Python 试读

来自【机器学习面试题汇总与解析(蒋豆芽面试题总结)】 | 185 浏览 | 3 回复 | 2021-07-02



蒋豆芽 👜



机器学习面试题汇总与解析——Python

- 1. python 深拷贝与浅拷贝 ☆ ☆ ☆ ☆
- 2. **python多线程能用多个cpu么?** ☆ ☆ ☆ ☆ ☆
- 3. **python垃圾回收机制** ☆ ☆ ☆ ☆
- 4. python里的生成器是什么 \diamondsuit \diamondsuit \diamondsuit \diamondsuit
- 5. **迭代器和生成器的区别** \diamondsuit \diamondsuit \diamondsuit \diamondsuit
- 6. 装饰器 ☆ ☆ ☆ ☆ ☆
- 7. **python有哪些数据类型** ☆ ☆ ☆ ☆
- 8. Python 中列表 (List) 中的 del, remove, 和 pop 等的用法和区别 ☆ ☆ ☆ ☆
- 9. python yeild 和return的区别 ☆ ☆ ☆ ☆
- 10. **python set底层实现** \diamondsuit \diamondsuit \diamondsuit \diamondsuit
- 11. **python字典和set()的区别** ☆ ☆ ☆ ☆
- 12. **怎么对字典的值进行排序?** ☆ ☆ ☆ ☆
- 13. init和new和call的区别 ☆ ☆ ☆ ☆ ☆
- 14. **import常用库**☆ ☆ ☆
- 15. python的lamda函数 🏠 🏠 🏠 🗘
- 16. **Python内存管理**☆ ☆ ☆ ☆ ☆
- 17. **python在内存上做了哪些优化?** ☆ ☆ ☆ ☆ ☆
- 18. **Python中类方法和静态方法的区别** \diamondsuit \diamondsuit \diamondsuit \diamondsuit
- 19. **python多线程怎么实现** ☆ ☆ ☆ ☆
- 20. 点积和矩阵相乘的区别? ☆ ☆ ☆ ☆
- 21. **Python中错误和异常处理** ☆ ☆ ☆ ☆
- 22. **Python 的传参是传值还是传址?** ☆ ☆ ☆ ☆

- 25. **Python 中的 is 和 == 有什么区别?** ☆ ☆ ☆ ☆
- 26. **gbk和utf8的区别** ☆ ☆ ☆ ☆
- 27. 遍历字典可以用什么方法 ☆ ☆ ☆
- 28. **反转列表的方法** \diamondsuit \diamondsuit \diamondsuit
- 29. **python 元组中元组转为字典** ☆ ☆ ☆
- 30. range在python2和python3里的区别 ☆ ☆ ☆ ☆
- 31. __init__. py **文件的作用以及意义** ☆ ☆ ☆

- 本专栏适合于Python已经入门的学生或人士,有一定的编程基础。
- 本专栏适合于**算法工程师、机器学习、图像处理求职**的学生或人士。
- 本专栏针对面试题答案进行了**优化,尽量做到好记、言简意赅。这才是一份面试题总结的正确打开** 方式。这样才方便背诵
- 如专栏内容有错漏,欢迎在评论区指出或私聊我更改,一起学习,共同进步。
- 相信大家都有着高尚的灵魂,请尊重我的知识产权,未经允许严禁各类机构和个人转载、传阅本专栏的内容。

关于Python的知识点,推荐一个网址: https://github.com/hantmac/Python-Interview-Customs-Collection

