# 目标：

编写、运行和调试Python“Hello World”应用

创建Python虚拟环境并安装包

编写一个简单的Python代码作图

# Prerequisites先决条件

安装VS Code

安装Python扩展包

安装Python3

## 安装VS Code和Python扩展包

## 安装Python解析器

windows版本：从python.org获取在window安装。

在命令提示器中核实当前系统python的版本：



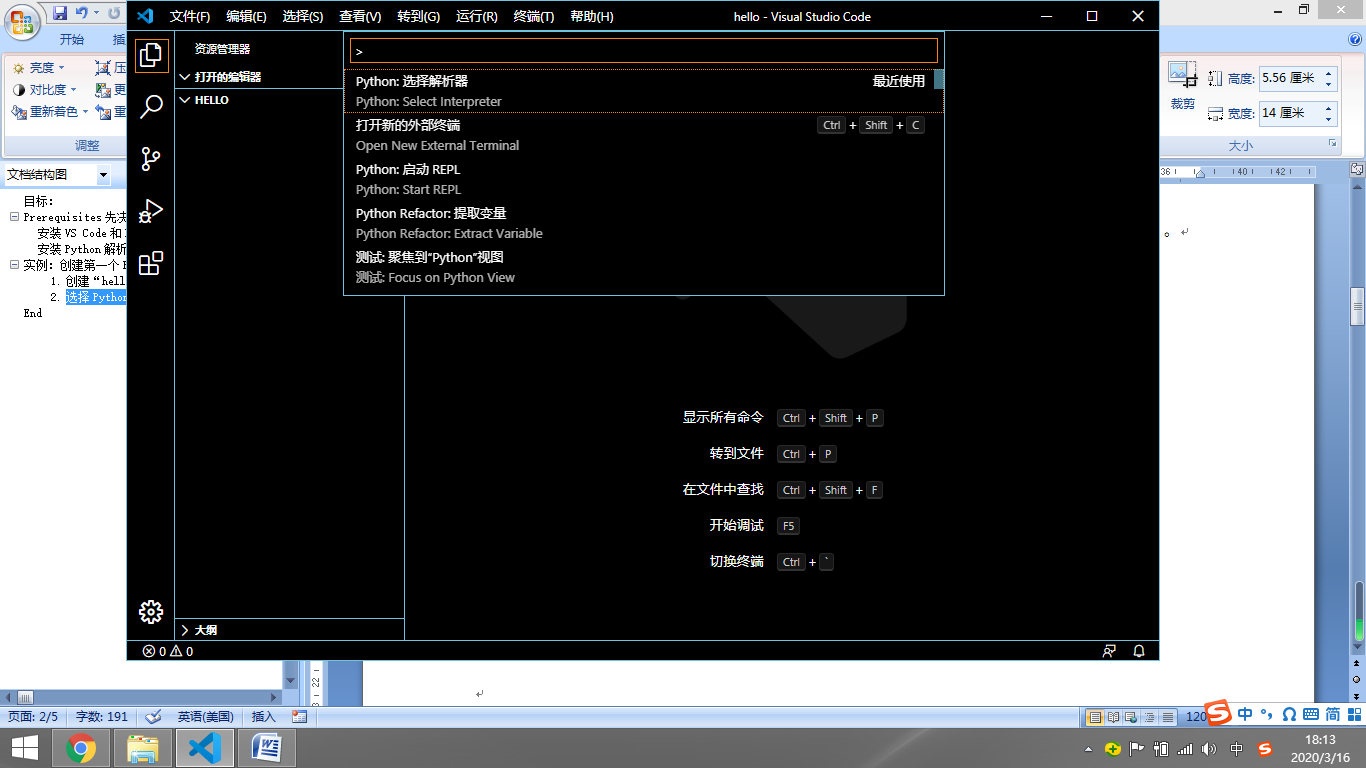
# 实例：创建第一个Python程序“Hello”

