## Python答辩题V9.8

1. Python你都会哪些？

我会（方法，程序控制【if…else ,for while】,面向对象，继承，封装，系统模块的使用[random,os ,smtplib,ddt,unittest……]，文件读取，数据库增删改查，excel表的读取，单元测试框架的封装）……

1. 解释面向对象是什么意思？

就是将所有过程进行大致罗列，进行编程。

面向对象编程，是一种封装代码的方法。

面向对象编程，也是一种封装的思想，它可以更好地模拟真实世界里的事物（将其视为对象），并把描述特征的数据和代码块（函数）封装到一起。

1. Python的优点和特点？

优点：

1、Python 易于学习;

2、用少量的代码构建出很多功能;（高效的高级数据结构）

3、Python 拥有最成熟的程序包资源库之一;

4、Python完全支持面向对象;

5、Python 是跨平台且开源的。

Python 是一种面向对象、解释型的脚本语言，同时也是一种功能强大而完善的通用型语言。相比其他编程语言（比如 Java），Python 代码非常简单，上手非常容易。 Python优点： （1）简单易学 （2）开源 （3）高级语言 （4）解释性语言

1. 基础,python的数据类型？

（7种，不可变数据类型。）

1）、数字 ---> int类

2）、布尔值 --->bool类

3）、字符串 --->str类

4）、列表 --->list类

5）、元组 --->tuple类

6）、字典 --->dict类

7）、集合 -->set类

1. 列表[ ]，元组()，字典{}，集合sets()
2. 能不能将列表list = [1,3,4,5,2,6,1]快速去重？

set (list)

ids = [1,4,3,3,4,2,3,4,5,6,1]

ids = list(set(ids))

1. (1)是不是元组？如果不是怎么才能变成元组呢？

不是， (1,)

1. 列表中的深拷贝和浅拷贝的区别？

import copy  
li = [1,2,3,[5,6,7]]  
li1 = copy.deepcopy(li)*# 浅拷贝 [1,2,3,[4,6,7]]*li1[3][0] = 4  
  
print(li[3][0])

浅拷贝：无法拷贝列表中的子列表

深拷贝：完全拷贝

1. 列表与元组的区别？

列表是动态数组，它们不可变且可以重设长度（改变其内部元素的个数）。

元组是静态数组，它们不可变，且其内部数据一旦创建便无法改变。

1. 能不能将元组快速转换成列表？

A = (1,4,5)

List(A)

使用 list 函数 可以把 元组 转换成 列表

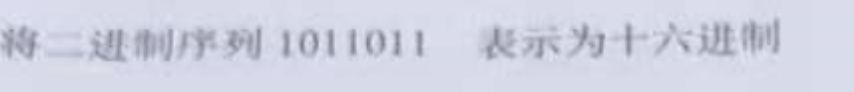
list(元组)

使用 tuple 函数 可以把 列表 转换成 元组

tuple(列表)

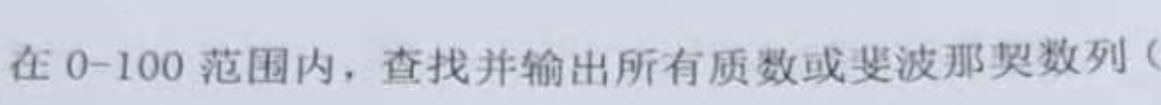
1. 写个算法呗？冒泡排序，选择排序，列表翻转，字符串字符的统计，每隔60秒读取日志文件数据。

arr = [2,3,4,1,5,6,9,8,7]  
  
n = len(arr)  
for i in range(n):  
 for j in range(0,n-i-1):  
 if arr[j]>arr[j+1]:  
 arr[j],arr[j+1] = arr[j+1],arr[j]  
  
print(arr)

1. 

bin() oct() int() hex(0b1011011)

>>> hex(int('101010',2))  
'0x2a'

1. 

for i in range(2,100):  
 for j in range(2,i):  
 if i%j==0:  
 break  
 else:  
 print(i)

