**1、python中is和==的区别**

Python中对象包含的三个基本要素，分别是：id(身份标识) 、type(数据类型)和value(值)。

‘==’比较的是value值 值是否相等

‘is’比较的是id 是否为同一个对象

**2、简述read、readline、readlines的区别**

read([size])方法从文件当前位置起读取size个字节，若无参数size，则表示读 取至文件结束为止，它范围为字符串对象

Readline（）**从字面意思可以看出，该方法每次读出一行内容，所以，读取时占 用内存小，比较适合大文件，该方法返回一个字符串对象。**

**readlines()方法读取整个文件所有行，保存在一个列表(list)变量中，每行作为一个元素，但读取大文件会比较占内存**

1. **\*args,\*\*kwargs的作用是分别什么？**

\*args和\*\*kwargs通常使用在函数定义里，\*args允许函数传入不定量个数的非关键字参数，\*\*kwargs允许函数传入不定量个数的关键字参数

**4、描述yield作用。**

保存当前运行状态（断点），然后暂停执行，即将函数挂起

将yeild关键字后面表达式的值作为返回值返回，此时可以理解为起到了return的作用，当使用next()、inte()函数让函数从断点处继续执行，即唤醒函数

**5、什么是装饰器。**

装饰器本质上是一个Python函数，它可以让其他函数在不需要做任何代码变动的前提下增加额外功能，装饰器的返回值也是一个函数对象。

**6、python是怎么进行内存管理的？**

引用计数：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。

**7、python中如何拷贝一个对象？（赋值，浅拷贝，深拷贝的区别）**

赋值（=），就是创建了对象的一个新的引用，修改其中任意一个变量都会影响到另一个。

浅拷贝：创建一个新的对象，但它包含的是对原始对象中包含项的引用（如果用引用的方式修改其中一个对象，另外一个也会修改改变）{1,完全切片方法；2，工厂函数，如list()；3，copy模块的copy()函数}

深拷贝：创建一个新的对象，并且递归的复制它所包含的对象（修改其中一个，另外一个不会改变）{copy模块的deep.deepcopy()函数}

1. **索引和切片**

索引：取出数组s中第3个元素：x=s[2]

切片：s[参数1:参数2:参数3] 参数1 起始位置 参数2 终止位置 参数3 步长

1. **pyhon 常用内建函数**
2. abs() 函数返回数字的绝对值。
3. all() 函数用于判断给定的可迭代参数 iterable 中的所有元素是否都为 TRUE，如果是返回 True，否则返回 False。
4. Ord(“a”) 返回字符串对应的十进制整数。
5. chr(i) 返回十进制数对应的字符（0~255）
6. bin(x）将整数转换为二进制字符串
7. Max(1,2,3) 返回最大值
8. Min(1,2,3) 返回最小值

**10、**python内建数据类型有哪些****

整型--int 布尔型--bool 字符串--str 列表--list 元组--tuple 字典--dict

****11、5个Linux常用命令****

**ls  pwd  cd  touch  rm  mkdir  tree  cp  mv  cat   **grep****

****12、python2和python3区别？列举5个****

**1、Python3 使用 print 必须要以小括号包裹打印内容，比如 print('hi')**

**Python2 既可以使用带小括号的方式，也可以使用一个空格来分隔打印内 容，比如 print 'hi'**

**2、python2 range(1,10)返回列表，python3** range() 函数返回的是一个可迭代对象（类型是对象），而不是列表类型， 所以打印的时候不会打印列表。

**3、python2中使用ascii编码，python中使用utf-8编码**

**4、python2中unicode表示字符串序列，str表示字节序列**

**python3中str表示字符串序列，byte表示字节序列**

**5、python2中为正常显示中文，引入coding声明，python3中不需要**

**6、python2中是raw\_input()函数，python3中是input()函数**

****13、列出5个python标准库****

os：提供了不少与操作系统相关联的函数

sys:   通常用于命令行参数

re:   正则匹配

math: 数学运算

datetime:处理日期时间

**14、什么是lambda函数?它有什么好处?**

lambda表达式，通常是在需要一个函数，但是又不想费神去命名一个函数 的场合下使用，也就是指匿名函数。

Python允许你定义一种单行的小函数。定义lambda函数的形式如下(lambda参数：表达式)lambda函数默认返回表达式的值。你也可以将其赋值 给一个变量。lambda函数可以接受任意个参数，包括可选参数，但是表达 式只有一个。

