1. Pyvisa库现在已经不支持vpp43了。原来的位置在pyvisa.legacy
2. Vpp43在1.6之后的版本中无需再导入，也无法导入了。如果需要用到它，可以换成如下方式：

Import visa

Rm=visa.ResourceManager

Lib=rm.visalib

Python模块：

1. 任何python程序均可以作为模块。
2. 模块主要用于定义，包括变量，函数和类等。因此多次导入并不会导致多次执行。
3. 有时候为了检查模块是否正常工作，会加入一些测试代码。为了区别作为程序运行和作为模块导入，一般的做法是将测试语句块放入一个if语句中：

>> if \_\_name\_\_ == ‘\_\_main\_\_’:

(测试语句块)

作为模块导入时，\_\_name\_\_变量为模块名，而作为主程序运行是则是\_\_main\_\_。

1. 为了更好的使用模块，还可以将它们分组成包（package），实际上是包含多个模块的目录。
2. 一个包必须包含一个\_\_init\_\_.py的模块。当包被导入是它将被执行。
3. 探索一个模块最便捷的方法可以导入之后用dir(module\_name)来查看其中的变量和函数。另外也可以直接查看模块的\_\_all\_\_变量。\_\_all\_\_是一个列表，它不一定包含模块的所有变量和函数名，往往包括了希望能为外部程序使用的那么函数和变量。采用from module\_name import \*语句时，导入的就是\_\_all\_\_变量中出现的那么函数和变量。对于每个函数，可以用help()了解其用法和意义。