### 创建“hello”文件夹

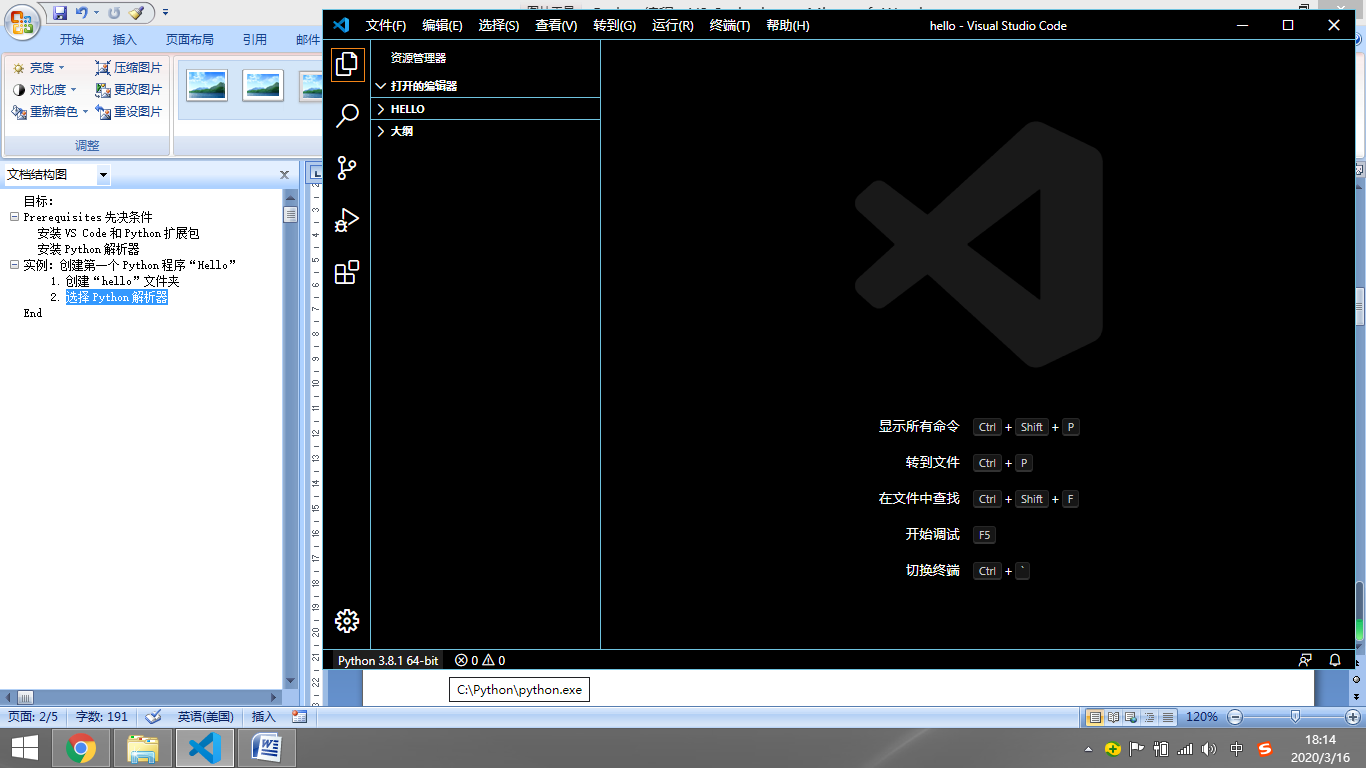
或在VS Code中“**文件 > 打开文件夹**”（**File > Open Folder**）

