以是整数

137、关于操作字符串变量`s`的使用方式，错误的选项是（ ）。

- A、`s.len()`
- B、`s.format()`
- C、`s.split()`
- D、`s.join()`

138、字符串`s = '中华人民共和国国庆日是10月1日'`，使用表达式提取`s`中的日期`'10月1日'`，正确的选项是（ ）。

- A、`s[12:]`
- B、`s[-5:-1]`
- C、`s[-5:]`
- D、`s[-4:]`

139、不用于Python循环结构保留字的选项是（ ）。

- A、`else`
- B、`continue`
- C、`break`
- D、`lambda`

140、以下代码的执行结果是（ ）。

```
a= "123"
if a > "Python":
    print("再学Python")
else:
    print("初学Python")
```

- A、初学Python
- B、再学Python
- C、没有输出
- D、执行出错

141、以下代码的执行结果是（ ）。

```
x = 99
a = list(range(100))
while x < len(a):
    a = a[:x]
    x -= 1
    if len(a) == 5:
        break
print(a)
```

- A、[0,1....98,99]
- B、[0,1,2,3,4]
- C、[4,3,2,1,0]
- D、执行出错

142、关于程序异常处理的描述，错误的选项是（ ）。

- A、在进行关键计算时使用异常处理，如除0运算
- B、在获得用户输入时使用异常处理，避免输入格式异常
- C、在读取文件时使用异常处理，避免文件读取异常
- D、在所有运算中都应该使用异常处理，避免程序任何可能的错误

143、关于while循环的描述，正确的选项是（ ）。

- A、while循环可以实现无限循环
- B、while循环不能实现计数循环
- C、while循环与遍历循环等效
- D、while循环不能用于函数内部

144、关于函数的描述，正确的选项是（ ）。

- A、每个递归函数都只能有一个基例
- B、一个函数中只允许有一条return语句
- C、定义函数和调用该函数的代码可以写在不同的Python程序里
- D、每个函数都必须有return语句

145、以下代码的执行结果是（ ）。

```
def func(m, n):
    m = 2
    s = m * n
    return s
m = 5
```

```
n = 2
print(func(5,2))
```

- A、 10
- B、 4
- C、 25
- D、 5

146、以下代码不可能的执行结果是（ ）。

```
import random
def func(n):
    if n == 1 or n == 2:
        return 1
    else:
        return random.randint(1,n-1)
print(func(10))
```

- A、 3
- B、 2
- C、 1
- D、 10

147、关于函数全局变量和局部变量的描述，错误的选项是（ ）。

- A、 在函数内部引用局部变量时不用nonlocal保留字声明
- B、 在函数内部引用全局变量时一定要用global声明
- C、 全局变量指在函数外部创建的变量
- D、 在函数内部创建的局部变量，在函数外部不能引用

148、关于Python组合数据类型的描述，正确的选项是（ ）。

- A、 列表类型的元素访问需要索引，有正向递增和反向递减两种方式
- B、 集合类型的元素访问需要索引，有正向递增和反向递减两种方式
- C、 元组类型和列表类型都属于集合类型
- D、 字典类型是“键-值”数据项的组合，各键值对元素之间有先后关系

149、关于组合数据类型的描述，错误的选项是（ ）。

- A、 s是一个集合类型变量，则语句"x not in s"表示如果x不是s的元素，返回True，否则返回False
- B、 赋值语句animal = "cat", "dog", "tiger", "rabbit", 产生一个元组类型变量
- C、 s是一个集合类型变量，若s={235,10,50,10}，则执行print(s)的结果可能是{10,235,50}

D、s是一个列表类型变量，则语句"x is in s" 表示如果x是s的元素，返回True，否则返回False

150、关于列表类型的描述，正确的选项是（ ）。

- A、list(x)可以将一个整数x转换为列表[x]
- B、列表ls=[1,2,3,4,5]，S=ls[3:4]结果是4，结果是整数类型
- C、列表ls=[1,2,3,4,5]，S=ls[3:4]结果是[4]，结果是列表类型
- D、list({"a":1,"b":2})结果为["a",1, "b",2]