1. **运算符优先级**

|  |  |
| --- | --- |
| 运算符 | 描述 |
| \*\* | 指数 (最高优先级) |
| ~ + - | 按位翻转, 一元加号和减号 (最后两个的方法名为 +@ 和 -@) |
| \* / % // | 乘，除，取模和取整除 |
| + - | 加法减法 |
| >> << | 右移，左移运算符 |
| & | 位 'AND' |
| ^ | | 位运算符 |
| <= < > >= | 比较运算符 |
| <> == != | 等于运算符 |
| = %= /= //= -= += \*= \*\*= | 赋值运算符 |
| is is not | 身份运算符 |
| in not in | 成员运算符 |
| not and or | 逻辑运算符 |

**\*\*\*\*记不住用括号,括号优先级最高**

**16、什么是PEP8？**

PEP8是一个编程规范，内容是一些关于如何让你的程序更具可读性的建议。

**17、.Python中的pass是什么？**

Pass是一个在Python中不会被执行的语句。在复杂语句中，如果一个地方需要暂时被留白，它常常被用于占位符。

**18、迭代器和生成器的区别**

迭代器是一个更加抽象的概念，任何对象，如果它的类有next方法和iter 方法返回自身。对于string、list、dict、tuple等这类容器对象，使用for循环遍历是很方便的。在后台for语句对容器对象调用iter()函数，iter()是Python的内置函数。iter()会返回一个定义了next()方法的迭代器对象，它在 容器中逐个访问 容器内元素，next()也是python的内置函数。在没有后续元素时，next()会抛出 一个StopIterration的异常。

生成器（Generator）是创建迭代器的简单而强大的工具。它们写起来就 像是正规的函数，只是在返回数据的时候需要使用yield语句。每次next() 被调用时，生成器会返回它脱离的位置（它记忆语句最后一次执行的位置和 所有的数据值）

**19、Python里面如何生成随机数？**

random模块

随机整数：random.randint(a,b)：返回随机整数x,a<=x<=b

random.randrange(start,stop,[,step])：返回一个范围在(start,stop,step) 之间的随机整数，不包括结束值。

随机实数：random.random( ):返回0到1之间的浮点数

random.uniform(a,b):返回指定范围内的浮点数。

# **20、单引号，双引号，三引号的区别**

单引号和双引号是等效的，如果要换行，需要符号(\),三引号则可以直接换行，并且可以包含注释

1. **面向对象中super的作用？**

在子类派生出新的方法中重用父类的功能

1. **列举面向对象中特殊成员(带双下划线的特殊方法，如：\_\_init\_\_、等)**

# \_\_init\_\_ 负责将类的实例化

# \_\_str\_\_ print打印一个对象时触发

# \_\_repr\_\_ print打印一个对象时而且该实例没有实现\_\_str\_\_触发

# \_\_setattr\_\_ 添加/修改属性会触发它的执行

#\_\_delattr\_\_ 删除属性的时候会触发

# \_\_delete\_\_ 采用del删除属性时，触发

1. **静态方法和类方法区别？**
2. 静态方法： 相当于普通函数，通过类直接调用，不需要创建对象，不会隐式传递 staticmethod

使用场景

不需要访问实例属性，比如我想定义一个关于时间操作的类，其中有一个获取当前 时间的函数。

2. 类方法： 通过类调用，第一个参数默认是类本身 classmethod

使用场景

**一般是需要去访问写死的变量，才会用到类方法装饰器**

**24、通过字符串来操作类与对象的属性，这种操作称为反射**

hasattr（obj,某种属性）

测对象自己中是否有指定的函数属性与数据属性,有返回True

setattr(obj,某种属性，属性对应的值)

在对象自己中新增或修改对象中的函数与数据属性

delattr(obj，某种属性，属性对应的值）

#删除对象自己中的属性

getattr(object,某种属性（字符串）,[default])

返回对象自己中数据属性的值,与函数属性的内存地址,没有则报错。如果写了默认值没有时则返回默认值