### 选择Python解析器

打开“命令面板”（**Command Palette**），输入“Python: Select Interpreter”进行搜索。

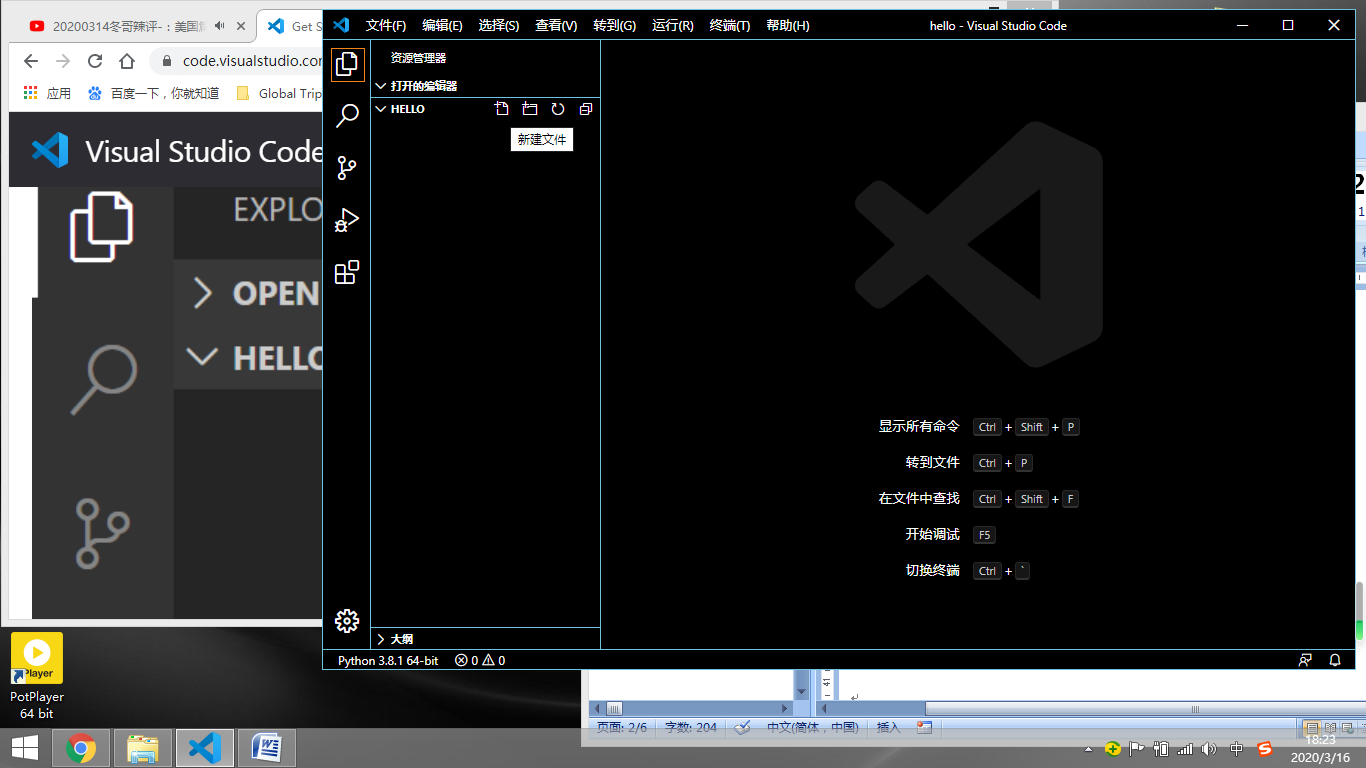
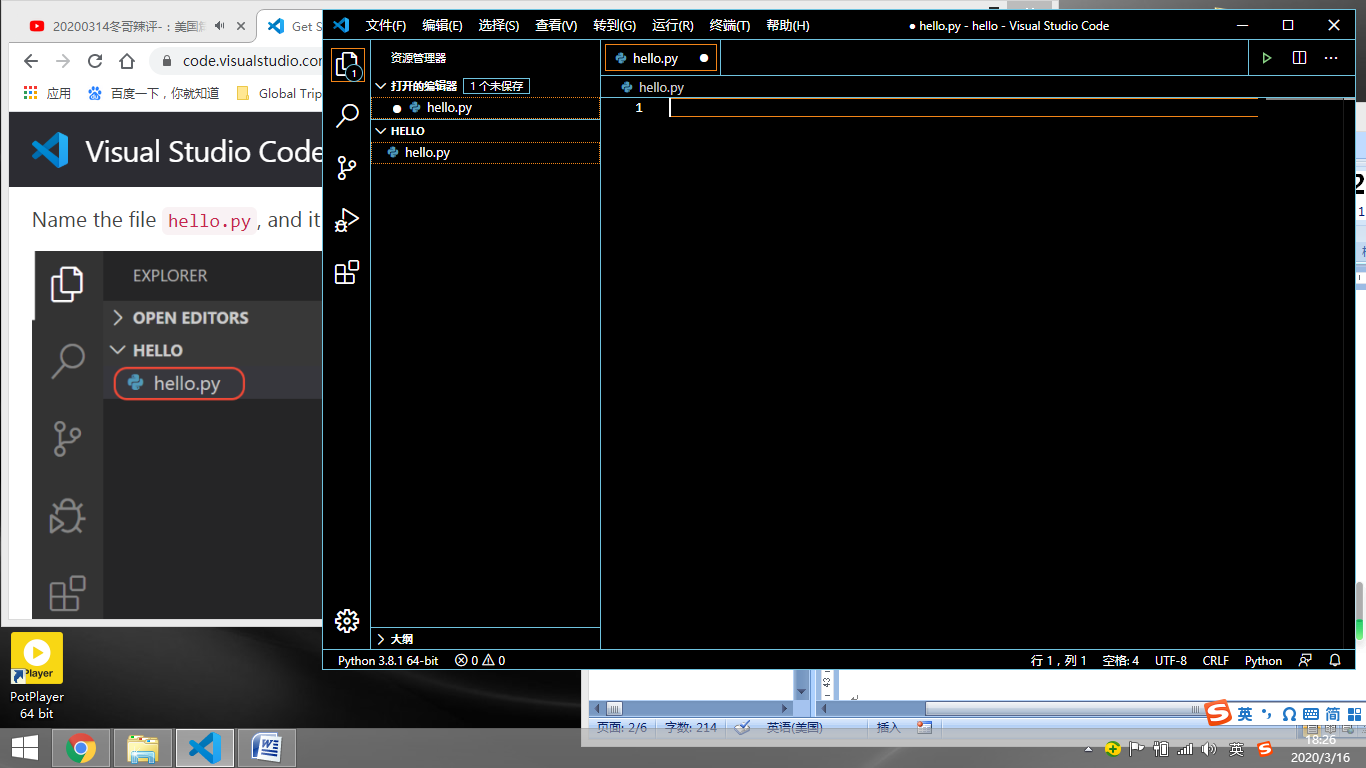