151、以下代码的执行结果是（ ）。

```
ls=["abc","abcdef","123","xyz"],["abc","xyz","123"],"abc"]
print(len(ls[1][-3]))
```

- A、 1
- B、 6
- C、 3
- D、 出错

152、以下代码的执行结果是（ ）。

```
ls=[]
for i in range(11):
    ls.append(i**2)
    ls.reverse()
print(ls)
```

- A、 [100,64,36,16,4,0,1,9,25,49,81]
- B、 [0,1,4,9,16,25,36,49,64,81,100]
- C、 [121,100,81,64,49,36,25,16,9,4,1]
- D、 [1,4,9,16,25,36,49,64,81,100,121]

153、以下代码将打印输出所有女生的名字:

```
d = {"张良":"男","李明明":"女","李华":"女","张丽":"女"}
for key in d:
    if ____ == "女":
        print(key)
```

补充完整横线处代码，正确的选项是（ ）。

- A、 d[key]
- B、 d.values(key)
- c、 get(key).value
- D、 d.value

154、文件data.csv里的内容如下：

zhang,17,5

wang,10,2

li,19,3

执行以下代码

```
f=open('data.csv', 'r')
```

```
print(f.readlines())
```

```
f.close()
```

关于执行结果的描述,正确的选项是（ ）。

- A、输出三行字符串
- B、输出三行列表，每行列表里面有一个字符串元素
- C、输出一行字符串，里面包括三个字符串
- D、输出一行列表，里面包括三个字符串元素

155、文件a.txt的内容如下：

[1,2,3,4]

以下代码的执行结果是（ ）。

```
f=open('a.txt','r')
```

```
print(f.read().split(","))
```

```
f.close()
```

- A、 '1','2','3','4'
- B、 [1,2,3,4]
- C、 ['1','2','3','4']
- D、 1,2,3,4

156、关于数据维度的描述，错误的选项是（ ）。

- A、一维数据由对等关系的有序或无序数据构成
- B、列表只能存储二维数据
- C、二维数据可以看成是多个一维数据的组合形式
- D、.txt格式文件可以存储一维数据和二维数据

157、turtle库中设置画笔宽度的函数是（ ）。

- A、 turtlesize()
- B、 pen()
- C、 write()
- D、 width()

158、显示pip命令帮助信息的命令格式是（ ）。

- A、pip -h
- B、pip search <拟查询关键字>
- C、pip install <拟安装库名>
- D、pip download <拟下载库名>

159、属于Python任务调度方向第三方库的选项是（ ）。

- A、scrapy
- B、doit
- C、pandas
- D、NLTK

160、属于Python网络爬虫方向第三方库的选项是（ ）。

- A、Click
- B、scikit-learn
- C、scrapy
- D、pytorch

161、关于Python语言特点的描述，错误的选项是（ ）。

- A、Python是一种脚本语言，经过编译后可多次执行
- B、同一个Python程序可以在不同的操作系统上执行
- C、Python的计算生态非常丰富
- D、Python语言在大数据和人工智能领域的应用很广泛

162、关于Python语言注释的描述，错误的选项是()。

- A、Python可以用井号#注释一行语句
- B、注释语句不能够被执行
- C、Python可以用一对三引号"注释多行语句
- D、Python可以使用缩进来表示注释

163、以下赋值语句，错误的选项是（ ）。

- A、a, b = b, c
- B、x == 0
- C、s = s.replace(':', '')
- D、b += 1

164、表达式'980'> '1000'的结果是()。

- A、1

- B、 False
- C、 True
- D、 0

165、表达式type(45)的结果是()。

- A、 None
- B、 <class 'str'>
- C、 <class 'float'>
- D、 <class 'int'>

166、 $x=2.6+4j$,表达式 $x.real+x.imag$ 的结果是()。

- A、 2.6
- B、 6.6
- C、 4.0
- D、 TypeError

167、变量 $s = 'shehuizhili'$ ，表达式 $s[-5:-1]$ 的结果是()。

- A、 zhil
- B、 zhili
- C、 izhi
- D、 izhil

168、执行结果为[23,56]的选项是()。

- A、 `print(eval([23,56]))`
- B、 `print(type(eval([23,56])))`
- C、 `print(eval('[23,56]'))`
- D、 `print(list('[23,56]'))`