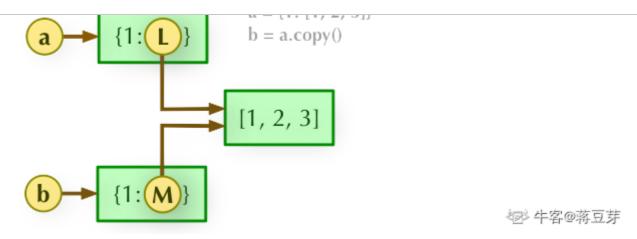
1. **python 深拷贝与浅拷贝** ☆ ☆ ☆ ☆

参考回答

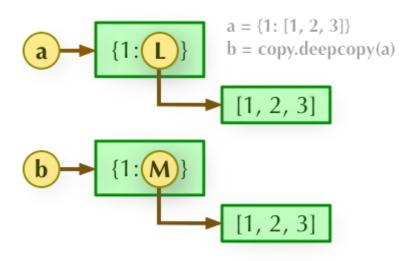
浅拷贝(copy):拷贝父对象,不会拷贝对象的内部的子对象。

深拷贝(deepcopy): copy 模块的 deepcopy 方法,完全拷贝了父对象及其子对象。

答案解析



b = copy.deepcopy(a): 深度拷贝, a 和 b 完全拷贝了父对象及其子对象,两者是完全独立的。



₩ 牛客@落豆芽

类似的问题还有:

2. **python多线程能用多个cpu么?** ☆ ☆ ☆ ☆ ☆

参考回答

python的多线程不能利用多核CPU

原因是python的解释器使用了GIL (Global Interpreter Lock),在任意时刻中只允许单个python线程运行。无论系统有多少个CPU核心,python程序都只能在一个CPU上运行。

答案解析

GIL 中文译为全局解释器锁,其本质上类似操作系统的 Mutex。GIL 的功能是:在 CPython 解释器中执行的每一个 Python 线程,都会先锁住自己,以阻止别的线程执行。

3. **python垃圾回收机制** ☆ ☆ ☆ ☆

参考回答

0,则Python垃圾回收机制会自动清除此对象。

方式2:标记-清除 (Mark—Sweep): 被分配对象的计数值与被释放对象的计数值之间的差异累计超过某个阈值,则Python的收集机制就启动

方式3: 当代码中主动执行 gc.collect() 命令时, Python解释器就会进行垃圾回收。

答案解析

无。

4. **python里的生成器是什么**☆ ☆ ☆ ☆ ☆

参考回答

生成器是一种可以简单有效的创建**迭代器**的工具。它们像常规函数一样撰写,但是在需要返回数据时使用**yield语句**。每当对它调用next()函数,生成器从它**上次停止的地方重新开始**(它会记住所有的数据值和上次执行的语句)。

答案解析

无。

5. **迭代器和生成器的区别** \diamondsuit \diamondsuit \diamondsuit \diamondsuit

参考回答

迭代器有两个方法next方法和iter方法,iter方法获取对象的迭代器,next方法返回下一个迭代器。

生成器:使用了 yield 的函数被称为生成器 (generator),在调用生成器运行的过程中,每次遇到 yield 时函数会暂停并保存当前所有的运行信息,返回 yield 的值,并在下一次执行 next() 方法时从当前位置继续运行。调用一个生成器函数,返回的是一个迭代器对象。

区别

- 1. **语法上**:生成器是通过函数的形式中调用 yield 或 ()的形式创建的;迭代器可以通过 iter () 内置函数创建
- 2. **用法上**:生成器在调用next()函数或for循环中,所有过程被执行,**且返回值**;迭代器在调用next()函数或for循环中,所有值被返回,没有其他过程或说动作。

答案解析

无。

参考回答

装饰器可以实现,在不改变函数或类的内部代码与调用方式的情况下,使其拥有额外的功能。

应用场景:插入日志、性能测试、事务处理、缓存、权限校验

答案解析

无。

7. python有哪些数据类型 \diamondsuit \diamondsuit \diamondsuit \diamondsuit

参考回答

数字,字符串,列表,元组,字典,集合

答案解析

无。

8. Python 中列表 (List) 中的 del, remove, 和 pop 等的用法和区别 ☆ ☆ ☆ ☆ ☆ ☆ 参考回答

1. pop

value = List.pop(index)

- 1、pop按照索引位置删除元素; 2、无参数时默认删除最后一个元素; 3、返回删除的元素值
- 2. remove
 - 1、remove 按照值删除,删除单个元素; 2、删除首个符合条件的元素; 3、返回值为空 None
- 3. **del**