或者在状态栏中选择：



### 创建Python代码文件“Hello World”

在hello文件夹创建hello.py文件

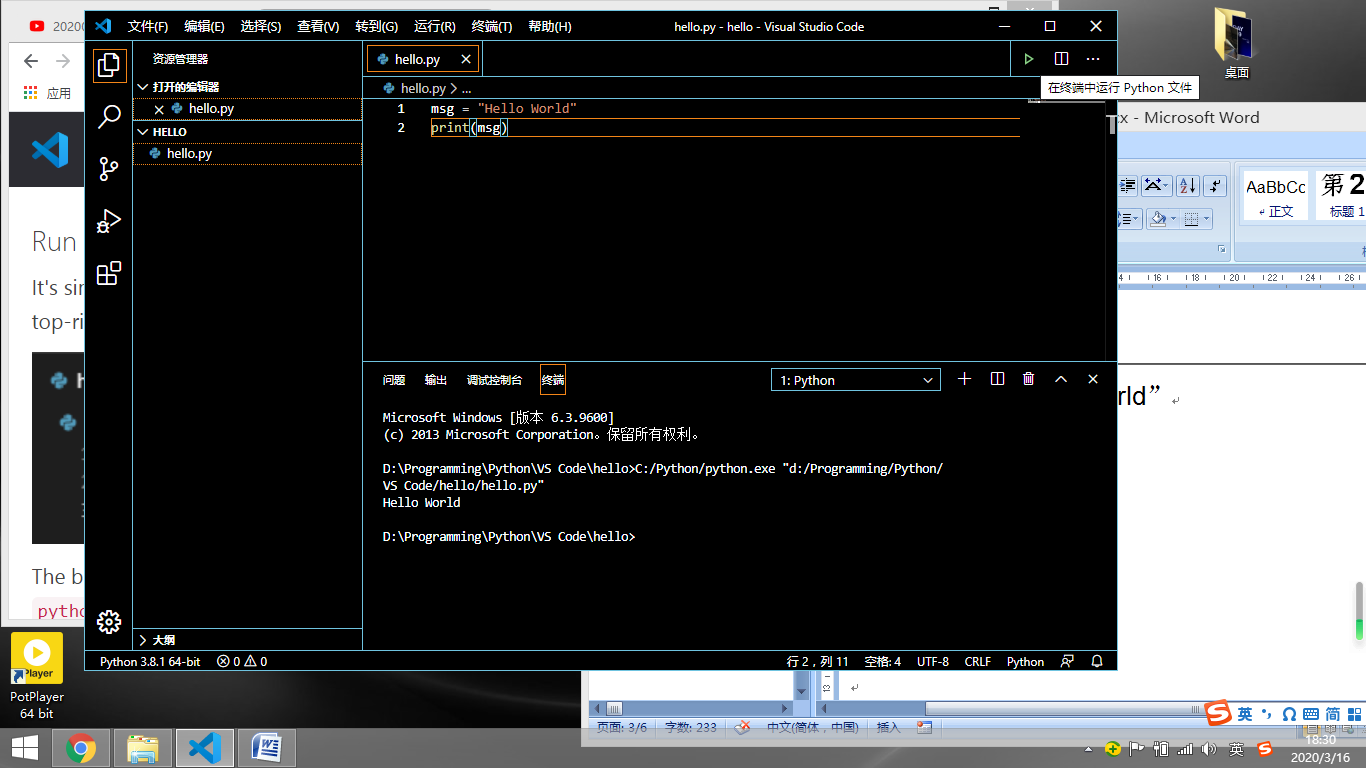
 

输入代码：

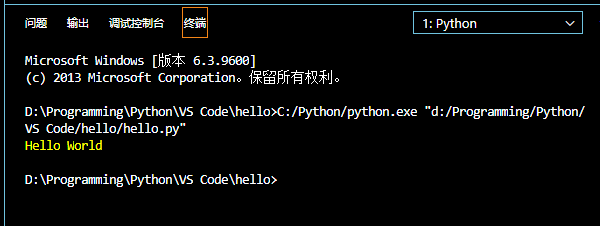
msg = "Hello World"

print(msg)