169、关于Python分支结构的描述，正确的选项是 ()。

- A、
if 条件:
 语句块
- B、
if 条件
 语句块
- C、
if 条件
 语句块

else 条件

D、

if 条件:

语句块

else

170、以下代码的执行结果是()。

```
a = '3000'
```

```
if a < 2000:
```

```
    print("少")
```

```
elif a < 4000:
```

```
    print("不多")
```

```
else:
```

```
    print("还行")
```

A、还行

B、不多

C、少

D、执行出错

171、以下代码的执行结果是 ()。

```
m= 'BeetHoven MoonLight'
```

```
a = 0
```

```
b=0
```

```
for i in m:
```

```
    if 'A' <= i <= "Z":
```

```
        a += 1
```

```
    elif 'a' <= i <= 'z':
```

```
        b+=1
```

```
    else:
```

```
        break
```

```
print(a, b)
```

A、 2 14

B、 4 14

C、 2 7

D、 4 7

172、关于以下程序执行结果的描述，正确的选项是 ()。

try:

```
f=open('data.txt', 'w')
ts = f.write()
print('写文件正确')
```

except:

```
print('文件操作错误')
```

- A、输出"写文件正确"
- B、无论data.txt文件是否存在，都输出"文件操作错误"
- C、如果data.txt文件不存在，不输出"文件操作错误" "
- D、如果data.txt文件存在，不输出"文件操作错误"

173、关于循环结构的描述，错误的选项是（ ）。

- A、for i in range(4,n)中循环变量i的初值是4
- B、循环条件中的空列表或空字符串等价于False
- C、for i in 'Hello world'的循环次数是11次
- D、while循环结构的循环次数必须是确定的

174、以下代码的执行结果是()。

```
k = 5
def test(n):
    global k
    for i in range(n):
        k += i
    return k
print(k,test(5))
```

- A、程序运行错误
- B、5 15
- C、5 10
- D、15 15

175、关于Python函数的描述，正确的选项是（ ）。

- A、函数外部定义的列表变量，不可以在函数内部引用其值
- B、函数内部定义的数字变量，可以在函数之外引用其值
- C、函数有默认参数的时候，调用函数可以不给该参数赋值
- D、函数内部定义的列表变量，可以在函数之外引用其值

176、以下代码的执行结果是()。

```
ls = ['try']
def mtry(lt):
```

```
lt.append(ls)
return lt
print(mtry(mtry(['try'])))
A、 ['try', [ 'try'], ['try']]
B、 ['try' , 'try', 'try']
C、 try try try
D、 [['try'], ['try'], ['try']]
```

177、关于函数参数的描述，错误的选项是()。

- A、采用参数名称传递参数，可以增加程序的可读性
- B、函数调用时，默认按照位置顺序传递参数
- C、采用参数名称传递参数，可以不按照参数的默认顺序
- D、函数的可选参数可以放在非可选参数的前面

178、以下代码的执行结果是()。

```
x=[ [3,4,5],[[10,11],[12,13],14]]
print(len(x))
A、 4
B、 2
C、 3
D、 5
```

179、以下代码的执行结果是()。

```
L1=[1,2,3,4]
st=""
for i in L1:
    st +=i
print(st)
```

- A、 '1234'
- B、 10
- C、 1234
- D、 程序执行错误

180、以下代码的执行结果是()。

```
L1=["月亮","河","宽宽","的","河"]
print(L1.index("河"))
A、 1
B、 3
```

C、 2

D、 4

181、以下代码的执行结果是()。

```
dt = { 'a': 10, 'b': 20 }
```

```
print(type(dt['a']))
```

A、 <class 'tuple'>

B、 <class 'str'>

C、 <class 'int'>

D、 <class 'dict'>

182、在以下程序后执行选项中代码，报错的是()。

```
d=[{'作者': 'Jone', '密码': '*****', '收藏': []}, {'作品': [ '日出', '平凡人生' ]}]
```

A、 print(d[0]['作者'])

B、 print(d[0]['作者'][4])

C、 print(d[1]['作品'][0])

D、 print(d[1]['作品'][1])

