

TOPICS Python Interactive ▼

使用“Python交互式”窗口



(https://github.com/Microsoft/vscode-docs/blob/master/docs/python/jupyter-support-py.md)

Jupyter (<http://jupyter-notebook.readthedocs.io/en/latest/>) (以前称为IPython Notebook) 是一个开源项目, 可让您轻松地在一个名为**Notebook**的画布上组合Markdown文本和可执行的Python源代码。Visual Studio Code支持本地 (/docs/python/jupyter-support)使用Jupyter Notebook (/docs/python/jupyter-support), 以及通过Python代码文件。本主题涵盖通过Python代码文件提供的支持, 并演示如何:

- 使用类似Jupyter的代码单元
- 在Python交互式窗口中运行代码
- 使用变量浏览器和数据查看器查看, 检查和过滤变量
- 连接到远程Jupyter服务器
- 调试Jupyter笔记本
- 导出Jupyter笔记本

要使用Jupyter笔记本, 您必须在VS Code中激活Anaconda环境, 或在其中安装Jupyter软件包的 (<https://pypi.org/project/jupyter/>)另一个Python环境中激活 (<https://pypi.org/project/jupyter/>)。要选择环境, 请使用**Python: 从命令面板中选择“解释器”**命令 (Ctrl + Shift + P)。

激活适当的环境后, 您可以创建和运行类似Jupyter的代码单元, 连接到远程Jupyter服务器以运行代码单元, 并将Python文件导出为Jupyter笔记本。

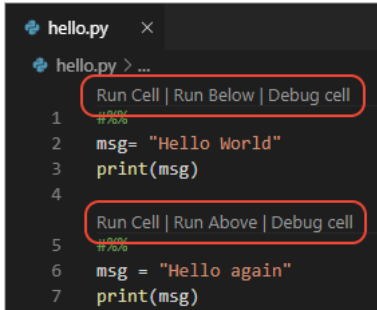
Jupyter代码单元

您可以使用 `# %%` 注释在Python代码中定义类似Jupyter的代码单元:

```
# %%  
msg = "Hello World"  
print(msg)  
  
# %%  
msg = "Hello again"  
print(msg)
```

注意: 请确保将上面显示的代码保存在扩展名为.py的文件中。

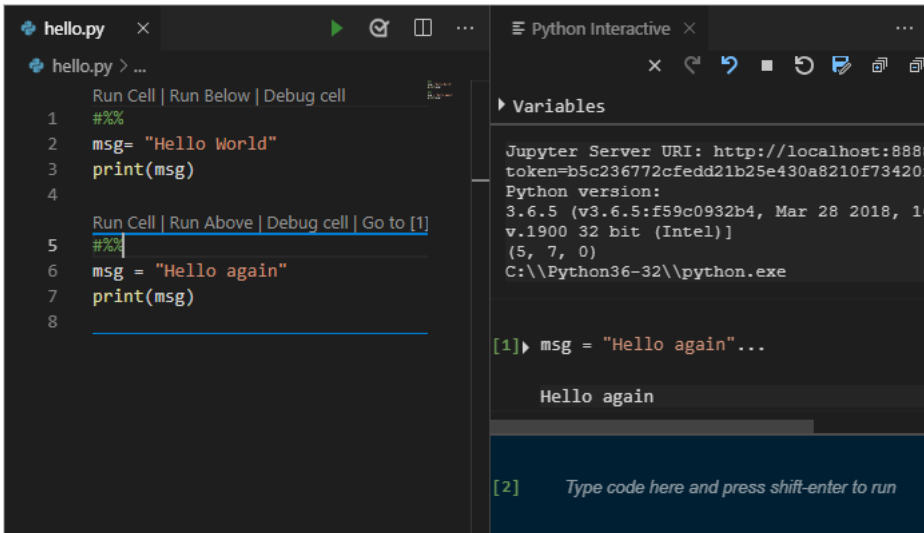
当Python扩展程序检测到代码单元时, 它将添加“**运行单元**”和“**调试单元**”CodeLens“装饰。第一个单元格还包括“**在下面运行**”, 所有后续单元格都包括“**在上面运行**”:



注意: 默认情况下, “**调试单元**”仅进入用户代码。如果要进入非用户代码, 则需要在Python扩展设置 (Ctrl + ,) 中取消选中Data Science: Debug Just My Code。

运行单元仅适用于一个代码单元。显示在第一个单元格上的“**在以下运行**”将运行文件中的所有代码。“**以上运行**”适用于所有代码单元, 但不包括带有装饰的单元。例如, 您将使用Run Above来在运行特定单元之前初始化运行时环境的状态。