### 运行代码“Hello World”



运行的结果在“终端”（Terminal）中显示：

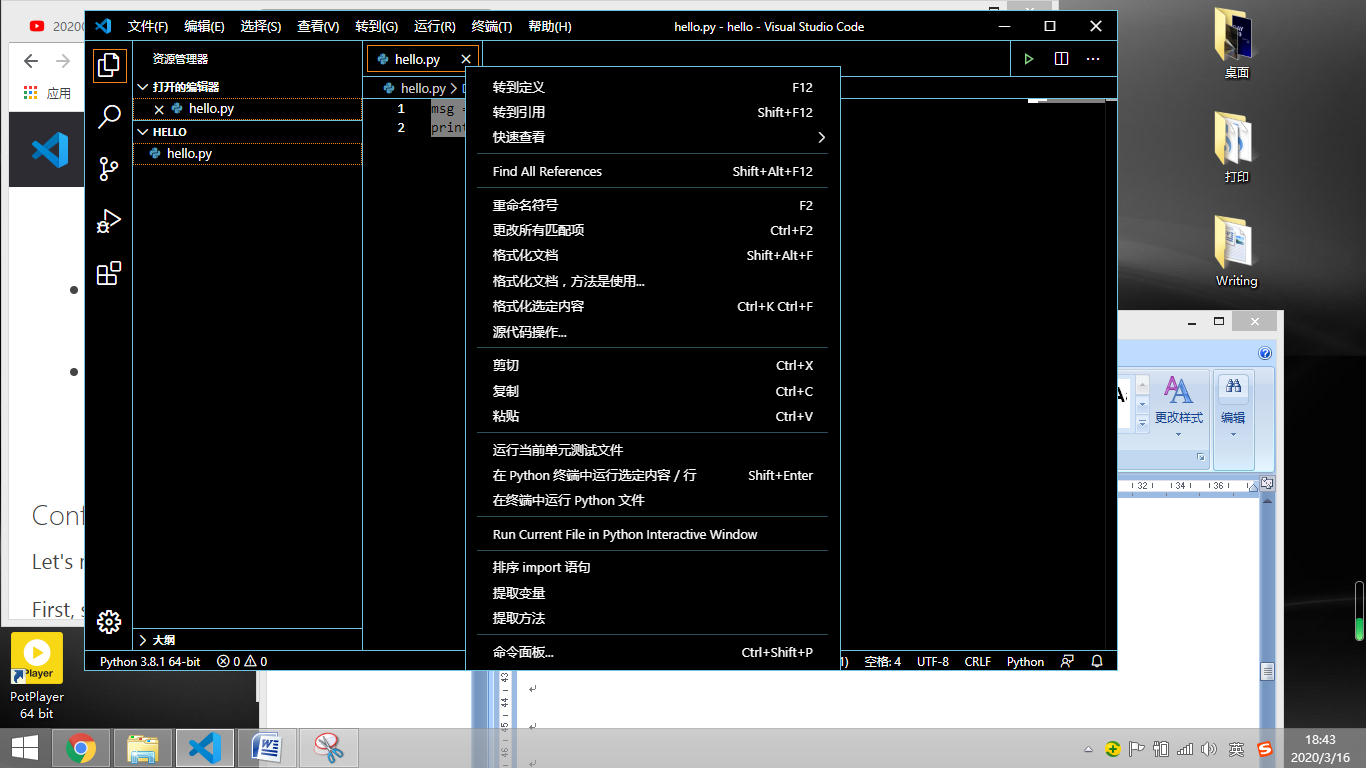


另外3中运行py代码的方式：

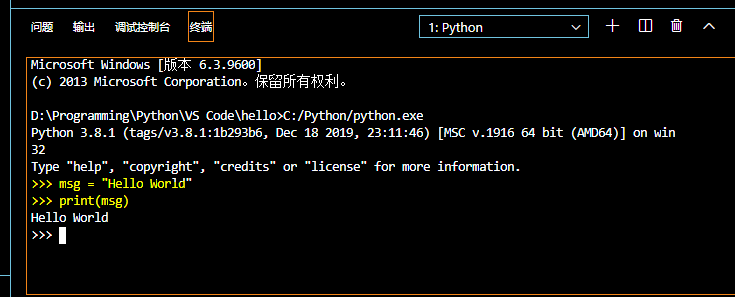
1. 右键菜单“在终端中运行Python文件”



1. 运行指定代码行。选择指定代码行，组合键Shift+Enter或者右键“在Python终端中运行选定内容/行”

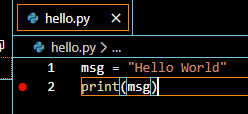


1. 在终端中直接输入代码运行

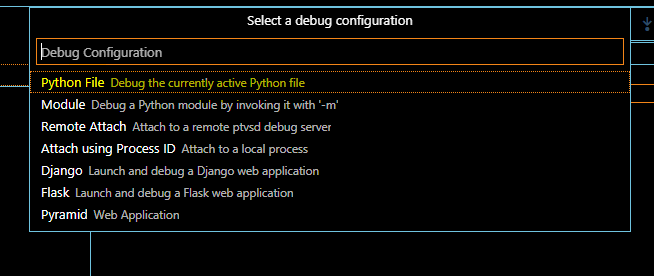


### 配置并运行调试器

1.在代码行中设置断点

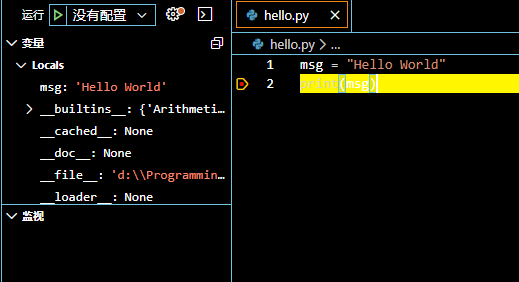


2.首次调试时，需要选择调试器配置。按下“F5”



