

我们检测到你可能使用了 Adblock 或 Adblock Plus，它的部分策略可能会影响到正常功能的使用（如关注）。
你可以设定特殊规则或将知乎加入白名单，以便我们更好地提供服务。（为什么？）



常见面试题整理--Python代码篇



常见面试题整理--Python代码篇

赞同 692



分享

路人甲

路人甲

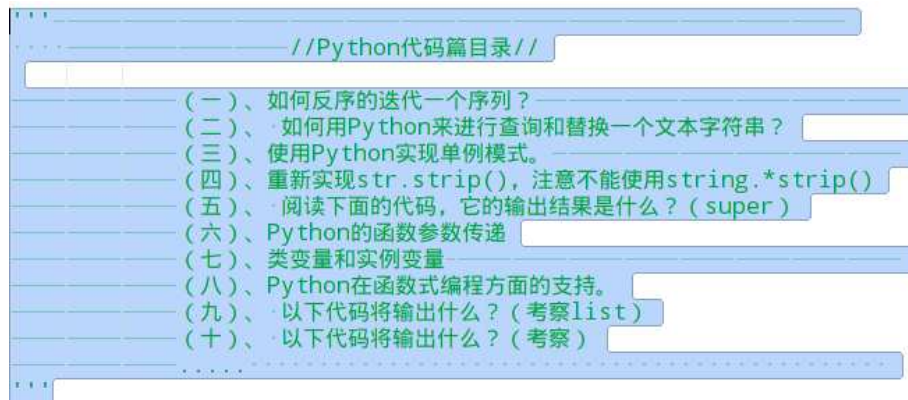
喜欢用数据讲故事，微信公众号：一个程序员的日常，合作请加微信：904727147

关注他

692 人赞了该文章

希望此文可以长期更新并作为一篇Python的面试宝典。每一道题目都附有详细解答，以及更加详细的回答链接。此篇是代码篇，上一篇是：[Python概念篇](#) 两者结合效果更佳。

喜欢听音乐的猿们，欢迎关注我最近新开的专栏：[音乐推荐-知乎专栏](#)



(一)、如何反序的迭代一个序列?

```
192 #如果是一个list, 最快的解决方案是
193 tempList = [1,2,3,4]
194 tempList.reverse()
195 for x in tempList:
196     print x,
197
198 #如果不是list, 最通用但是稍慢的解决方案是
199 print '\n-----|'
200 tempTuple = (1,2,3,4)
201 for i in range(len(tempTuple)-1,-1,-1):
202     print tempTuple[i],
203
4 3 2 1
-----
4 3 2 1
[Finished in 0.0s]
```

(二)、如何用Python来进行查询和替换一个文本字符串?

```
205 #Python中的replace()可以用进行字符串替换。
206 tempStr = 'Hello Java,Hello Python,Use JavaScript!'
207 print tempStr.replace('Hello','Bye')
208
209 #Python中的sub() 可以用来查找并替换字符串, sub()是使用正则来匹配。
210 import re
211 rex = r'(Hello|Use)'
212 print re.sub(rex, 'Bye', tempStr)]

Bye Java,Bye Python,Use JavaScript!
Bye Java,Bye Python,Bye JavaScript!
[Finished in 0.0s]
```

赞同 692

30 条评论

分享

收藏

...

所谓单例就是一个类从始至终只能产生一个实例。

方法一：使用__new__方法

在__new__方法中把类实例绑定到类变量_instance上，如果cls._instance为None表示该类还没有实例化过，实例化该类并返回。如果cls._instance不为None表示该类已实例化，直接返回cls._instance

```

215 class Singleton(object):
216
217     def __new__(cls, *args, **kwargs):
218         if not hasattr(cls, '_instance'):
219             cls._instance = object.__new__(cls, *args, **kwargs)
220         return cls._instance
221
222 class TestClass(Singleton):
223     a = 1
224
225 test1 = TestClass()
226 test2 = TestClass()
227 print test1.a, test2.a
228
229 test1.a = 2
230 print test1.a, test2.a
231
232 print id(test1), id(test2)

```

```

1 1
2 2
140407955713424 140407955713424
[Finished in 0.1s]

```

方法二：使用装饰器(decorator)

```

234 def Singleton(cls, *args, **kwargs):
235     instances = {}
236     def _singleton():
237         if cls not in instances:
238             instances[cls] = cls(*args, **kwargs)
239         return instances[cls]
240     return _singleton
241
242 @Singleton
243 class TestClass(object):
244     a = 1
245     test1 = TestClass()
246     test2 = TestClass()
247     print test1.a, test2.a
248
249     test1.a = 2
250     print test1.a, test2.a
251
252     print id(test1), id(test2)

```

```

1 1
2 2
140424195876752 140424195876752
[Finished in 0.1s]

```

方法三：使用__metaclass__ (元类)



赞同 692



分享

方法四：共享属性

所谓单例就是所有的引用（实例，对象）拥有相同的属性和方法，同一个类的实例天生都会有相同的方法，那我们只需要保证同一个类所产生的实例都具有相同的属性。所有实例共享属性最简单直接的方法就是共享 `__dict__` 属性指向。

方法五：Python模板

(四)、重新实现`str.strip()`，注意不能使用`string.*strip()`



赞同 692



分享

(五)、 阅读下面的代码，它的输出结果是什么？（**super**）

想要更佳详细的了解有关于**super**，点击这里：[python super\(\) - 漩涡鸣人 - 博客园](#)