选择命令将启动Jupyter (如有必要, 可能需要一分钟), 然后在Python Interactive窗口中运行适当的单元格:



您还可以使用（Ctrl + Enter）或Python：在Python Terminal中运行“选择/行”命令（Shift + Enter）来运行代码单元。使用此命令后，Python扩展程序会自动将光标移动到下一个单元格。如果您位于文件的最后一个单元格中，则扩展名会自动 # %% 为新单元格插入另一个定界符，以模仿Jupyter笔记本的行为。

您也可以单击行号左侧的边距来设置断点。然后，您可以使用“调试单元”为该代码单元启动调试会话。调试器在断点处停止执行，并允许您一次单行执行代码并检查变量（有关详细信息，请参见调试 (debugging.md)）。

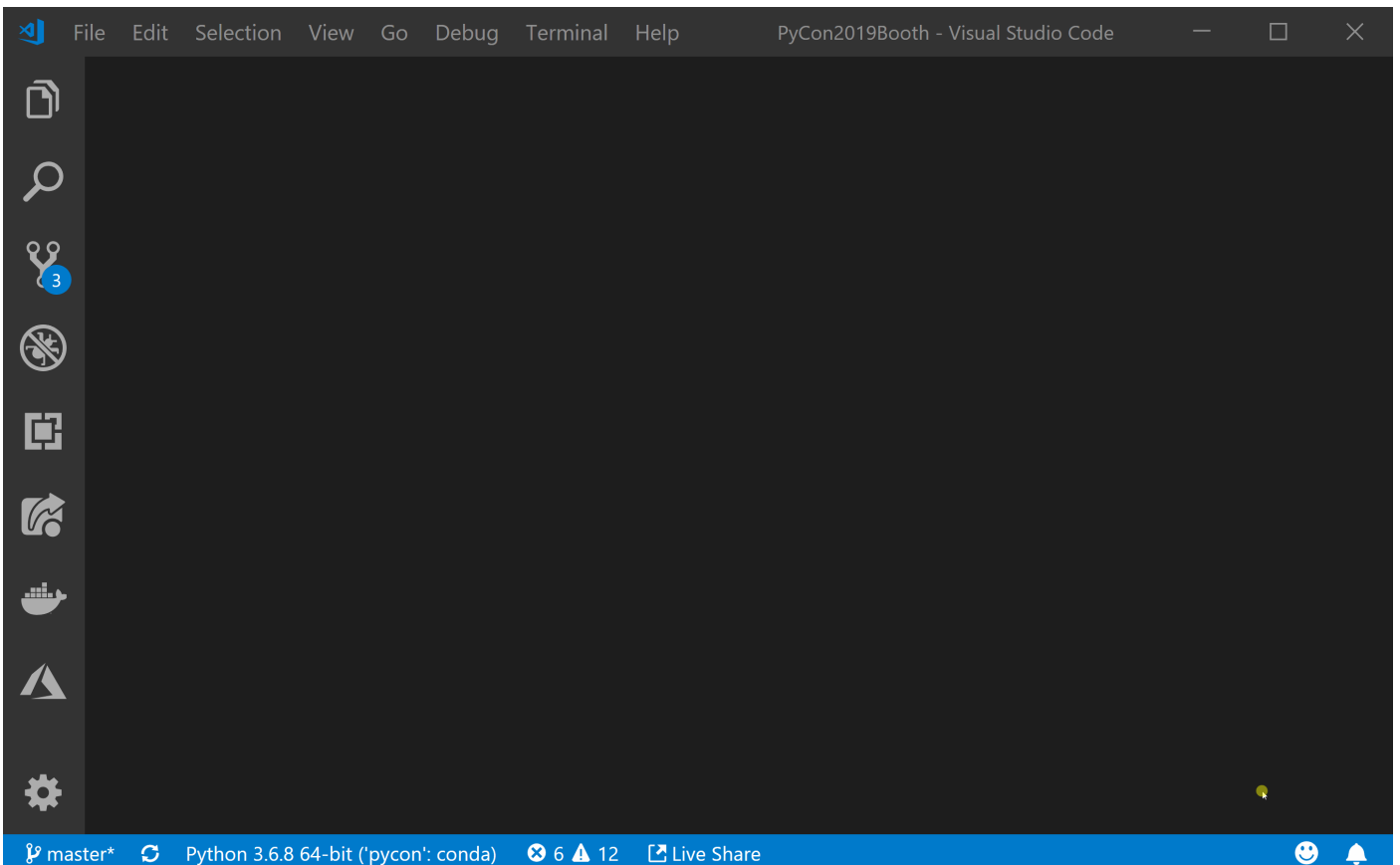
Python交互式视窗

上一节中提到的Python Interactive窗口可以用作带有任意代码（带有或不带有代码单元）的独立控制台。要将窗口用作控制台，请使用Python将其打开：“命令面板”中的“显示Python交互式”窗口命令。然后，您可以键入代码，使用Enter键转到新行，然后按Shift + Enter键运行代码。