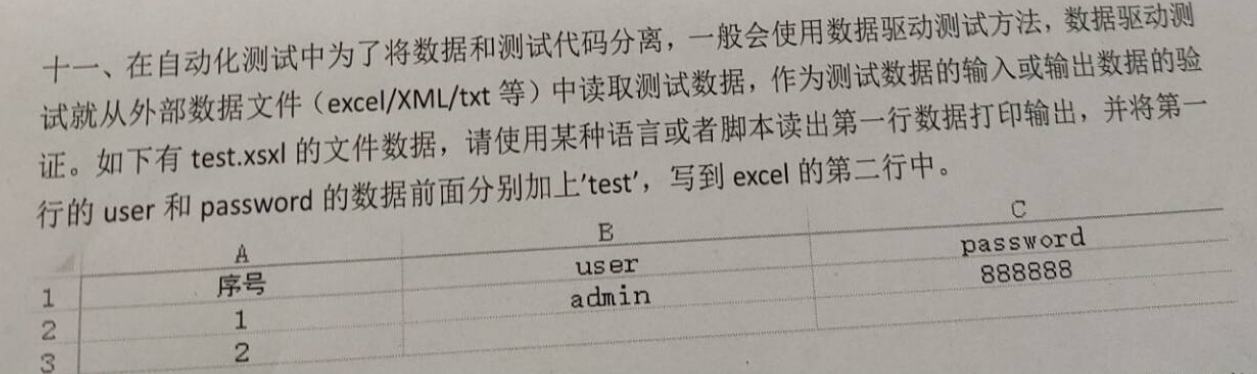
1. 如何用面向对象描述一个水杯？

class Cup: 属性 行为

height = “”

wide = “”

Color = “”

1. 

import xlrd

Import xlwt

------读取数据

# 获取工作簿

wb = xlrd.open\_workbook(filename="test.xlsx",encoding\_override=True)

# 通过wb获取选项卡

sheet = wb.sheet\_by\_name("test")

# 获取行列数据

rows = sheet.nrows #多少行

cols = sheet.ncols #多少列

# 打印

for i in range(rows):

a= sheet.row\_values(i)

if i==1:

print(a)

-----插入数据

# 创建工作表对象

workbook = xlwt.Workbook(encoding="utf-8")

# 设置表名

sh = workbook.add\_sheet("test")

# 插入数据

for i in range(rows):

for j in range(cols):

onedata = sheet.cell\_value(i, j)

sh.write(i,j,onedata)

sh.write(2,1,"testadmin")

sh.write(2,2,"test888888")

# 设置文件名

workbook.save("test.xlsx")

1. Python的lambda表达式？lambda表达式写一个遍历列表？

lambda 表达式，又称匿名函数，常用来表示内部仅包含 1 行表达式的函数。

--------------------------------------------------

lambda 表达式的语法格式如下：

name = lambda [list] : 表达式

函数名 = lambda 参数 ：返回值

add = lambda x,y:x+y

print(add(3,4))

--------------------------------------------------

li = [1,2,3,4,5,6]  
A =[x for x in li]  
print(A)

1. 什么是继承？继承有几种方式？继承的好处和坏处

继承是类与类之间的关系（子类继承父类的特点）

两种（普通继承和super继承)

好处：提高了代码的利用率，方便了代码的书写

坏处：内置类型的方法不会调用子类覆盖的方法

1. Python中如何实现多线程？多线程中的start和run方法的区别？

使用import threading threading.Thread

1. 通过threading.Thread进行创建多线程
2. 通过继承threading.Thread定义子类创建多线程

start() 方法是启动一个子线程

run() 方法并不启动一个新线程，就是在主线程中调用了一个普通函数而已。

1. Python是如何管理内存？

Python的内存管理主要有三种机制：引用计数机制，垃圾回收机制和内存池机制。

python内部使用引用计数，来保持追踪内存中的对象，Python内部记录了对象有多少个引用，即引用计数，当对象被创建时就创建了一个引用计数，当对象不再需要时，这个对象的引用计数为0时，它被垃圾回收。Python提供了对内存的垃圾收集机制，但是它将不用的内存放到内存池而不是返回给操作系统。

1. 方法中\*args与\*\*kwargs的区别？

\*args 表示任何多个无名参数，它是一个tuple；\*\*kwargs 表示关键字参数，它是一个dict。

1. 什么是python的字典？字典的键能用列表做键吗?

字典也是Python语言中经常使用的一种数据类型

Python不允许用可变数据类型做键。

1. 如何删除字符串两边的空格？ “ hello world ” ,能讲讲str里的常用的api方法么？如何将字符串”3”转换成数字3?

strip()

" xyz ".strip() # returns "xyz"

" xyz ".lstrip() # returns "xyz "

" xyz ".rstrip() # returns " xyz"

" x y z ".replace(' ', '') # returns "xyz"

获取字符串长度len()、

 字符串分割组合join()、strip()

字符串替换replace()、

字符串查找find()、index()等

“3” 🡪3 int()

3 🡪”3” str()

1. pass关键字的作用？如何申明一个变量为全局变量？

表示一个占位符，能够保证程序的代码结构正确！

global

1. 有几种进制？

|  |  |  |  |
| --- | --- | --- | --- |
| 二进制 | 八进制 | 十进制 | 十六进制 |
| 0b | 0o |  | 0x |
| 0b01 | 0o76420 | 672864892789 | 0xf4e3 |

1. 什么是迭代器，和生成器？

迭代器：含有\_\_iter\_\_和\_\_next\_\_方法 (包含\_\_next\_\_方法的可迭代对象就是迭代器)

