

## [学习交流] python 面试常见的 25 个问题

### 1. 到底什么是 Python？你可以在回答中与其他技术进行对比

下面是一些关键点：

- Python 是一种解释型语言。这就是说，与 C 语言和 C 的衍生语言不同，Python 代码在运行之前不需要编译。其他解释型语言还包括 PHP 和 Ruby。
- Python 是动态类型语言，指的是你在声明变量时，不需要说明变量的类型。你可以直接编写类似 `x=111` 和 `x="I'm a string"` 这样的代码，程序不会报错。
- Python 非常适合面向对象的编程（OOP），因为它支持通过组合（composition）与继承（inheritance）的方式定义类（class）。

Python 中没有访问说明符（access specifier，类似 C++ 中的 `public` 和 `private`），这么设计的依据是“大家都是成年人了”。

- 在 Python 语言中，函数是第一类对象（first-class objects）。这指的是它们可以被指定给变量，函数既能返回函数类型，也可以接受函数作为输入。类（class）也是第一类对象。
- Python 代码编写快，但是运行速度比编译语言通常要慢。好在 Python 允许加入基于 C 语言编写的扩展，因此我们能够优化代码，消除瓶颈，这点通常是可以实现的。

`numpy` 就是一个很好地例子，它的运行速度真的非常快，因为很多算术运算其实并不是通过 Python 实现的。

- Python 用途非常广泛——网络应用，自动化，科学建模，大数据应用，等等。它也常被用作“胶水语言”，帮助其他语言和组件改善运行状况。
- Python 让困难的事情变得容易，因此程序员可以专注于算法和数据结构的设计，而不用处理底层的细节。

为什么提这个问题：如果你应聘的是一个 *Python* 开发岗位，你就应该知道这是门什么样的语言，以及它为什么这么酷。以及它哪里不好。

### 2. 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。

### 3. 什么是 *pickling* 和 *unpickling*?

*Pickle* 模块读入任何 *Python* 对象，将它们转换成字符串，然后使用 *dump* 函数将其转储到一个文件中——这个过程叫做 *pickling*。

反之从存储的字符串文件中提取原始 *Python* 对象的过程，叫做 *unpickling*。

### 4. *Python* 是如何被解释的?

*Python* 是一种解释性语言，它的源代码可以直接运行。*Python* 解释器会将源代码转换成中间语言，之后再翻译成机器码再执行。

### 5. *Python* 是怎样管理内存的?

*Python* 的内存管理是由私有 *heap* 空间管理的。所有的 *Python* 对象和数据结构都在一个私有 *heap* 中。程序员没有访问该 *heap* 的权限，只有解释器才能对它进行操作。

为 *Python* 的 *heap* 空间分配内存是由 *Python* 的内存管理模块进行的，其核心 *API* 会提供一些访问该模块的方法供程序员使用。

*Python* 有自带的垃圾回收系统，它回收并释放没有被使用的内存，让它们能够被其他程序使用。

### 6. 有哪些工具可以帮助 *debug* 或做静态分析?

*PyChecker* 是一个静态分析工具，它不仅能报告源代码中的错误，并且会报告错误类型和复杂度。*Pylint* 是检验模块是否达到代码标准的另一个工具。

### 7. 什么是 *Python* 装饰器?

Python 装饰器是 Python 中的特有变动，可以使修改函数变得更容易。

## 8. 数组和元组之间的区别是什么？

数组和元组之间的区别：数组内容是可以被修改的，而元组内容是只读的。另外，元组可以被哈希，比如作为字典的关键字。

## 9. 参数按值传递和引用传递是怎样实现的？

Python 中的一切都是类，所有的变量都是一个对象的引用。引用的值是由函数确定的，因此无法被改变。但是如果一个对象是可以被修改的，你可以改动对象。

## 10. 字典推导式和列表推导式是什么？

它们是可以轻松创建字典和列表的语法结构。

## 11. Python 都有哪些自带的数据结构？

Python 自带的数据结构分为可变的和不可变的。可变的有：数组、集合、字典；不可变的有：字符串、元组、数。

## 12. 什么是 Python 的命名空间？

在 Python 中，所有的名字都存在于一个空间中，它们在该空间中存在和被操作 \_\_ 这就是命名空间。

它就好像一个盒子，每一个变量名字都对应装着一个对象。当查询变量的时候，会从该盒子里面寻找相应的对象。

## 13. Python 中的 lambda 是什么？

这是一个常被用于代码中的单个表达式的匿名函数。

#### 14.为什么 `lambda` 没有语句?

匿名函数 `lambda` 没有语句的原因，是它被用于在代码被执行的时候构建新的函数对象并且返回。

#### 15.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。

#### 16.Python 中什么是遍历器?

遍历器用于遍历一组元素，比如列表这样的容器。

#### 17.Python 中的 `unittest` 是什么?

在 `Python` 中，`unittest` 是 `Python` 中的单元测试框架。它拥有支持共享搭建、自动测试、在测试中暂停代码、将不同测试迭代成一组，等等的功能。

#### 18.在 `Python` 中什么是 `slicing`?

`Slicing` 是一种在有序的对象类型中（数组，元组，字符串）节选某一段的语法。

#### 19.在 `Python` 中什么是构造器?

生成器是实现迭代器的一种机制。它功能的实现依赖于 `yield` 表达式，除此之外它跟普通的函数没有两样。

#### 20.Python 中的 `docstring` 是什么?

`Python` 中文档字符串被称为 `docstring`，它在 `Python` 中的作用是为函数、模块和类注释生成文档。

## 21. 如何在 Python 中拷贝一个对象？

如果要在 Python 中拷贝一个对象，大多数时候你可以用 `copy.copy()` 或者 `copy.deepcopy()`。但并不是所有的对象都可以被拷贝。

## 22. Python 中的负索引是什么？

Python 中的序列索引可以是正也可以是负。如果是正索引，0 是序列中的第一个索引，1 是第二个索引。如果是负索引，(-1) 是最后一个索引而 (-2) 是倒数第二个索引。

## 23. 如何将一个数字转换成一个字符串？

你可以使用自带函数 `str()` 将一个数字转换为字符串。如果你想要八进制或者十六进制数，可以用 `oct()` 或 `hex()`。

## 24. xrange 和 range 的区别是什么？

Xrange 用于返回一个 xrange 对象，而 range 用于返回一个数组。不管那个范围多大，Xrange 都使用同样的内存。

## 25. Python 中的模块和包是什么？

在 Python 中，模块是搭建程序的一种方式。每一个 Python 代码文件都是一个模块，并可以引用其他的模块，比如对象和属性。

一个包含许多 Python 代码的文件夹是一个包。一个包可以包含模块和子文件夹。