要将窗口与文件一起使用，请使用“命令面板”中的“在Python Interactive中运行当前文件”窗口命令。

智能感知

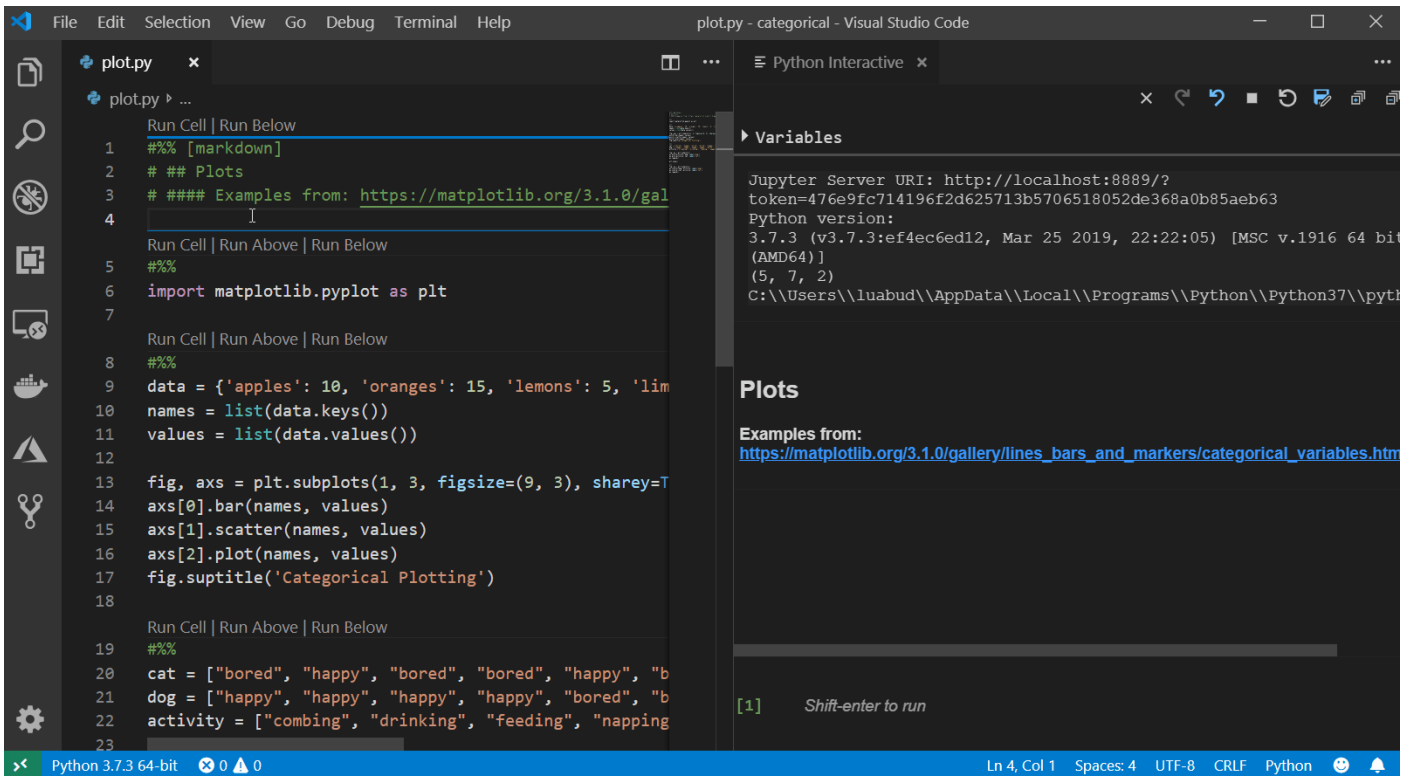
Python Interactive窗口具有完整的IntelliSense -代码完成，成员列表，方法的快速信息以及参数提示。您可以像在代码编辑器中一样在Python Interactive窗口中高效地键入内容。



