# python AI助教项目训练题库

*1=单选题，2=判断题*

1

【1】以下代码段的输出是什么？

x = 10

y = 5

if x > y:

if x < 20:

print("A")

elif x == 10:

print("B")

else:

print("C")

else:

print("D")

A. "A"

B. "B"

C. "C"

D. "D"

A

1

【2】以下哪个选项正确地表示了if-else语句的语法？

A. if condition:\nstatement\nelse:\nstatement

B. if condition:\nstatement;\nelse:\nstatement

C. if condition\n{ statement }\nelse\n{ statement }

D. if (condition)\n{ statement; }\nelse\n{ statement; }

A

1

【3】如果要检查变量 x 是否等于 10 或者 y 是否等于 20，应该使用哪个语法？

A. if x = 10 or y = 20:

B. if x == 10 or y == 20:

C. if x == 10 || y == 20:

D. if x = 10 || y = 20:

B

1

【4】为什么在编写条件语句时，应尽量避免过度嵌套？

A. 因为会降低代码的运行速度

B. 因为会增加内存消耗

C. 因为会使代码难以理解和维护

D. 因为Python不支持超过两层的嵌套

C

1

【5】如果 x = 3 和 y = 4，下面代码的输出是什么？

if x \* y > 10:\n print("A")\nelif x + y < 10:\n print("B")\nelse:\n print("C")