根据索引位置来删除单个值或指定范围内的值。

del是删除引用(变量)而不是删除对象(数据),对象由自动垃圾回收机制(GC)删除

答案解析

无。

9. python yeild 和return的区别 ☆ ☆ ☆ ☆ ☆

的。

不同点:如果要返回的数据是通过for等循环生成的迭代器类型数据(如列表、元组),return只能在循环外部一次性地返回,yeild则可以在循环内部逐个元素返回。

答案解析

无。

10. **python set底层实现** ☆ ☆ ☆ ☆

参考回答

散列表。

答案解析

无。

11. **python字典和set()的区别** ☆ ☆ ☆ ☆

参考回答

- 1. 字典是一系列无序的键值对的组合;集合里的元素不是键值对,是单一的一个元素。
- 2. 字典无序; 集合有序
- 3. 字典键不能重复;集合元素不能重复

答案解析

无。

12. **怎么对字典的值进行排序?** $\Diamond \Diamond \Diamond \Diamond \Diamond$

参考回答

方法一:转化为元组, (91, "张三")的形式, 用sorted()函数进行排序

方法二: 设置sorted() 中key的参数的值

答案解析

- 1 #-*-coding:UTF-8-*-
- 2 #先生成一个随机字典

```
#方法一: 元组形式
7
    #把值放前面是因为sorted()排序默认按从头比较
8
    z = zip(d.values(),d.keys())
9
    #x = zip(d.itervalues(),d.iterkeys()) 迭代类型,省空间
    #排序
10
11
    s = sorted(z)
12
    print(s)
13
   #方法二: sorted(key的设置)
14
15
   a = d.items()
   s = sorted(a,key= lambda x:x[1])
16
17
   print(s)
```

13. __init__和__new__和__call__的区别 ☆ ☆ ☆ ☆ ☆

参考回答

__init__是初始化方法,

new 实例化对象

 $__{call}$ _:允许一个类的实例像函数一样被调用。实质上说,这意味着 x() 与 $x._{call}()$ 是相同的

答案解析

无。

14. **import常用库** ☆ ☆ ☆

参考回答

time, random, sys, os, math

答案解析

无。

15. python的lamda函数 🌣 🗘 🗘 🗘

参考回答

它是一个单独的表达式匿名函数,通常用作内联函数

答案解析

Python Lambda函数也称为丢弃函数,它可以与其他预定义函数(如filter(),map()等)一起使用。相对于我们定义的可重复使用的函数来说,这个函数更加简单便捷。

16. **Python内存管理** ☆ ☆ ☆ ☆

参考回答

Python的内存管理机制可以从三个方面来讲

- 1. 垃圾回收
- 2. 引用计数
- 3. 内存池机制

答案解析

无。

17. **python在内存上做了哪些优化?** ☆ ☆ ☆ ☆ ☆

参考回答

Python的内存优化可以从三个方面来讲

- 1. 垃圾回收
- 2. 引用计数
- 3. 内存池机制

答案解析

无。

18. **Python中类方法和静态方法的区别** ☆ ☆ ☆ ☆

参考回答

Python **类方法**和实例方法相似,它最少也要包含一个参数,只不过类方法中通常将其命名为 cls, Python 会自动将类本身绑定给 cls 参数 (注意, 绑定的不是类对象)。**也就是说, 我们在调用类方法时, 无需显式为 cls 参数传参**。

静态方法没有类似 self、cls 这样的特殊参数,因此 Python 解释器不会对它包含的参数做任何类或对象的绑定。也正因为如此,**类的静态方法中无法调用任何类属性和类方法**。

19. **python多线程怎么实现** ☆ ☆ ☆ ☆

参考回答

Python 提供多线程编程的模块有以下几个:

- thread
- threading
- Queue
- multiprocessing

答案解析

无。

20. 点积和矩阵相乘的区别? ☆ ☆ ☆ ☆

参考回答

<stron>: 维度完全一致的两个向量相乘,得到一个标量</stron>

矩阵相乘: X*N N*Y==>X*Y, 得到的依然是一个矩阵

答案解析

无。

21. **Python中错误和异常处理** ☆ ☆ ☆

参考回答

略

答案解析

参考文章: https://www.cnblogs.com/myblou/p/14154391.html

22. **Python 的传参是传值还是传址?** ☆ ☆ ☆ ☆

参考回答

答案解析

无。

23. **什么是猴子补丁?** ☆ ☆ ☆ ☆

参考回答

所谓的猴子补丁的含义是指在动态语言中,不去改变源码而对功能进行追加和变更

答案解析

无。

24. **当退出 Python 时是否释放所有内存分配?** ☆ ☆ ☆ ☆

参考回答

答案是否定的。那些具有对象循环引用或者全局命名空间引用的变量,在 Python 退出是往往不会被释放。另外不会释放 C 库保留的部分内容。

答案解析

无。

25. **Python 中的 is 和 == 有什么区别?** ☆ ☆ ☆ ☆

参考回答

is比较的是两个对象的id值是否相等,也就是比较两个对象是否为同一个实例对象,是否**指向同一个内存地址。**

==比较的是两个对象的**内容**是否相等,默认会调用对象的eq()方法。

答案解析

无。

26. **gbk和utf8的区别** ☆ ☆ ☆ ☆

参考回答

GBK是在国家标准GB2312基础上扩容后兼容GB2312的标准。GBK编码专门用来解决中文编码的,是双字节的。不论中英文都是双字节的。

2021/7/12 Python 牛客博客

:■ 蒋豆芽

国人访问你的GBK网页,需要下载中文语言包支持。访问UTF-8编码的网页则不出现这问题。可以直接访问。

GBK包含全部中文字符; UTF-8则包含全世界所有国家需要用到的字符。

答案解析

无。

27. **遍历字典可以用什么方法** \diamondsuit \diamondsuit \diamondsuit

参考回答

```
1
   dic1 = {'date':'2018.11.2', 'name':'carlber', 'work':"遍历", 'number':3}
   for i in dic1:
3
      print(i)
4
   for key in dic1.keys(): #遍历字典中的键
5
      print(key)
   for value in dic1.values(): #遍历字典中的值
6
7
      print(value)
8
   for item in dic1.items(): #遍历字典中的元素
9
      print(item)
```

答案解析

无。

28. 反转列表的方法 ☆ ☆ ☆ ☆

参考回答

- 1. Reverse()
- 2. sorted()指定reverse

答案解析

无。

29. **python 元组中元组转为字典** ☆ ☆ ☆ ☆

参考回答

```
#create a tuple
tuplex = ((2, "w"),(3, "r"))
print(dict((y, x) for x, y in tuplex))
```

无。

30. range在python2和python3里的区别 ☆ ☆ ☆ ☆

参考回答

在py2中, range得到的是一个列表,

在py3中, range得到的是一个生成器, 节约内存

答案解析

无。

31. __init__. py **文件的作用以及意义** ☆ ☆ ☆

参考回答

这个文件定义了包的属性和方法,它可以什么也不定义;可以只是一个空文件,但是必须存在。如果 __init__.py 不存在,这个目录就仅仅是一个目录,而不是一个包,它就不能被导入或者包含其它的模块和嵌套包。

或者可以这样理解。这样,当我们导入这个包的时候,__init__.py文件自动运行。帮我们导入了这么多个模块,我们就不需要将所有的import语句写在一个文件里了,也可以减少代码量。

答案解析

无。

31. **Python列表去重复元素** ☆ ☆ ☆ ☆

参考回答

答案解析

无。

资源分享 python 机器学习 算法工程师 春秋招 面试题 软件开发 面经

1X.IIIS 贝

相关专栏



机器学习面试题汇总与解析 (蒋豆芽面试题总结)

27篇文章 90订阅

已订阅

3条评论

○↑ 默认排序 ~



蒋豆芽 N

2021-07-10 07:34:35

多多点赞, 会变好看, 多多留言, 会变有钱。



蒋豆芽 N

② 0 **心** 0 2#

⑤ 0 ₺ 0 1#

2021-07-10 07:35:03

大家有任何问题,都可以提出来,我们共同完善专栏



蒋豆芽 N

⑤ 0 ₺ 0 3#

2021-07-10 07:35:31

550681844,大家可以加入这个群,里面有学习资料下载,希望能帮到大家。问题答案:牛客网

请留下你的观点吧~

发布

/ 牛客博客, 记录你的成长

关于博客 意见反馈 免责声明 牛客网首页