生成器：包括含有yield这个关键字，生成器也是迭代器，调动next把函数变成迭代器。

1. 文件读写？（步骤）有几种操作模式？（r,w,+,a,b）。用过哪些API？你用在测试的什么地方？

Python文件的打开或创建可以使用函数open()。

open(file, mode='r', buffering=-1, encoding=None,errors=None, newline=None, closefd=True, opener=None)

1、打开文件

模式：

r、w、b(图片、MP3\MP4),a(附加),+可读可写

2、写入数据

3、关闭资源

用在生成测试报告阶段

1. 所有类的父类？

**object**

1. Python2.x 与python3.x区别？

1.默认编码不一样

　　python2中默认编码是ASCII码

　　python3中默认编码是unicode码

2.python2中字符串和python3中的字符串

　　python2中字符串有两种类型,分别为unicode和str前者为文本字符串,后者是字节序列,由于没有明显的划分,比较混乱,在pyton3中严格区分,分别用str表示字符串,byte表示字节序列,任何网络的传输只接受字节序列

3.True和False

　　python2中True和False是全局变量,是能够修改的,而在python3中True和False是关键字

4.经典类和新式类

　　python2中默认是 经典类同时也存在新式类,经典类多继承搜索顺序是深度优先:

　　　　深度优先是先深刻继承树左侧查找,而后在返回开始查找右侧

　　python3中只有新式类,新式类多继承的查找顺序是广度优先:

　　　　广度优先是水平查找,而后在 向上查找

5.keys和values

　　python2中keys方法直接返回的是一个列表

　　python3中返回的是一个dict\_keys([])对象,这个对象更像是set集合,而不是list

　　同理,values在python2中返回的是一个列表

　　python3中返回的是一个dict\_values([])对象

6.range和xrange

　　python2中range是返回一个列表值,而xrange返回的是一个可迭代对象

　　python3中不存在xrange,只有range,可是python3中的range与python2中的xrange是返回一样的结果

1. Python如何处理异常？

Try..except.finally

1. PDBC技术的操作步骤？获取所有数据?

连接mysql：

import pymysql

1、获取连接

2、通过连接创建控制台

3、控制台执行sql语句

4、关闭资源

#获取数据库连接

con = pymysql.connect(host='localhost',user='root',password='root',database='bank')

# 创建控制台

cursor = con.cursor()

# 查询表中数据

sql = "select \* from table"

# 执行sql

cursor.execute(sql)

# 提交

con.commit()

# 关闭

cursor.close()

con.close()

1. 参数化怎么做(步骤)？用过哪些注解？@ddt @data @unpack

from ddt import ddt

from ddt import data

from ddt import unpack

@ddt # 将测试类用@ddt修饰

@data(\*da) # 引入数据源

@unpack # 将数据源的数据进行解包，不然的话会当成一个整数传进来

1. 如何使用测试集来测试？

import unittest  
from HTMLTestRunner import HTMLTestRunner  
  
# 1.创建测试集  
suite = unittest.TestSuite()  
# 2.让测试加载器自己加载所用用例  
tests = unittest.defaultTestLoader.discover(r"E:\ytk\develop\works\python\class\day16\class",pattern="Test\*.py")  
# 3.将所用例 放入测试集  
suite.addTests(tests)  
# 4.创建测试运行器  
f = open(file="银行系统测试报告.html",mode="w+",encoding="utf-8")  
runner = HTMLTestRunner.HTMLTestRunner(  
 stream=f,  
 title = "这是一个银行系统测试报告！",  
 verbosity=3,  
 description="执行了开户、存钱、取钱、转账、查询用例"  
)  
# 5.让运行器生成测试报告  
runner.run(suite)

1. 测试报告？

import unittest  
from HTMLTestRunner import HTMLTestRunner  
  
# 1.创建测试集  
suite = unittest.TestSuite()  
# 2.让测试加载器自己加载所用用例  
tests = unittest.defaultTestLoader.discover(r"E:\ytk\develop\works\python\class\day16\class",pattern="Test\*.py")  
# 3.将所用例 放入测试集  
suite.addTests(tests)  
# 4.创建测试运行器  
f = open(file="银行系统测试报告.html",mode="w+",encoding="utf-8")  
runner = HTMLTestRunner.HTMLTestRunner(  
 stream=f,  
 title = "这是一个银行系统测试报告！",  
 verbosity=3,  
 description="执行了开户、存钱、取钱、转账、查询用例"  
)  
# 5.让运行器生成测试报告  
runner.run(suite)

1. 断言？多列举几个

assertEquals() assertNone()

1. 你的单元测试框架分为哪些模块？写个大致代码？数据读取工具写？邮件发送模块？