所有语言都属于编程语言，这个还好理解，但脚本语言和解释语言的区别。。。。我也是懵逼的，说一下解释型和编译型的区别吧。  
  
编译型语言和解释型语言的界限实际上已经比较模糊了，想弄清楚它们的区别，我觉得还挺麻烦的。  
  
简单地说，编译型语言的特点是：**代码必须作为整个工程来编译，**优点是便于类型检查，降低运行时错误率，编译之后可以在相应的平台上独立运行，不再依赖编译器，**一次执行，多次运行（特点1）**，比如C经过gcc编译产生obj文件，只要源码不变，编译一次，每次执行obj文件即可，少了重复编译，也就**运行效率高（特点2），**缺点是**不能跨平台（特点3），**因为编译是针对操作系统进行进行的。  
解释型语言的特点是：代码的每一行可独立解释执行（代码块除外），所以**灵活（特点1）**，比如你可以在命令行状态下一行一行敲python，敲一行执行一下，**跨平台（特点2）**，因为解释型语言是由解释器执行的，只要有解释器（有时候也叫虚拟机）就能执行，缺点是**执行速度慢、效率低（特点3）**，因为每次执行都要解释一次。  
  
综上，编译器把源码编译成可执行文件就不管了，执行是交给操作系统做（所以无法跨平台），但是注意，我们说它运行效率高，但整体编译的过程效率不一定高，编译器把源码到运行的过程分离成了编译+运行，所以运行效率确实高，但解释器是一个整体，就是解释运行，一行一行的解释运行，所以效率低。  
  
对于编译型语言，如果修改一点点代码就要重新编译，是不是效率很低，请看JAVA。  
为什么说JAVA是解释 + 编译的混合型语言，早期的Java是先通过 Javac 将源码翻译成.class字节码，然后JVM 对字节码逐行解释执行，标准的解释型语言。但是当某个方法或代码块运行很频繁时，重复的解释执行效率就很低，为了提高执行速度，引入了即时编译，JVM中的JIT编译器会把热点代码（经常执行的代码）编译成与本地平台相关的机器码，以后执行到这部分代码，操作系统直接执行编译好的机器码就行了，不必重复解释执行，其它代码由JVM中的解释器逐行解释执行。**简单说，经常变动的代码解释执行，拥有了解释型语言灵活的特点，不变的代码（比如引入的库文件）直接编译成机器码，拥有了编译型语言的高效**。  
  
为什么说编译型语言和解释型语言的界限越来越模糊呢？  
你说编译型语言是先**编译**出一个可执行文件，再执行，解释型语言是直接逐行解释执行，但很多解释型语言现在也是先**编译**出一个字节码文件，再执行啊。这里的编译到底该怎么理解？什么是编译？比如python会把重复使用的模块编译到pyc文件中，下一次运行不再重复编译，那编译是指语法分析语义分析吗？但任何一个高级语言都避不开这两步啊，编译是指产生了中间文件吗？那解释型/编译型语言也都产生了中间文件啊，我以为我懂了，但是我又懵了，到最后我觉得，编译型语言和解释型语言都是分为编译和执行两步：  
1. 编译这一步，编译型语言就是**把源码编译成了做系统可执行的文件**，解释型语言就是**把源码编译成了解释器可执行的文件**，  
2. 执行这一步，**编译型语言是由操作系统执行**，解释**型语言是由解释器执行**，就理解为在操作系统上面又封装了一层吧，有了这个解释器，就可以跨平台了。  
  
它们的区别：  
1. 执行流程上不同。编译型语言有一个单独的编译过程，把程序的执行分离成编译和执行两步。而解释型语言只有一步，解释器对源码解释并执行（虽然这个解释的过程也很复杂，但解释和执行是一个整体，由解释器完成），  
2. 产生的中间码不同。编译型语言会产生一个本地平台能运行的可执行文件，然后由当前操作系统运行。解释型语言为了提高效率一般也会产生一个中间码，叫做字节码，然后由解释器执行字节码。  
3. 跨平台。编译型语言针对不同型号的CPU和操作系统，需要编译成不同的可执行文件。  
最后回到你的问题“Python究竟少了哪一部分”。  
其实不仅python，任何一个高级语言，从源码到执行，该走的比如词法分析，语法分析等一步都少不了，最后还不是要变成01去执行？所以不如说是隐藏了什么，对于编译型语言，我们能看到它经过编译器产生了可执行文件，可执行文件其实就是汇编码了，汇编和机器码是一一对应的关系，所以操作系统基本上就可以直接执行了，对于Python呢，你就当在操作系统上又封装了一个python解释器，python经过编译产生的字节码，还没到汇编码那一步，也就是说操作系统没法直接执行pyc文件，但python解释器可以执行啊，从字节码之下，全部隐藏了。  
  
再打个不太恰当的例子，编译型语言就是经过编译，产生了qq.apk 或者 qq.ipa，或者qq.exe，对应小米手机，苹果手机，win电脑这些不同的设备来安装运行，而解释型语言就是，我弄个网页版qq，管你什么设备，有浏览器就能用，我浏览器负责执行，你Android，IOS，Windows瞧好喽。现在虚拟机和解释器的概念也经常混着用，其实也是，解释器不就是虚拟机吗，我管你什么操作系统，装了我虚拟机，你就随便玩，当然，虚拟机和解释器的区别，我是看了很久也没名吧，我就当他们，没区别吧