Python Editors (IDE)

There are many free and commercial editors available for Python. Here, we will learn how to use some open-source editors to execute Python scripts or statements.
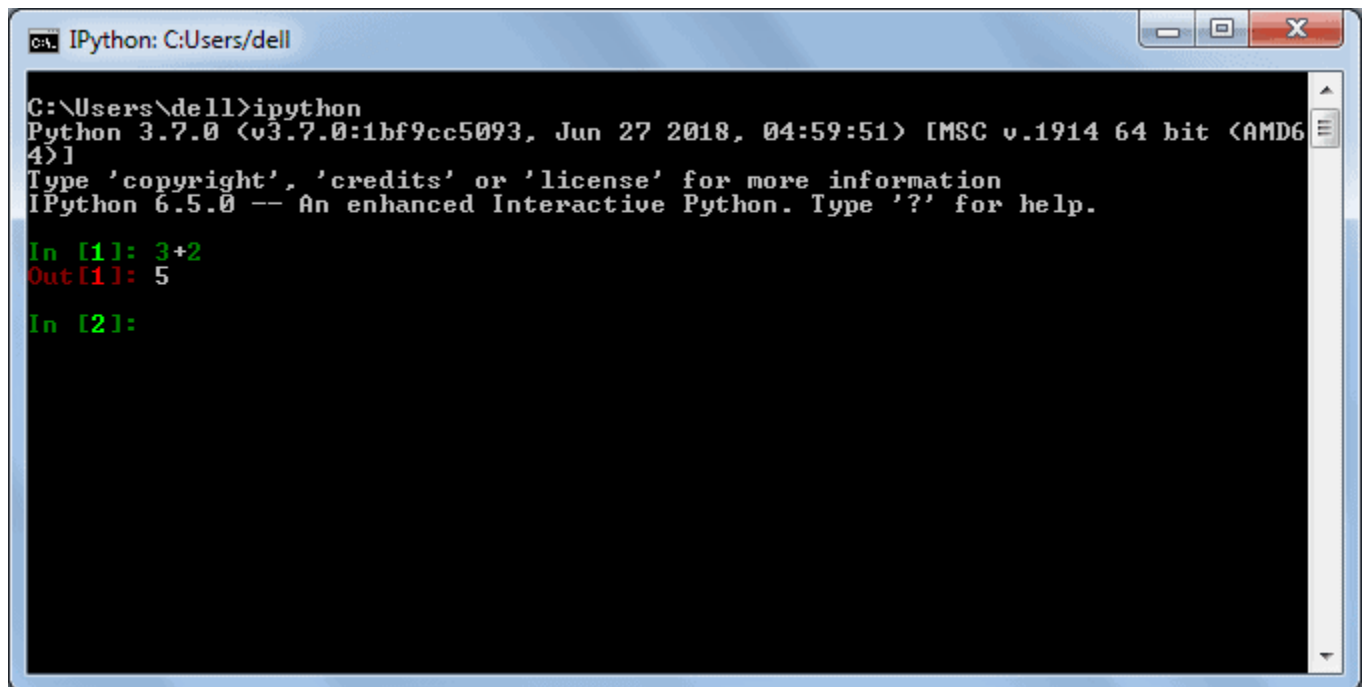
IPython

IPython stands for Interactive Python. Created by Fernando Perez in 2001, IPython is an enhanced read-eval-print loop (REPL) environment particularly suitable for scientific computing. It offers a fully compatible replacement for the standard Python interpreter, with convenient shell features, special commands, command history mechanism and output results caching.

To install IPython, execute the pip3 command which is the in-built Python package installer.

Execute the pip3 install ipython command in the command prompt to install iPython, as shown below.

C:\Users\{user name}>pip3 install ipython

After successful installation, invoke IPython from the command prompt by entering ipython and pressing enter. You can then execute the Python statement as we execute it below in command prompt.

Execute Python Statements in IPython

Jupyter Notebook

The Jupyter Notebook is a browser-based graphical interface to the IPython shell. It allows the user to include formatted text, static and dynamic visualizations, mathematical equations, JavaScript widgets etc. along with the Python code. The Jupyter Notebook document can be exported to PDF, Python script or HTML.

By default, the IPython kernel drives the Jupyter Notebook application. However, it supports other languages like Julia and R. (Jupyter stands for **JU**lia, **PYT**hon and **R**).

To install Jupyter, use the pip utility shipped with Python software.

C:\Users\{user name}>pip3 install jupyter

After successful installation, we can start the Jupyter editor from the command prompt as below.

