 本集要点

认识Python的运行环境（为什么选择VN Studio？）

动手安装运行环境及两个重要安装建议

正式学习前，四个你需要提前理解的名词解释

 交流方式

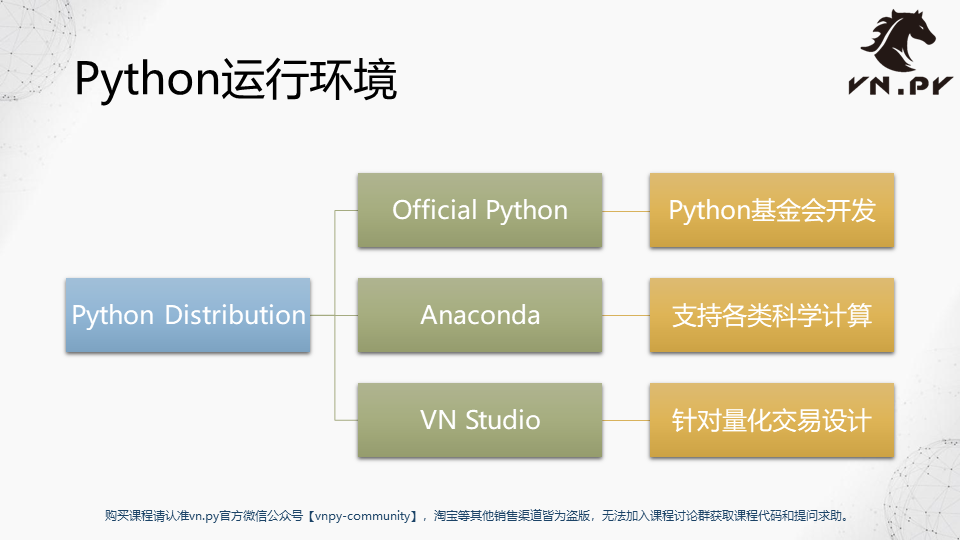
本课程提问可至vn.py论坛，[提问](https://www.vnpy.com/forum/forum/17-liang-hua-jiao-yi-ling-ji-chu-ru-men)！‹‹点击

 完整文字版

欢迎来到全新的【量化交易零基础入门系列】——《30天解锁Python量化开发》课程。今天是第二节课，在上一节课里面已经初步认识了Python这么一门编程语言，在这节课里面就先把运行Python所需要的运行环境VN Studio给安装上。

### **为什么选择VN Studio?**

首先要来认识一下什么叫做Python运行环境呢？



它的英文叫做Python Distribution，直译过来可以叫做Python发行版。我们这边举了三个例子，也是三个不同领域比较经典的例子：

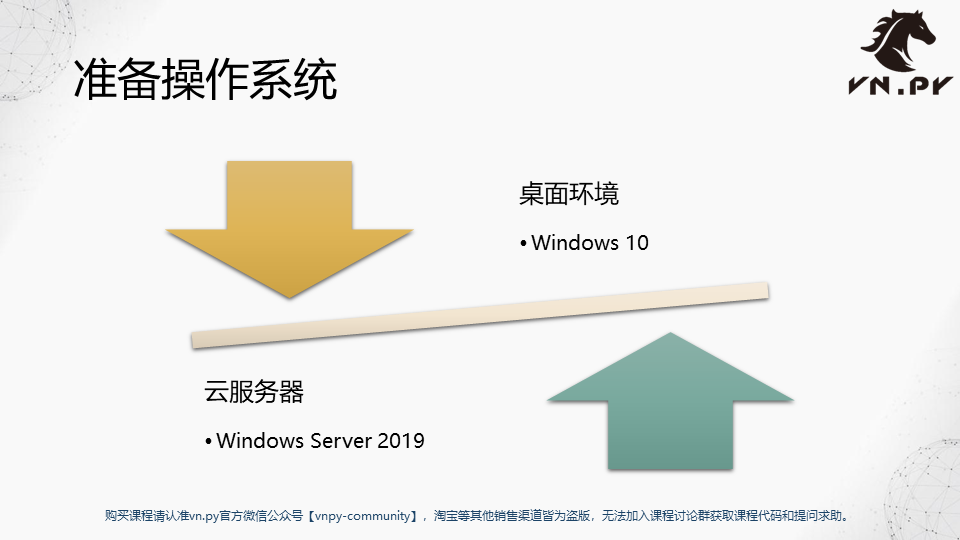
第一个最为常见的Python发行版是Official Python——Python官方版本。它是由Python基金会开发和维护的。在这个版本里面它只包含了 Python的解释器以及Python的内置库，只有这两个东西！如果用它来做比如说数据运算、做一些分析的时候，可能会缺乏各种各样的工具包。比如说上节课里面提到过的numpy、pandas……而你如果要去手动安装这些工具包的话，确实可以装，但是对于新手来说很容易掉到各种各样的坑里面，而且这些坑有可能让你好几周都爬不出来，所以非常的麻烦。