绘图查看器

绘图查看器使您能够更深入地处理绘图。在查看器中，您可以平移，缩放和浏览当前会话中的图。您还可以将图导出为PDF，SVG和PNG格式。

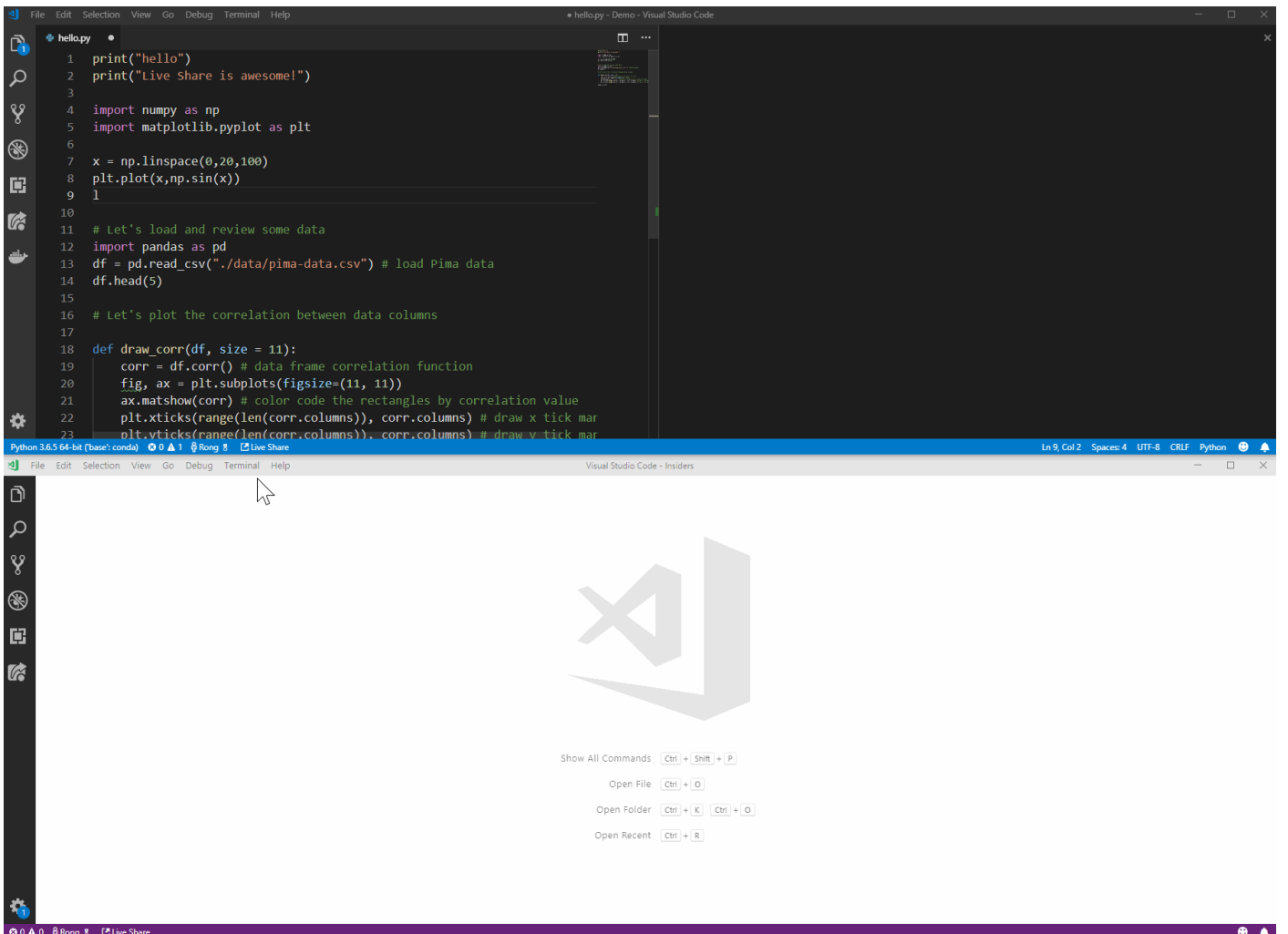
在“Python交互式”窗口中，双击任何图以在查看器中将其打开，或选择图左上角的展开按钮。



注意：“Python交互式”窗口支持使用matplotlib (<https://matplotlib.org/>)和Altair (<https://altair-viz.github.io/index.html>)创建的渲染图。

