# Python学习笔记-简介

Python是一种高层次，解释，互动性和面向对象的脚本语言。Python被设计成具有很强的可读性语言。它采用英语关键字，而其他语言一般使用标点符号，并且具有比其他语言有较少的句法结构。

* Python是解释型的语言：Python 是在运行时被解释处理。你不需要在执行前编译程序。这类似于Perl和PHP。
* Python是交互式的： 实际上，可以在 Python 提示和解释直接交互来编写程序。
* Python是面向对象：Python支持面向对象式或编程的对象中封装代码的技术。
* Python是一种初学者的语言：Python是初学者级别程序员伟大的语言，并支持广泛从简单的文字处理，网页浏览器应用程序到游戏的开发。

## Python解释器

当你编写了一段Python程序，Python解释器将读取程序，并按照其中的命令执行，得出结果。解释器是代码与机器的计算机硬件之间的软件逻辑层。当Python包安装在机器上后，它包含了一些最小化的组件：一个解释器和支持的库。根据使用情况的不同，Python解释器可能采取可执行程序的形式，或是作为链接到另一个程序的一系列库。

## 字节码编译

当程序执行时，Python内部会先将源代码编译成字节码的形式。编译是一个简单的翻译步骤，而且字节码是源代码底层的、与平台无关的表现形式。概括的说，Python通过把每一条源语句分解为单一步骤来讲这些源语句翻译成一组字节码指令。这些字节码可以提高执行速度。Python这样保存字节码是作为一种启动速度的优化。下一次运行程序时，如果你在上次保存字节码之后没有修改过源代码的话，Python将会加载.pyc文件并跳过编译这个步骤。当Python必须重编译时，它会自动检查源文件和字节码文件的时间戳。如果Python在机器上拥有写入权限，Python会将源代码文件编译为字节码文件，与源文件同一目录，如果无写入权限，字节码将会在内存中生成并在程序结束时丢弃。

## Python虚拟机

一旦程序编译成字节码，之后的字节码被发送到Python虚拟机上执行。PVM就是迭代运行字节码指令的一个大循环。PVM是Python的运行引擎，它时常表现为Python系统的一部分，并且它是实际运行脚本的组件。

由于Python在程序开始执行之前不需要预编译和连接，使得Python具有更多的动态语言特性：在运行时，Python程序构建并执行另一个Python程序是可能的。例如，eval和exec内置模块，能够接受并运行包含Python程序代码的字符串。