尤其对于新手来说，不想去折腾这些安装什么的，就想用，对吧？那也就有人很聪明的打包了这么一个针对科学计算的版本：Anaconda。Anaconda是由美国的一家Anaconda公司专门针对科学计算打包的这么一个Python发行版。除了Offical Python里面的 Python解释器、官方内置库以外，还加上了用来做科学计算的很多库。

最后的VN Studio和Anaconda比较的话，更多是一个目的上的区别。VN Studio是由vn.py官方，也就是我们团队打包的Python发行版，**更多是针对量化交易方面的应用去设计的**。所以除了这些比如说用来做策略研究的numpy、pandas等等这些库以外，还额外打包了比如说vn.py以及很多的底层交易接口比如说TALib这样用来做技术指标运算的库、还有比如像FIX engine，用来对接FIX交易服务器的这样的一些库……因为这个课程是针对量化的一个Python入门课程，所以后面我们都会基于VN Studio来讲。

### **操作系统的建议方案**

第一步，先准备一个操作系统来运行Python。这里把它分成了两块：



第一块叫桌面环境。什么叫桌面环境？你的桌子上放着的电脑不管是笔记本还是台式机，它就是我们的桌面机器。一般在这样机器上（现在2020年）大家用的一般都是windows10了。当然了如果还在用老一点的windows8、windows7等等，理论上也可以运行VN Studio，但是毕竟所有的软件在操作系统上运行的时候都有一些操作系统依赖，万一你的windows版本可能正好太老了，有些工具没有，就跑不起来的话，我建议**最简单的解决方案还是做一个升级，升级到windows10就行了**。同时后面其他的一些程序等等也可以从中受益，应该来说还是性价比最高的方案。

另外一种选择就是云服务器了。这也是得益于现在比如说阿里云、腾讯、海外亚马逊AWS的普及，你随随便便就可以很方便的在网上注册一个云服务器，然后远程桌面登上去用它了。**对于云服务器我们统一推荐的都用 windows server的最新版本就是window server 2019**。server 2019的内核和windows 10是比较接近的，也是我们vn.py官方自己用的环境。

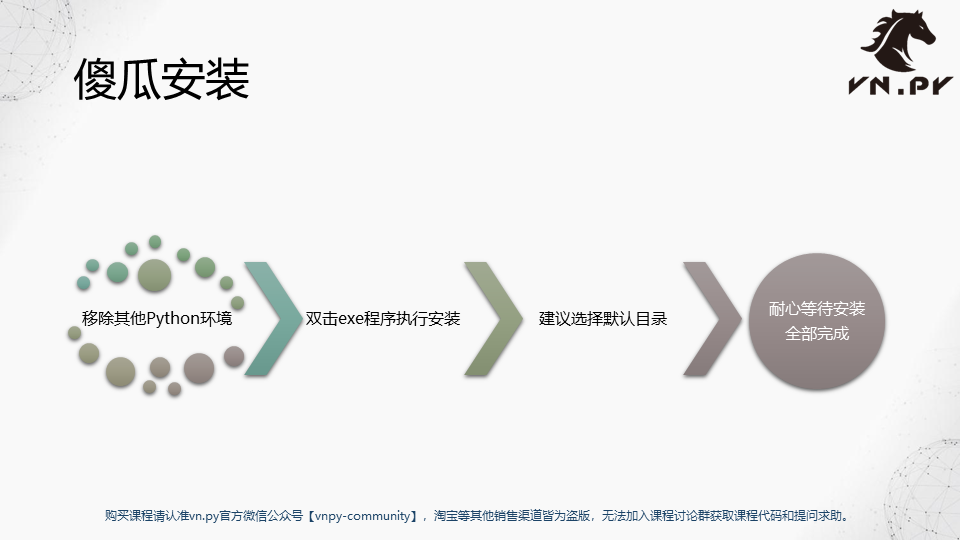
所以VN Studio在这两个操作系统上有最完善的测试，也建议大家如果自己要选的话，就选这两个操作系统。

### **安装VN Studio**

操作系统准备好了，下一步就可以来装VN Studio了。我们先去[www.vnpy.com](http://www.vnpy.como/)主页上下载（每发布一个新版本，版本号会变化，下载最新版本即可）：



下载最新版VN Studio安装包 ››双击运行 ›› VN Studio开始傻瓜安装



**安装前的一个重要提示**

在开始安装之前，我建议一件事情：如果你电脑上装过其他的python运行环境的话，请把它们都先移除掉。

之所以要这么做，是因为Python它本身在运行的时候需要有一些环境变量的支持，同时装多个Python环境有可能造成A环境和B环境之间的环境变量污染，就会导致一些很奇怪的行为：比如说启动不起来、一运行就崩溃等等。