实时共享Python交互式

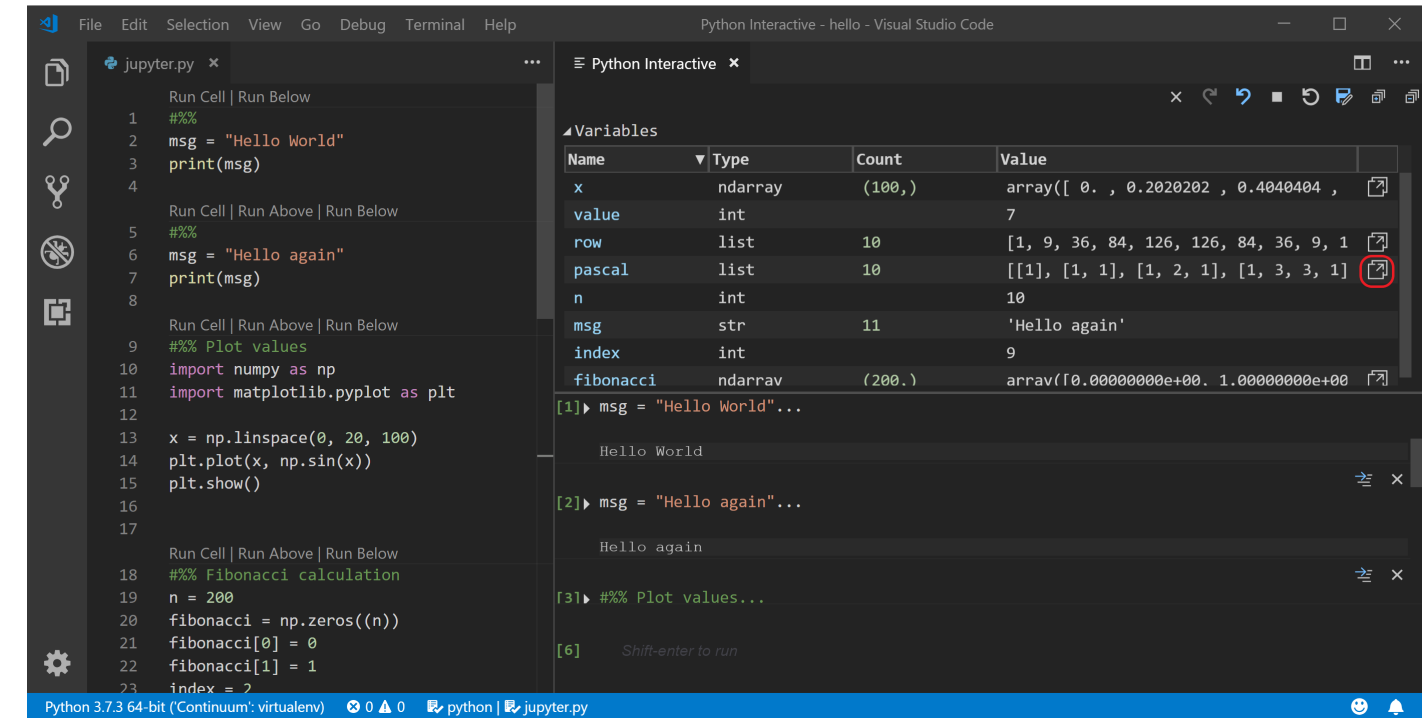
“Python交互式”窗口还支持Visual Studio Live Share (<https://visualstudio.microsoft.com/services/live-share/>)进行实时协作。Live Share使您可以在共享音频，服务器，终端，差异，评论等进行共同编辑和共同调试。



