

基本信息

编号ID: 1 得分: 120

开始时间: 2020-03-23 20:19:42 结束时间: 2020-03-23 20:26:48

120

答题详情

1. 下面关于Python描述错误的是:

- ☐ Python是一种高级编程语言
- ☐ Python是一种解释型语言
- ☐ Python文件的后缀名一般是.py
- ☒ Python版本2和版本3的代码可以通用 ✓

2. 下面关于Python描述错误的是:

- ☐ Python是一种高级编程语言
- ☐ Python文件的后缀名一般是.py
- ☒ Python是一种编译型语言 ✓
- ☐ Python版本2和版本3的代码是不通用的

3. 以下对 Python 程序缩进格式描述错误的选项是:

- ☐ 不需要缩进的代码顶行写, 前面不能留空白
- ☐ 缩进可以用 tab 键实现, 也可以用多个空格实现, 但应尽量保持一致
- ☐ 缩进主要用来表示程序的框架结构, 缩进可以嵌套
- ☒ 缩进是用来格式美化 Python 程序的 ✓

4. 以下对 Python 程序中的注释描述错误的选项是:

- ☐ 注释一般使用#开头, 或使用三个单引号包裹
- ☐ 三个单引号包裹的注释内部可以包括多行
- ☐ #开头的注释又称为单行注释, 不能跨行
- ☒ 注释不会被执行, 所以没有任何意义, 不应该为代码添加注释 ✓

5. 下面哪个变量名是合法的?

- ☐ nonlocal
- ☐ 3class
- ☐ student.name
- ☒ _name_ ✓

6.下面哪个变量名是合法的?

- ☐ global
- ☒ _str_ ✓
- ☐ 人口 . 数量
- ☐ class@school

7.下面哪个不是Python的关键字?

- ☐ True
- ☐ pass
- ☒ none ✓
- ☐ import

8.下面哪个不是Python的关键字?

- ☒ false ✓
- ☐ import
- ☐ None
- ☐ def

9.Python中，以下哪个函数是用来从控制台接收输入的?

- ☐ eval()
- ☒ input() ✓
- ☐ print()
- ☐ int()

10.Python中，以下哪个函数是用来向控制台输出内容的?

- ☐ input()
- ☐ output()
- ☒ print() ✓
- ☐ eval()

11.下面关于表达式结果类型说法错误的是:

- ☐ $4**2$ 的结果为整型int
- ☐ $4**-2$ 的结果类型为浮点型float
- ☐ $"123"*2$ 的结果是字符串str
- ☒ $9/3$ 的结果是整型int ✓

12.下面关于表达式结果类型说法错误的是:

- ☐ $4**2.0$ 的结果为浮点型float
- ☐ $5\%2$ 的结果类型为整型int
- ☒ $"123"*2$ 的结果是整数int ✓
- ☐ $6/2$ 的结果是浮点型float

13.Python表达式中,可以提升优先级改变计算顺序的是?

- ☒ 圆括号() ✓
- ☐ 方括号[]
- ☐ 大括号{}
- ☐ 上述都可以

14.Python表达式中,关于表达式计算优先级描述错误的是?

- ☐ 比较运算的优先级低于数值运算
- ☐ 布尔运算not的优先级高于or
- ☐ 数值运算中幂运算**的优先级高于乘法*
- ☒ 可以通过小括号()、中括号[]和大括号{}改变表达式的优先级顺序 ✓

15. $-7//2.0$ 的结果是:

- ☐ -3
- ☐ -3.0
- ☐ -4
- ☒ -4.0 ✓

16. $-4**2.0$ 的结果是

☐ 16.0

☐ 16

☒ -16.0

☐ -16

17. 下面关于表达式计算结果正确的为:

☐ $8\%3**2$ 的结果为4

☐ $3-6/2$ 的结果为0

☒ $-5//2*(3+1)$ 的结果为-12

☐ $12+-9/3*2$ 的结果为2.0

18. 下面关于表达式计算结果正确的为:

☐ $6-4*2$ 的结果为4

☐ $6-12/2$ 的结果为0

☐ $10// -3$ 的结果为-3.0

☒ $7\%9$ 的结果为7

19. 下面哪个语句在执行时会发生错误:

☐ $x = 4+-2*6e3$

☐ $x = \text{str}(8)*2$

☐ $x = \text{not } 3 \text{ not in } [1,2,3]$

☒ $x = \text{"abcd"}[4]+\text{"e"}$

20. 下面哪个语句在执行时会发生错误:

☐ $x = \text{int}(\text{"8"})+4$

☐ $x = -2*6e3$

☒ $x = [1,2,3] + \text{"456"}$

☐ $x = \text{"abcd"}[0]+\text{"e"}$

21.下面哪个布尔表达式的结果为True:

☐ 5.0<3 and 12**3 < 1628

☒ 3 not in (3,4,5) or "h" in "python" ✓

☐ not "123" + "4" == "1234"

☐ not 4+5<12 or 5/2!=2.5

22.下面哪个布尔表达式的结果为True:

☒ 5.0>=3 or 12**3 < 1628 ✓

☐ "h" in "python" and 6 not in (2,7,8-2)

☐ not 4+5<12 or 5//2==2.5

☐ "123"*2 != "123123"

23.表达式 (78<10) == False 的结果是:

☒ True ✓

☐ False

☐ None

☐ 运行错误

24.表达式 (78<10) != False 的结果是:

☐ True

☒ False ✓

☐ None

☐ 运行错误

25.下面代码的输出结果是什么：

```
x = 3
y = 5
x,y = y+1, x
z += x*y
print(x,y,z)
```

☐ 6 3 18

☐ 4 3 12

☐ 6 6 36

☒ 运行错误 

26.下面代码的输出结果是什么：

```
x = 3
y = 5
x,y = y+1, x-1
print(x,y)
```

☒ 6 2 

☐ 6 5

☐ 3 2

☐ 运行错误

27.len("pytho\n") 的结果是：

☐ 5

☒ 6 

☐ 7

☐ 8

28. len('xi\\'an') 的结果是:

- ☐ 4
- ☒ 5 
- ☐ 6
- ☐ 运行错误


29. 下面代码的输出结果是:

```
x = [1, 2, 3, 4]
y = [5, 6]
x.append(y)
print(len(x))
```


- ☐ 4
- ☒ 5 
- ☐ 6
- ☐ 7

30. 下面代码的输出结果是:

```
x = [1, 2, 3]
x.append( (4, 5) )
print(x)
```

- ☐ [1, 2, 3, 4, 5]
- ☐ [1, 2, 3, [4, 5]]
- ☒ [1, 2, 3, (4, 5)] 
- ☐ 运行错误

31. 已知 x = [1, 2, 3, 4, 5], 下面描述错误的是:

- ☐ len(x) 为 5
- ☐ x[3] 为 4
- ☐ x.index(2) 的结果为 1
- ☒ x.pop(5) 的结果为 5 

32. 已知 $x = [0, 1, 2, 3, 4]$, 下面描述错误的是:

☐ $\text{len}(x)$ 为 5

☒ $x[\text{len}(x)]$ 为 4 ✓

☐ $x[-1]$ 为 4

☐ $x[1]$ 为 1

33. 已知 $x = (1, 2, 3, 4, 5, 6, 7, 8)$, 则 $x[2:9:2]$ 为:

☐ 运行错误

☒ (3, 5, 7) ✓

☐ [3, 5, 7]

☐ [2, 4, 6, 8]

34. 已知 $x = (1, 2, 3, 4, 5, 6, 7, 8)$, 则 $x[-2:-6:-1]$ 为:

☒ (7, 6, 5, 4) ✓

☐ (7, 6, 5, 4, 3)

☐ [7, 6, 5, 4]

☐ [7, 6, 5, 4, 3]

35. 已知 $s = "0123456789"$, 以下表示 "0123" 的选项是

☒ $s[0:4]$ ✓

☐ $s[0:3]$

☐ $s[-10:-5]$

☐ $s[1:5]$

36. 已知 $s = "123456789"$, 以下表示 "123" 的选项是

☒ $s[0:3]$ ✓

☐ $s[0:4]$

☐ $s[1:3]$

☐ $s[1:4]$

37. 已知 $x = \text{"abcde"}$, 则 $x[-1:0:-1]$ 为:

☐ "edcba"

☒ "edcb" ✓

☐ "dcb"

☐ "dcba"

38. 已知 $x = \text{"abcde"}$, 则 $x[::-1]$ 为:

☐ "abcde"

☐ ""

☒ "edcba" ✓

☐ 运行错误

39. 已知 $x = (1, 2, 3, [4, 5, 6], \text{"abc"})$, 下面描述错误的是:

☐ $x[3][1]$ 为 5

☐ $x[4]$ 为 "abc"

☐ 执行 $x[3][2] = 9$ 不会出现错误

☒ 执行 $x[4] = 9$ 不会出现错误 ✓

40. 已知 $x = [1, 2, 3, (4, 5, 6), \text{"abc"}]$, 下面描述错误的是:

☐ $x[3][2]$ 为 6

☐ $x[4]$ 为 "abc"

☒ 执行 $x[3][1] = 9$ 不会出现错误 ✓

☐ 执行 $x[4] = 9$ 不会出现错误

41. 已知 $x = \{1:\text{"one"}, 2:\text{"two"}, 3:\text{"three"}\}$, 则 $x[1]$ 的值为:

☐ 1

☐ 2

☒ "one" ✓

☐ "two"

42. 已知 $x = \{1:10, 2:20, 3:30\}$ ，则 $x[1]$ 的值为：

☐ 1

☐ 2

☒ 10 ✓

☐ 20

43. 下面哪个说法是错误的？

☐ 元组中的元素是不能修改的

☐ 对列表进行切片，结果一定是列表

☒ 字典中可以使用元组或列表作为键 ✓

☐ 元组中的元素可以是列表或字典

44. 下面哪个说法是错误的？

☐ 对列表进行切片，结果一定是列表

☐ 对元组进行索引，结果有可能是列表

☐ 字典不能使用列表类型作为它的键

☒ 字典一旦定义，就不能对其中的元素进行修改 ✓

45. 下面哪个表达式的结果是"张三今年18岁"？

☐ "{0}今年{1}岁".format(18, "张三")