A. A

B. B

C. C

D. 无

A

1

【6】在以下代码中，最终 y 的值是多少？

x = 7\ny = 10\nif x > 5:\n y += 5\nif x < 10:\n y += 10

A. 15

B. 20

C. 25

D. 30

C

1

【7】若检查一个变量 x 是否在1 到 10 之间，不能选哪个选项？

A. if x >=1 and <= 10:

B. if x >= 1 and x <= 10:

C. if 1<= x <= 10:

D. if 10 >= x >= 1:

A

1

【8】if-elif-else 语句中，哪种情况下会执行 else 语句？

A. 当 if 条件为 True

B. 当所有 if 和 elif 条件均为 False

C. 当第一个 if 条件为 False 时

D. 当 elif 条件为 True 时

B

1

【9】在使用 if-elif-else 语句时，如何确保某个特定条件优先被评估？

A. 将该条件放在 if 语句中

B. 将该条件放在 elif 语句的最前面

C. 在 if 和 elif 语句中同时使用该条件

D. 使用嵌套的 if 语句来优先评估该条件

A

1

【10】下列选项中，能求出x和y中最小值，并赋值给min的是\_\_\_\_\_。

A. min = x if x < y else y

B. min = x < y ? x : y

C. if x < y: min=x

D. if (x<y): x, y = y, x

A

1

【11】对于两个集合s1和s2，s1 < s2的意思是？

A. s1的大小小于s2的大小

B. s1的元素比s2的小

C. s1是s2的真子集

D. s2是s1的真子集

C

1

【12】对于集合s，以下哪个操作是不存在的？

A. len(s)

B. s.append(1)

C. max(s)

D. s - {1}

B

1

【13】对于正确的表达式a[2]，a不可能是以下哪个类型？

A. 集合

B. 列表

C. 元组

D. 字典

A

1

【14】可以使用\_\_\_\_运算符来确定一个键是否在字典中。

A. &

B. in

C. ^

D. ?

B

1

【15】你可以使用\_\_\_\_从字典中删除元素。

A. remove

B. rease

C. delete

D. del

D

1

【16】返回集合中元素个数的函数是\_\_\_\_\_\_。

A. size()

B. len()

C. elements()

D. count()

B

1

【17】以下哪个语句得到的 C 不是集合 A 和集合 B 的并集？

A. C = A | B

B. C = A.update(B)

C. C = set(list(A) + list(B))

D. C = A.union(B)

B

1

【18】下面那个不是Python可以接受的变量名？

A. abc

B. \_23ac

C. i

D. good-name

D

1

【19】输入10，下面程序行号为2的语句输出是多少？

for number in range(1,10):\n if number%2 and not number%3:\n the\_sum += number

A. 12

B. 7

C. 13

D. 8

A

1

【20】下面定义字典的语句那个是正确的？

A. momthdays=dict(Jan=31,Feb=28,Mar=31,Apr=30)

B. momthdays=dict("Jan"=31,"Feb"=28,"Mar"=31,“Apr"=30)

C. momthdays={Jan:31,Feb:28,Mar:31,Apr:30}

D. momthdays={Jan=31,Feb=28,Mar=31,Apr=30}

A

1

【21】下列程序运行输出结果为\_\_\_\_\_\_\_。

递归因式分解 18 -> 3 6 -> 2 3，输出：3 2 3

A. 3 2 3

B. 2 9 3

C. 3 6 3

D. 2 3 6

B

1

【22】选择下面程序的运行结果

number = 25，25%5==0，i=5，isPrime=False

A. i is 5 isPrime is True

B. i is 5 isPrime is False

C. i is 6 isPrime is True

D. i is 6 isPrime is False

B

1

【23】下列程序运行输出结果为\_\_\_\_\_\_\_。

day=4，x=(x+1)\\*2 → 4 次，最终 x=46

A. 22

B. 46

C. 47

D. 45

B

1

【24】下列程序运行输出结果为\_\_\_\_\_\_\_。

函数内 b=1 局部，乘全局 c=4 得 b=4，函数外打印全局 b=2，c=4

A. 4 4 2 4

B. 4 4 4 4

C. 4 2 4 4

D. 2 4 2 4

A

1

【25】对于两个集合s1和s2，s1 < s2的意思是？

A. s1的大小小于s2的大小

B. s1的元素比s2的小

C. s1是s2的真子集

D. s2是s1的真子集

C

1

【26】对于集合s，以下哪个操作是不存在的？

A. len(s)

B. s.append(1)

C. max(s)

D. s - {1}

B

1

【27】对于正确的表达式a[2]，a不可能是以下哪个类型？

A. 集合

B. 列表

C. 元组

D. 字典

A

1

【28】可以使用\_\_\_\_运算符来确定一个键是否在字典中。

A. &

B. in

C. ^

D. ?

B

1

【29】你可以使用\_\_\_\_从字典中删除元素。

A. remove

B. rease

C. delete

D. del

D

1

【30】返回集合中元素个数的函数是\_\_\_\_\_\_。

A. size()

B. len()

C. elements()

D. count()

B

1

【31】下列程序运行输出结果为\_\_\_\_\_\_\_。

键 '2' 为字符串，2 是整数，不存在，返回 -1

A. 'B'

B. 'A'

C. None

D. -1

D

1

【32】以下程序执行的结果是()

遍历字典默认是遍历键：'a', 'c'

A. a:b c:d

B. a c

C. b d

D. 以上都不对

B

2

【1】第三方模块要先安装才能使用。

True / False

True

2

【2】len(set([0,4,5,6,0,7,8]))的结果是7。

True / False

False

2

【3】字符串可以使用 max() 和 min() 函数进行比较大小。

True / False

True

2

【4】对于元组 t = (1, 2, {3: 4})，执行 t[2][0] = 5 会报错。

True / False

False

2

【5】列表的 sort() 方法返回一个新的已排序列表。

True / False

False

2

【6】执行python语句 a=int("9") 后，此刻变量 a代表的是整数类型的数据。

True / False

True

2

【7】该语句可以正确创立一个字典：monthdays=dict(Jan=31,Feb=28,Mar=31,Apr=30)。

True / False

True

2

【8】在 Python 中，实例方法必须包含 self 参数。

True / False

True

2

【9】{(1,2,[1]),10,'abc'}是一个合法的python集合。

True / False

False

2

【10】执行以下代码， numbers 的内容将变成[1, 3, 2, 4]。

numbers = [1, 2, 3, 2, 4]

numbers.pop(1)

True / False

True

2

【11】集合的元素可以是任意数据类型。

True / False

False

2

【12】a={},type(a)结果是<class 'set'>。

True / False

False

2

【13】下面程序的运行结果一定是：1 2 3 4

set1={1,2,3,4}

for i in set1:

print(i,end=" ")

True / False

False

2

【14】列表可以作为字典的键。

True / False

False

2

【15】下面程序的输出是张秀华。

dic={"赵洁" : 15264771766,"张秀华" : 13063767486,"胡桂珍" : 15146046882,"龚丽丽" : 13606379542,"岳瑜" : 13611987725}

reversedic={v:k for k,v in dic.items()}

print(reversedic[13063767486])

True / False

True

2

【16】下面程序输出的是True。

dic1={"赵洁" : 15264771766,"张秀华" : 13063767486,"胡桂珍" : 15146046882,"龚丽丽" : 13606379542,"岳瑜" : 13611987725}

dic2={"王玉兰" : 15619397270,"王强" : 15929494512,"王桂荣" : 13794876998,"邓玉英" : 18890393268,"何小红" : 13292597821}

dic3={\*\*dic1,\*\*dic2}

dic1.update(dic2)

print(dic1==dic3)

True / False

True

2

【17】下面程序输出的是15146046882。

dic1={"赵洁" : 15264771766,"张秀华" : 13063767486,"胡桂珍" : 15146046882,"龚丽丽" : 13606379542,"岳瑜" : 13611987725}

dic2={"王玉兰" : 15619397270,"王强" : 15929494512,"王桂荣" : 13794876998,"邓玉英" : 18890393268,"胡桂珍" : 13292597821}

dic3={\*\*dic1,\*\*dic2}

print(dic3["胡桂珍"])

True / False

False

2

【18】下面程序的输出是3。

s={1,2,3}

s1=s

s1.remove(2)

print(len(s))

True / False

False

2

【19】下面程序输出是circle。

d={"color":"red","shape":"circle"}

d1=d

d1["shape"]="triangle"

print("shape")

True / False

False

2

【20】嵌套if语句中的else分支总是与最近的未闭合的if语句匹配。

True / False

True

2

【21】在Python中，if语句的条件表达式可以是一个复杂的表达式，包括函数调用。

True / False

True

2

【22】在Python中，如果if语句的条件为一个空字符串，条件会被解释为True。

True / False

False

2

【23】在嵌套if语句中，内层的if条件表达式无法访问外层的变量。

True / False

False

2

【24】在if语句中，and运算符的优先级低于or运算符。

True / False

False

2

【25】Python的if-elif-else语句中，elif分支的数量是可以任意多个的。

True / False

True

2

【26】在Python中，if语句的条件表达式必须是一个布尔值。

True / False

False

2

【27】Python中的if-else语句可以嵌套在单行代码中，使用三元运算符实现。

True / False

True

2

【28】如下两段代码功能一样，程序结构也一样

if a < 0:

print("<0")

elif a == 0:

print("=0")

elif a > 0:

print("a>0")

if a < 0:

print("<0")

if a == 0:

print("=0")

if a > 0:

print("a>0")

True / False

False

2

【29】在if-elif-else结构中，else子句是可选的。

True / False

True