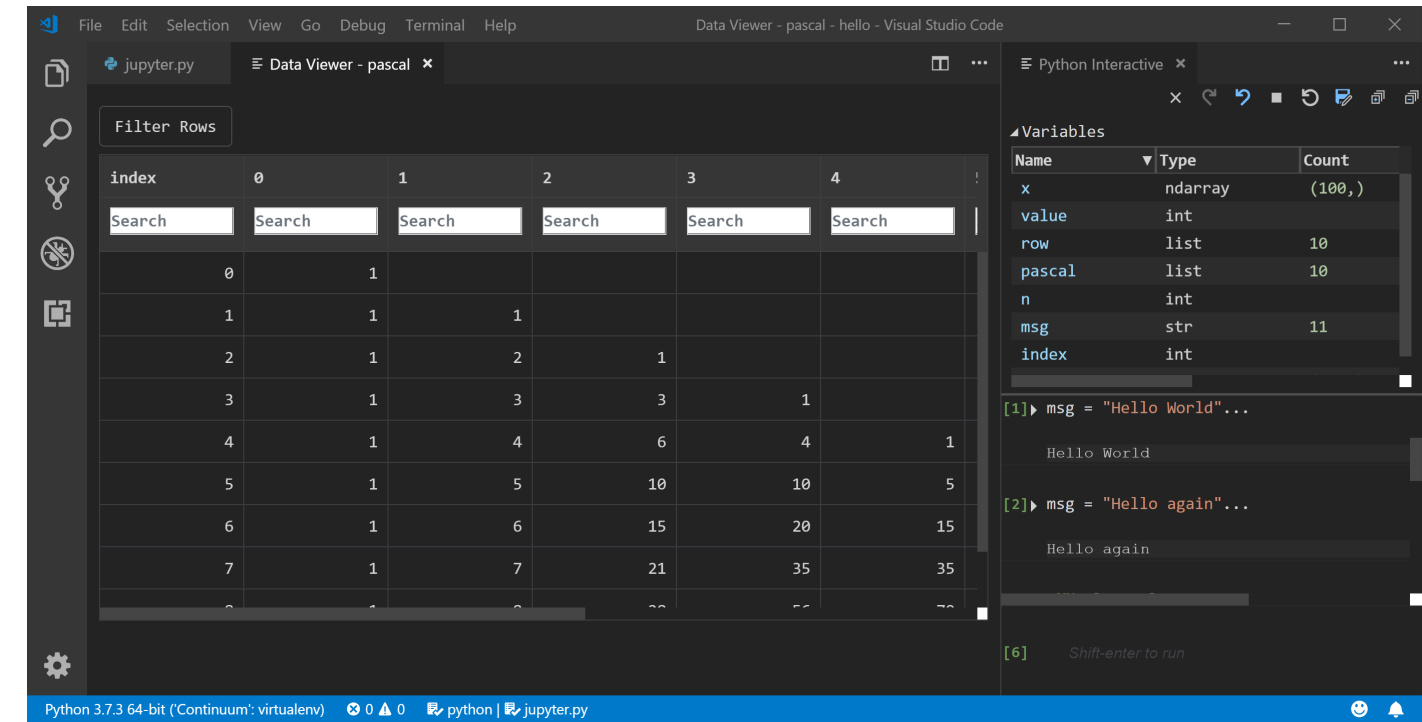
此功能需要在主机和来宾计算机上都安装Live Share扩展 (<https://marketplace.visualstudio.com/items?itemName=MS-vsliveshare.vsliveshare-pack>)。

变量浏览器和数据查看器

在“Python交互式”窗口中，可以在当前的Jupyter会话中查看，检查和过滤变量。通过在运行代码和单元格之后展开“变量”部分，您将看到当前变量的列表，当在代码中使用变量时，这些列表将自动更新。