选择Python File调试器。VS Code会自动创建配置文件launch.json。

3.调试。按下“F5”，运行程序直到断点位置。查看调试窗口，可以查看本地的变量。



在调试模式下，在调试控制台中输入变量或变量方法，可以查看变量值和方法结果：



# 实例：使用python绘图

### 创建Python文件standardplot.py

import matplotlib.pyplot as plt

import numpy as np

x = np.linspace(0, 20, 100)  # Create a list of evenly-spaced numbers over the range

plt.plot(x, np.sin(x))       # Plot the sine of each x point

plt.show()                   # Display the plot

此时，运行代码会弹出异常"ModuleNotFoundError: No module named 'matplotlib'"

原因是没有安装matplotlib库。

### 配置专用虚拟环境

tips：一般不会将使用的库安装到通用解析器环境中，而是安装到项目专用的虚拟环境。一旦激活该虚拟环境，则安装的库之间是相互独立，不会冲突的。

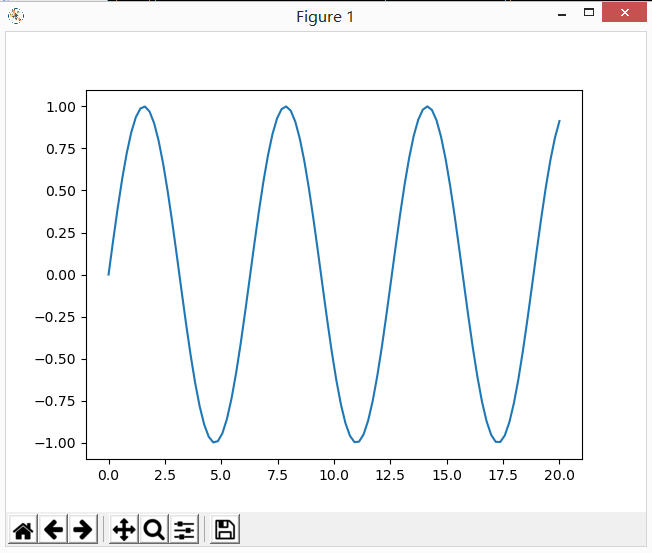
1.进入工作区（workspace）的文件夹

2.创建虚拟环境，在终端中（Terminal）输入命令：py -3 -m venv .venv

3. 选择解析器，在“命令面板”（**Command Palette**），输入“Python: Select Interpreter”。

4.安装库，在终端中输入命令：python -m pip install matplotlib

5.运行代码：



6.当需要关闭虚拟环境时，在终端中输入：deactivate。

# 总结Summary：

先决条件：

* 安装VS Code和Python扩展包
* 安装Python到OS

步骤：

1.创建project文件夹

2.选择Python解析器

3.进入project文件夹，配置专用虚拟环境：py -3 -m venv .venv

4.安装专用库：python -m pip install matplotlib

5.在project文件夹下创建py文件

6.运行和调试

# 参考：

## 创建虚拟环境：

python3 -m venv */path/to/new/virtual/environment*

* -m : run library module as a script (terminates option list)
* venv 是创建虚拟环境的标准工具，从 Python 3.3 开始成为 Python 的组成部分。 从 Python 3.4 开始，它会默认安装 pip 到所创建的全部虚拟环境。
* virtual environment 是一种半隔离的 Python 环境，允许为特定的应用安装各自的包，而不是安装到整个系统。

## 安装库：

python -m pip install *SomePackage*

* pip 是首选的安装程序。从Python 3.4开始，它默认包含在Python二进制安装程序中。

*pip install [options] <requirement specifier> [package-index-options] ...*

*pip install [options] -r <requirements file> [package-index-options] ...*

*pip install [options] [-e] <vcs project url> ...*

*pip install [options] [-e] <local project path> ...*

*pip install [options] <archive url/path> ...*

# End