183、关于文件的描述，错误的选项是()。

A、 readline()函数从文件中读入一行

B、 Python程序退出后，打开文件自动关闭，不一定需要close()函数

C、 open()函数的参数'r'表示对文件进行读操作

D、 open()函数的参数'a'表示只读文件

184、关于数据维度的描述，错误的选项是()。

A、 列表只能用来表示一维数据

B、 二维数据可以看作多个一维数据的组合形式

C、 字典可以表示高维数据

D、 CSV格式可以存储二维数据，每行使用英文半角逗号分隔各数据

185、不能返回列表类型的选项是()。

A、 s.split()

B、 jieba.lcut()

C、 jieba.cut()

D、 list()

186、

```
d=[(-0.71,0.6,1.0),(0.0,0.06,4.0),(0.71,0.6,8.0)]
```

for n in d:

print(-----)

填写空格，输出下述内容，

-0.71 0.0 0.71

正确的选项是 () 。

- A、'{}'.format(n[0]), end=' '
- B、'{}'.format(n[0])
- C、'{}'/n'.format(n[0])
- D、'{}\n' .format(n[0])

187、turtle库中，绘制圆弧的函数是()。

- A、seth()
- B、eclipse()
- C、forword()
- D、circle()

188、关于Python语言time标准库的描述，错误的选项是()。

- A、localtime()返回系统当前时间对应的struct_time形式
- B、localtime()返回系统当前时间对应的时间戳
- C、strftime()按照指定的格式返回易读字符串形式的时间
- D、mktime()将struct_time对象转换成时间戳

189、属于Python文本处理方向的第三方库是 ()。

- A、beautifulsoup4
- B、matplotlib
- C、vispy
- D、wxpython

190、属于Python网络爬虫方向的第三方库是 ()。

- A、myqr
- B、numpy
- C、scrapy
- D、pillow

191、关于 Python 程序异常处理的描述,错误的选项是()。

- A、try、except 等保留字提供异常处理功能
- B、程序发生异常后经过妥善处理可以继续执行
- C、Python 的异常和错误是完全相同的概念

D、异常语句可以与 else 和 finally 保留字配合使用

192、不是 Python 内置函数的选项是()。

- A、divmod()
- B、compare()
- C、ord()
- D、type()

193、关于全局变量和局部变量的描述,错误的选项是()。

- A、局部变量标识符不能与任何全局变量的标识符相同,即严格不能重名
- B、在函数内部用数字类型全局变量时,必须使用 global 保留字声明
- C、在函数内部用组合类型全局变量时,可以不通过 global 保留字声明
- D、全局变量在 Python 文件最外层声明时,语句前没有缩进

194、关于 Python 列表类型的描述,错误的选项是()。

- A、列表创建后可以修改其中元素,但每个元素类型不能修改
- B、列表类型的元素可以是列表
- C、二维数据可以用列表类型表示
- D、列表类型的元素可以是字典

195、以下代码的执行结果是()。

```
ls = ['中国', ['北京', '上海', '广州'], ['河北省', '浙江省', '广东省'], 300, 400, 500]  
print(ls[-4][1][:2])
```

- A、河北
- B、北京
- C、浙江
- D、浙江省

196、以下代码的执行结果是()。

```
s = ['well', 'good', 'best', 'how', 'do', 'you', 'do']  
str1 = s[3] + ' ' + s[4] + ' ' + s[5] + ' ' + s[6]  
print(str1)
```

- A、 do you do
- B、执行错误
- C、 howdoyou do
- D、 how do you do

197、属于 Python 图像处理方向的第三方库是 ()。

- A、matplotlib
- B、opencv-python
- C、scrapy
- D、wxpython

198、以下代码的执行结果是()。

```
a = "100"  
print(eval(a+"1+2"))
```

- A、100+1+2
- B、103
- C、1003
- D、执行出错

199、 以下代码的执行结果是()。

```
x=-4+3j  
y=4-3j  
print(x+y)
```

- A、0
- B、<class 'complex' >
- C、0j
- D、无输出

200、属于 Python 网络爬虫方向的第三方库是()。

- A、pytorch
- B、requests
- C、pillow
- D、NLT

小黑课堂计算机二级