学到本课程大概30多节课的时候，会讲到 Python环境变量的一些管理和配置的内容，到那个时候你会去掌握说“同时多python环境的时候要怎么样让它们共存、如何正确去管理。”但现在这部分内容对初学者来说太复杂了一点。所以现在我建议大家确保只安装一个运行环境，其他曾经安装过的都给卸载掉。等到后面学会了之后再把其他的给装回来就好了。

**安装目录的选择建议**

安装程序启动后你唯一要选的就是安装目录，推荐大家还是选默认目录。因为后面讲课的时候，我们在讲比如说在某个目录下的配置、或者说要去找到vn.py内部的某个源代码文件去看一看……都会以这个默认目录为准。你如果装到别的地方去的话，就是多一步，记得要把课程所说的默认目录改成你这自定义的目录。

最后就是耐心等待安装的全部完成了。这取决于电脑速度，快的可能5分钟就装好了，慢的可能要花个20分钟，老一点的机器也许要花半个小时。

### **四个你需要熟悉的名词解释**

最后我们来看一个叫做名词解释，因为这里我们主要列出了4个词：

vn.py

VN Studio

VN Station

VN Trader



对于很多刚入门或者刚接触vn.py用户来说，很有可能就遇到这四个词。都是vn开头的，大概根本不知道它们各自代表着什么。

这四个指代的是不同的东西，当然，之间肯定都有相关的。说白了这四个都是由我们vn.py官方开发出来的程序，但是不同的程序，也有不同的作用。

**vn.py**，注意vn.py所有的字母都是小写的，中间有一个点。开源社区里很多人喜欢一个个字母念成vnpy，也不能说错，但是官方可能更喜欢的名字是vn.py。vn.py是一个开源的、针对量化开发的一个框架，它也是Python的一个库，和numpy、pandas、matlab……都一样，都是一个包。只不过比如说numpy是针对矩阵、矩阵运算的pandas是针对时间序列的，tensorflow是针对机器学习的……vn.py主要针对的就是量化交易这么一个功能。

**VN Studio**，刚刚讲过它是一个Python的运行环境。这个运行环境是针对量化交易打包的，有Python的解释器、有Python的官方库，也有一系列用来做量化交易的三方库，都在这个里面。也就我们正在安装的东西。

**VN Station**，它是一个量化交易应用的启动器。什么意思呢？做量化交易的时候，有可能要启动交易程序、有可能要启动一个交互式的研究环境、有可能想对某个策略源代码做个加密，也可能纯粹只是想打开vn.py的官方网站去论坛提问，或者说有的时候vn.py官方有更新的时候你希望可以实现一键更新……么NV Station这个程序就是用来做这些事情的，它是一个启S动器，本身并不提供什么很复杂的功能，就是用来把别的那些东西给你启动起来的。所以在装完VN Studio之后，桌面上会出现一个白底黑色码头的图标，这个图标它的名字就叫VN Station。双击启动之后，需要微信扫码注册一下，注册完了之后直接登进去，就可以看到主界面。

如果你用VN Station启动器要去跑实盘的量化交易的话，此时你就会启动到VN Trader了。你在VN Station里点完启动之后会另外弹出一个窗口来。

**VN Trader**是一个全功能的量化交易平台，你可以用它去接CTP的交易接口、去做期货的实盘交易、可以去接比如火币或者ok的接口做数字货币的实盘交易、也可以接IB接口去做外盘的实盘交易；同时它内部也内嵌比如说CTA策略回撤、实时行情数据的录制到数据库等等一系列的应用的功能……以上这些是VN Trader这么一个量化交易平台所能提供的功能。

最后我们总结一下：

vn.py是开发框架，是一个包；

VN Studio是运行环境，是一整个刚刚装的这么一个Python的运行环境；

VN Station是一个启动器，可以把它理解为就是点完快捷方式之后弹出来一个小界面。你要启动什么就点一下相应按钮，它就能给你对应的启动出来。等于说就不用在桌面上放上一堆图标了。如果没有这个启动器的话，其实我们还有另外一个方案，就是每一个功能我给你在桌面放一个图标，这肯定就很麻烦，所以我们就专门做了这么一个启动器。

VN Trader更多是用来做实盘交易、做各种量化交易的时候用到的一个交易平台。

这四个名词即使我这么讲完，有些人还会有点懵，也不用担心，没关系的。因为在我们接下来课程里面会一点一点在实际操作里面接触到这不同的组件，到时候我们还会提到这些名字的。

在学到比如说到20节课-30节课的时候，我相信你对这些词就一定非常清楚了，不会再出现搞不清楚的情况。

这集的内容就先到这里，更多精华内容还是欢迎扫码关注我们的社区公众号，我们下节课再见。