☐ "{1}今年{0}岁".format("张三", 18)

☐ "张三今年" + 18 + "岁"

☒ "张三今年" + str(18) + "岁" ✓

46. 已知 $x, y = 3, 4$ ，下面哪个表达式的结果不是 $"3+4=7"$ ？

☐ "{0}+{1}={2}".format(x,y,x+y)

☐ "{1}+{0}={2}".format(y,x,x+y)

☐ str(x) + "+" + str(y) + "=" + str(x+y)

☒ x + "+" + y + "=" + (x+y) ✓

47.下面程序的运行结果是什么?

```
x = (1,2,3,4)
for item in x:
    item = 9
print(x)
```

☐ [1,2,3,4]

☒ (1,2,3,4) ✓

☐ (9,9,9,9)

☐ 程序运行错误

48.下面程序的运行结果是什么?

```
x = [1,2,3,4]
for item in x:
    item = 9
print(x)
```

☒ [1, 2, 3, 4] ✓

☐ [9, 9, 9, 9]

☐ (9, 9, 9, 9)

☐ 程序运行错误

49.下面程序的运行结果是什么？

```
ls = []  
for i in range(5):  
    ls.insert(0, i)  
print(ls)
```

☐ [0,1,2,3,4]

☒ [4,3,2,1,0] ✓

☐ [1,2,3,4,5]

☐ [5,4,3,2,1]

50.下面程序的运行结果是什么？

```
ls = [9,99]  
for i in range(5):  
    ls.insert(1, i)  
print(ls)
```

☐ [9, 0, 1, 2, 3, 4, 99]

☐ [9, 1, 2, 3, 4, 5, 99]

☐ [0, 1, 2, 3, 4, 9, 99]

☒ [9, 4, 3, 2, 1, 0, 99] ✓

51.下面程序的运行结果是什么?

```
ls = [ ]  
for i in range(10):  
    if i%2==0:  
        continue  
    ls.append(i)  
print(ls)
```

☐ [0,2,4,6,8]

☐ [0,2,4,6,8,10]

☒ [1,3,5,7,9] ✓

☐ []

52.下面程序的运行结果是什么?

```
ls = [ ]  
for i in range(10):  
    if i%2==1:  
        continue  
    ls.append(i)  
print(ls)
```

☒ [0, 2, 4, 6, 8] ✓

☐ [1, 3, 5, 7, 9]

☐ [8, 6, 4, 2, 0]

☐ [10, 8, 6, 4, 2, 0]

53.下面程序的运行结果是什么？

```
ls = [ ]  
for i in range(10):  
    if i%2==1:  
        break  
    ls.append(i)  
print(ls)
```

☐ [1,3,5,7,9]

☒ [0] ✓

☐ [0,2,4,6,8]

☐ [0,2,4,6,8,10]

54.下面程序的运行结果是什么？

```
ls = [ ]  
for i in range(10):  
    if i%2==0:  
        break  
    ls.append(i)  
print(ls)
```

☒ [] ✓

☐ [1, 3, 5, 7, 9]

☐ [0, 2, 4, 6, 8]

☐ [0, 2, 4, 6, 8, 10]

55.下面程序的运行结果是什么？

```
score = 79
if score > 85:
    print("优秀")
elif score > 60:
    print("及格")
elif score > 75:
    print("良好")
else:
    print("不及格")
```

☐ 不及格

☐ 良好

☒ 及格 ✓

☐ 优秀

56.下面程序的运行结果是什么？

```
score = 79
grade = ""
if score > 85:
    grade = "优秀"
if score > 75:
    grade = "良好"
if score > 60:
    grade = "及格"
else:
    grade = "不及格"
print(grade)
```

☐ 优秀

☐ 良好

☒ 及格 ✓

☐ 不及格

57.下面程序的运行结果是什么?

```
s = ""
for x in "abcd":
    for i in range(3):
        if x == "b":
            break
        s += x
print(s)
```

☐ abcdabcdnab

☐ aaabbbcccddd

☒ aaacccddd ✓

☐ aaa

58.下面程序的运行结果是什么?

```
s = ""
for x in "abcd":
    for i in range(3):
        s += x
        if x == "b":
            break
print(s)
```

☐ aaabbbcccddd

☒ aaabcccddd ✓

☐ aaab

☐ aaa

59.下面程序的运行结果是什么？

```
r = 1
area = 3.1415926 * r ** 2
print("{:.4f}".format(area))
```

☐ 3.14

☐ 3.141

☐ 3.1415

☒ 3.1416 ✓

60.下面代码的输出结果是什么：

```
r = 1
area = 3.1415926 * r ** 2
print("{:.3f}".format(area))
```

☐ 3.1

☐ 3.14

☐ 3.141

☒ 3.142 ✓