有关变量的其他信息，您也可以双击一行或使用“在数据查看器中显示变量”按钮在“数据查看器”中查看变量的更详细视图。打开后，您可以通过搜索各行来过滤值。



注意：默认情况下，变量资源管理器处于启用状态，但可以在设置中关闭（Python>数据科学：显示Jupyter变量资源管理器）。

连接到远程Jupyter服务器

您可以通过连接到远程Jupyter服务器，将Jupyter笔记本中的密集计算卸载到其他计算机上。连接后，代码单元将在远程服务器而不是本地计算机上运行。

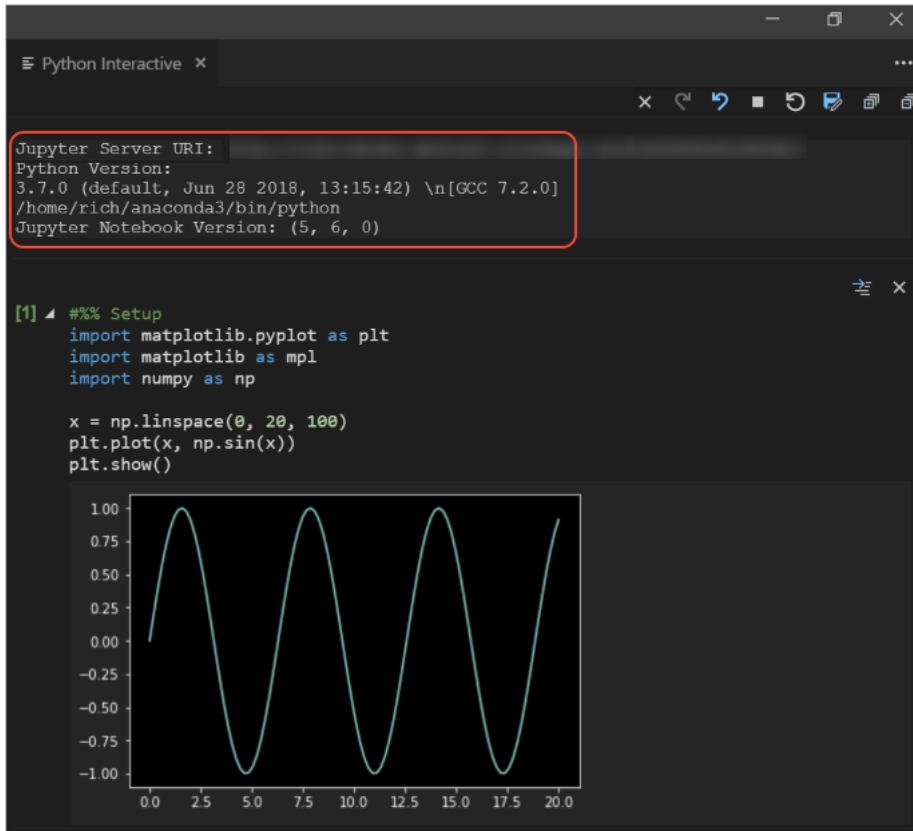
要连接到远程Jupyter服务器：

1. 运行Python：从命令面板（Ctrl + Shift + P）指定Jupyter服务器URI命令。
2. 出现提示时，请提供服务器的URI（主机名）以及 ?token= URL参数随附的身份验证令牌。（如果在启用了身份验证令牌的VS Code终端中启动服务器，则带有令牌的URL通常会出现终端输出中，您可以在其中进行复制。）或者，您可以在提供URI之后指定用户名和密码。

```
https://hostname:8080/?token=849d61a414abafab97bc4aab1f3547755ddc232c2b8cb7fe
```

Enter the URI of a Jupyter server (Press 'Enter' to confirm or 'Escape' to cancel)