(六)、 Python的函数参数传递

关于以上代码：**Python**中**string**、**tuple**、**number**属于不可更改对象，而**list**和**dict**属于可修改对象。

(七)、 类变量和实例变量

类变量就是供给类使用的变量，

self.name不再引用**Person**的类变量**name**

- 下班部分：**name**是**list**（可更改对象）

▲

赞同 692

▼

分享

（八）、Python在函数式编程方面的支持。

方面一：**lambda**表达式：体现原理就是python允许**lambda**关键字创建匿名函数，所谓匿名就是python不会创建名称空间，**lambda**表达式返回可调用的函数对象，这些都符合函数式编程的思想。

方面二：内建函数：**filter()**、**reduce()**、**map()**

有关于三个内建函数的详细解答，请点击这里：[Python 函数式编程 - BeginMan - 博客园](#)

（九）、以下代码将输出什么？（考察**list**）

可能你会问：why？为什么它没有报错：是的，它不会报错。试图访问一个列表的以超出列表成员数作为开始索引的切片将不会导致**IndexError**，并且仅仅返回一个空**list**

（十）、以下代码将输出什么？（考察）

赞同 692



分享

这个原因是Python的闭包的后期绑定导致的**late binding**。这意味着在闭包中的变量是在内部函数被调用的时候被查找。所以结果是，当任何的**testFun()**返回的函数被调用，在那时，**i**的值是在它被调用时的周围作用域中查找，到那时，无论哪个返回的函数被调用，for循环已经完成，**i**的最后值是3，因此每个返回的函数**testFun**的值都是3。

更详细的解答可以看这里：[Python中后期绑定\(late binding\)是什么意思？](#)

本文参考文献资料：

- 「1」 [七、PYTHON 一些基础面试题总结](#)
- 「2」 [Python单例模式的4种实现方法](#)
- 「3」 [python 面试题 - 知识点整理](#)
- 「4」 [很全的 Python 面试题 - Python - 伯乐在线](#)
- 「5」 [Python 函数式编程 - BeginMan - 博客园](#)

学习编程，欢迎关注专栏：[学习编程 - 知乎专栏](#)

编辑于 2017-05-23

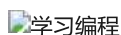
「真诚赞赏，手留余香」

赞赏

3 人已赞赏

Python 编程 程序员

文章被以下专栏收录



学习编程

微信公众号：一个程序员的日常 莫道君行早，更有早行人。全心敲代码，天道...

关注专栏

推荐阅读

董伟明

发布于学习编程

路人甲

发布于学习编程

路人甲

发布于学习编程

赞同 692

分享

30 条评论 切换为时间排序

写下你的评论...

😊

「已注销」

2 年前

第一题非数组用 String 的 index，比如string[::-1] 如何?

👍 6

 路人甲 (作者) 回复 「已注销」

2 年前

那么就会倒序一个string。
tempStr = 'hello'
print tempStr[::-1]
结果就是: olleh

👍 赞

香依 回复 路人甲 (作者)

2 年前

讲道理你只是输出的话，pop要更快吧

👍 赞

展开其他 1 条回复

MeeLu

2 年前

这些题很细了呀，中级题了吧!

👍 1

Jack Zhou

2 年前

不错，正需要Mark, THX

👍 赞

ky二

2 年前

作者辛苦了

👍 赞

Withfilling

2 年前

我要关注你关注到我大学毕业!

👍 2

 路人甲 (作者) 回复 Withfilling

2 年前

你的意思是大学毕业就不关注我咯!

👍 3

邵昊

2 年前

strip的话用 正则可以不

👍 赞

 路人甲 (作者) 回复 邵昊

2 年前

实现即可。

👍 赞

赞同 692

分享

颜真卿

2 年前

这是python2

赞

柳傾

2 年前

写的不错，我建议代码部分不要贴图，直接贴代码，不知道知乎支不支持代码显示。

1

洛卜哒

2 年前

语言相关的基础题

赞

老谭

2 年前

第一题这样可以不？
a = (1,2,3,4)
for e in a[::-1]:
print(e)

1

juggle 回复 老谭

2 年前

感觉这样更好

赞

大狗罗宾逊

2 年前

第一题的第二种答案不太直观，直接迭代 reversed(list(tempTuple)) 更好吧。

1

牛奶咖啡8023

2 年前

mark

赞

Jon Wang

2 年前

反序迭代调用reverse()是O(n)的时间，为什么会是最快的呢？

赞

异次元代码

2 年前

MARK

1

花生

2 年前

第一题的第一种方式，那样直接放那儿将原列表逆序了放那儿了也不管不太好吧。题目是反序迭代，建议`` for x in reserved(tempList):``

赞

花生

2 年前

另，一点小建议。改进下命名方式，符合 PEP8 多好。虽然是演示给新人的代码，但一开始形成好的风格对新人很有益。

赞

王宇

1 年前

有个疑问：既然想要所有实例都共用（单例模式），为什么实际实用中不只生成一个对象实例呢？这样不就解决问题了？为什么要创建一堆实例，又要让它们时刻保持一致？

赞

赞同 692

30 条评论

分享

收藏

...

404 了

 赞

Beamouser

1 年前

第一题如果不考虑内存 序列(字符串 元组 列表)不是都可以使用 seq[::-1]吗 也不会改变原内容啊 这是个副本啊

 赞

1 2 下一页