3. “Python交互式”窗口通过显示URI（在下图中模糊显示）指示在哪里运行代码：



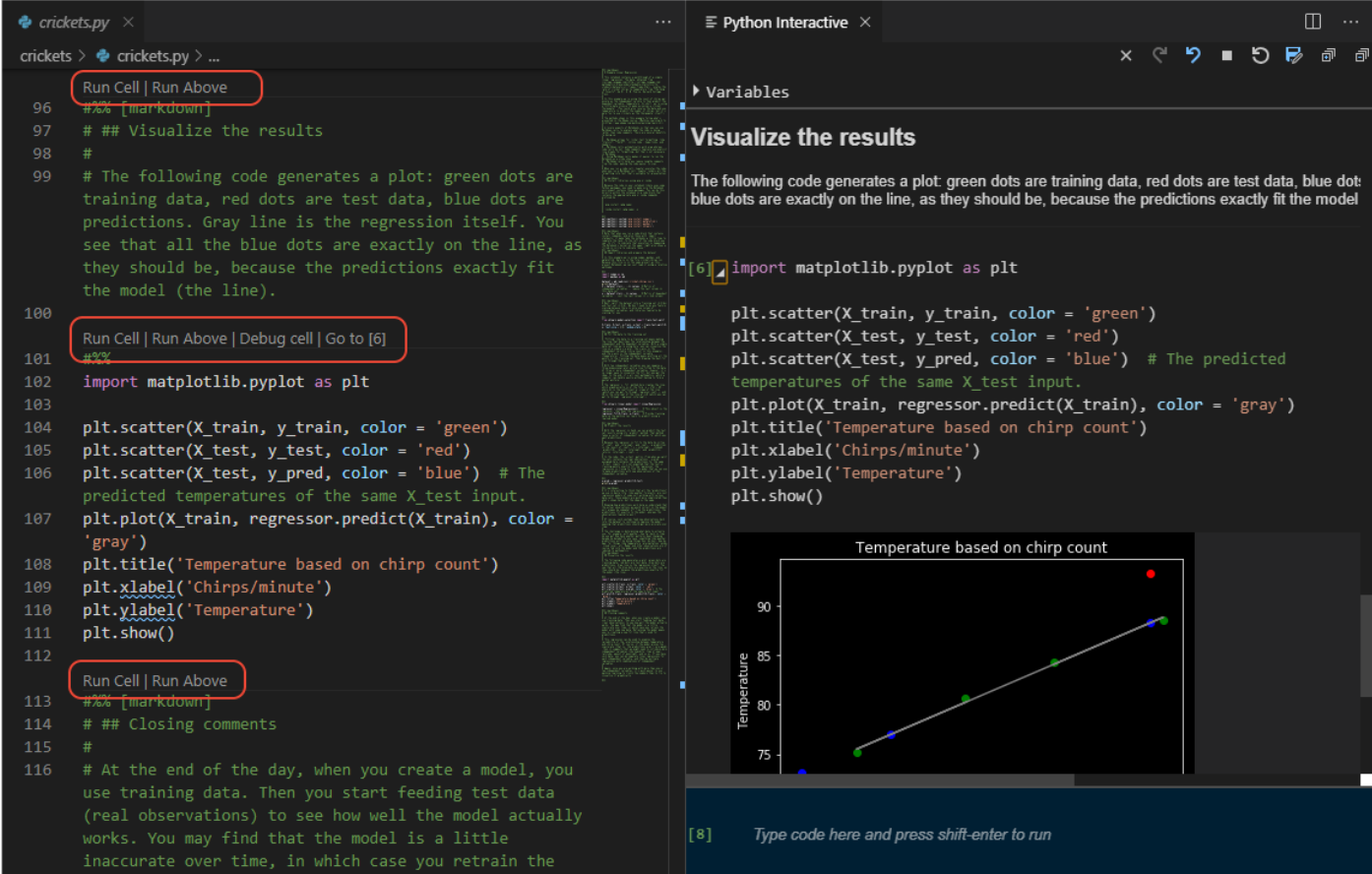
将Jupyter笔记本转换为Python代码文件

激活安装了Jupyter的环境后，可以 .ipynb 在VS Code中打开Jupyter笔记本文件（），然后将其转换为Python代码。转换文件后，您可以像处理其他任何Python文件一样运行代码，也可以使用VS Code调试器。在VS Code中打开和调试笔记本是查找和解决代码错误的便捷方法，这在Jupyter笔记本中很难直接完成。

当您打开笔记本文件时，Visual Studio Code将自动在笔记本编辑器中将其打开。使用工具栏上的转换图标将Notebook（.ipynb）文件转换为Python文件（.py）。



选择转换图标，等待几秒钟，然后VS Code以无标题的文件打开转换后的笔记本。笔记本的单元格在Python文件中用 # %% 注释界定；Markdown单元格将全部转换为以开头的注释 # %% [markdown]，并在交互式窗口中与HTML和代码和输出（如图形）一起呈现为HTML：



注意：首次在Python文件中运行代码单元时，Python扩展会启动Jupyter服务器。服务器启动可能需要一些时间，并且Python交互窗口可能会显示代码结果。

调试Jupyter笔记本

Visual Studio Code调试器使您可以单步执行代码，设置断点，检查状态并分析问题。使用调试器是查找和更正笔记本代码中问题的有用方法。

- 如本文开头所述，在VS Code中，激活安装了Jupyter的Python环境。
- .ipynb 如上一节所述，将笔记本的文件导入VS Code。（如果您使用的是基于云的Jupyter环境（例如Azure Notebooks，<https://docs.microsoft.com/azure/notebooks>）请先下载文件。）
- 要启动调试器，请使用以下选项之一：
 - 对于整个笔记本，请打开“命令面板”（Ctrl + Shift + P），然后运行“Python：在Python交互式窗口中调试当前文件”命令。
 - 对于单个单元，请使用显示在单元上方的“调试单元”装饰。调试器专门从该单元中的代码开始。默认情况下，调试单元仅进入用户代码。如果要进入非用户代码，则需要在Python扩展设置（Ctrl + ,）中取消选中Data Science: Debug Just My Code。
- 要熟悉VS Code的常规调试功能，例如检查变量，设置断点和其他活动，请查阅VS Code调试（</docs/editor/debugging>）。
- 找到问题后，请停止调试器，更正代码，保存文件，然后再次启动调试器。
- 当您对所有代码都正确感到满意时。保存文件，然后按照以下部分中的说明导出笔记本。然后，您可以将笔记本上载到正常的Jupyter环境。

导出Jupyter笔记本 #

除了打开Jupyter笔记本之外，您还可以使用命令面板（Ctrl + Shift + P）中的以下命令之一，将内容从VS Code中的Python文件导出到Jupyter笔记本（带有.ipynb扩展名）。

- Python：将当前Python文件导出为Jupyter Notebook：使用# %%和# %% [markdown]分隔符指定各自的单元格类型，从当前文件的内容创建Jupyter Notebook。
- Python：将当前的Python文件导出并输出为Jupyter Notebook：从当前文件的内容创建Jupyter Notebook，并包括代码单元的输出。
- Python：将Python Interactive窗口导出为Jupyter Notebook：从Python Interactive窗口的内容创建Jupyter Notebook。

导出内容后，VS Code将显示提示，您可以通过该提示在浏览器中打开笔记本。

该文档对您有帮助吗？

是 没有