C:\Users\{user name}>jupyter notebook

Jupyter Notebook is a client-server application. The server is deployed on the localhost's default port 8888 and the client is opened in a browser window as shown below:
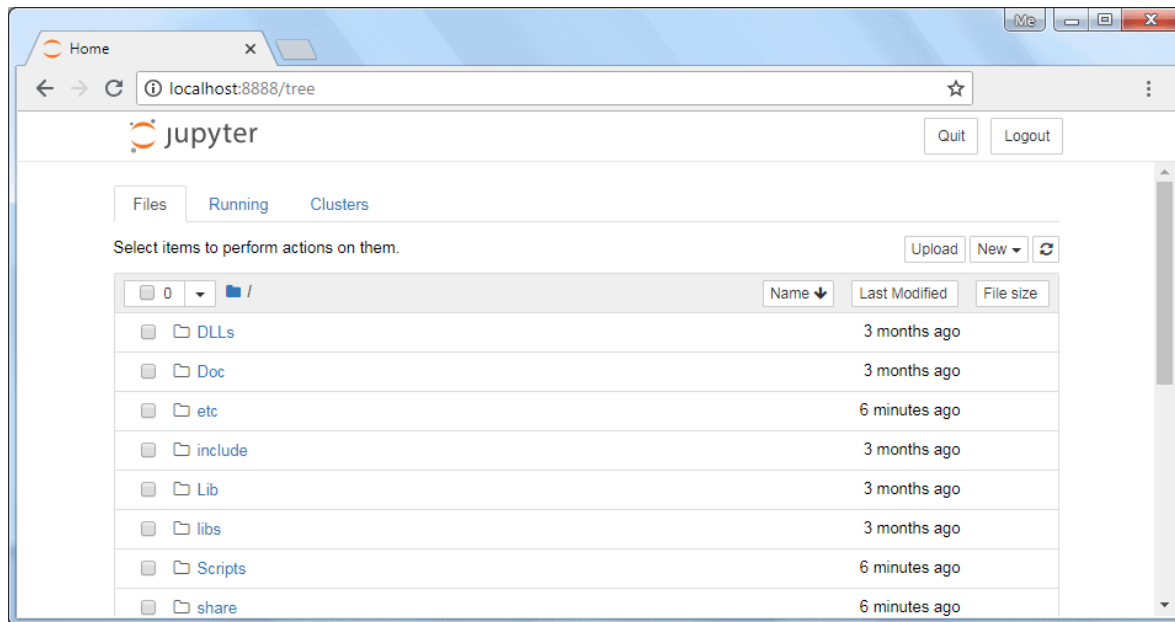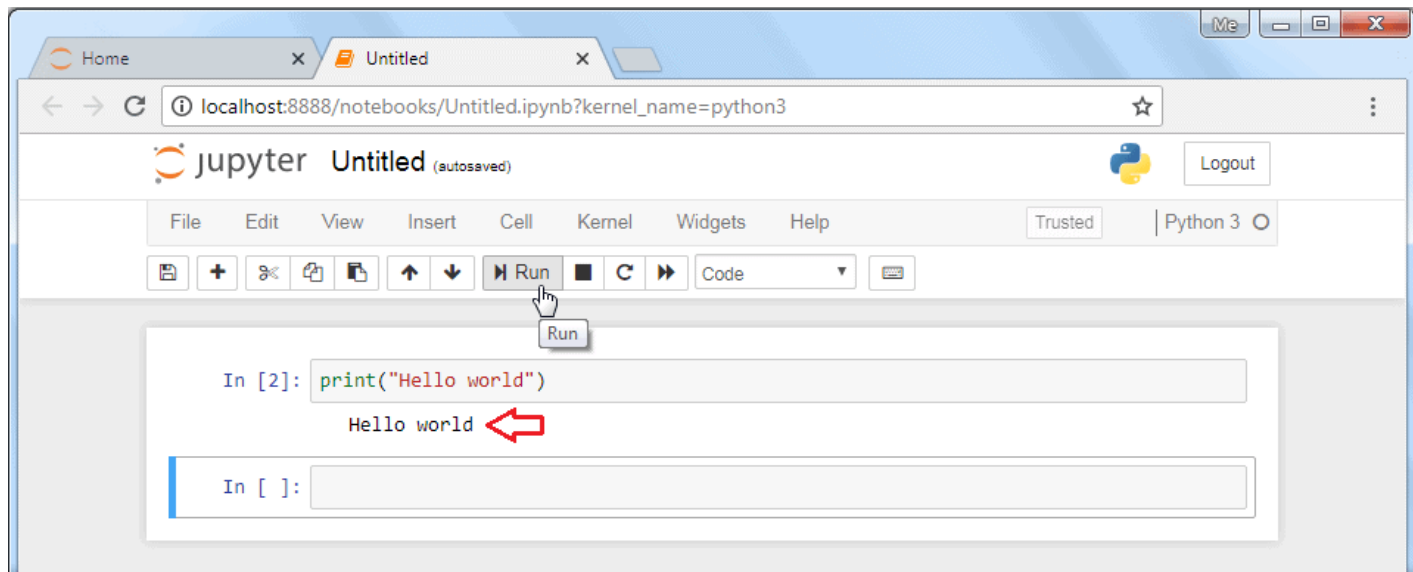

Jupyter

As you can see, Jupyter will display files and folders from the Python installation folder. You can create, open and execute python scripts from the appropriate folders. Start a new notebook by choosing Python 3 from the "new" drop down as shown below:

New Python Script in Jupyter

This will open another window to enter python statements and run them as shown below.

New Python Script in Jupyter

The interface is similar to IPython shell. However, there are a lot of other advantages.

For instance, you can insert and delete cells. A cell can contain code, heading or a markdown text which acts as documentation. The code in any cell can be run. Another advantage is that data visualizations generated by libraries like Matplotlib can be incorporated inline.

The notebook is saved with the .ipynb extension. It can be exported to HTML or PDF format so that it can be shared.

Virtual Environment

Many times, Python packages developed by certain third-parties have to be installed while developing Python-based applications. However, requirement for a specific version of the same package may sometimes be conflicting with other applications' requirements. Hence, it is desired to have side-by-side environments for each purpose to avoid

compatibility issues. This is achieved by setting up a virtual environment.
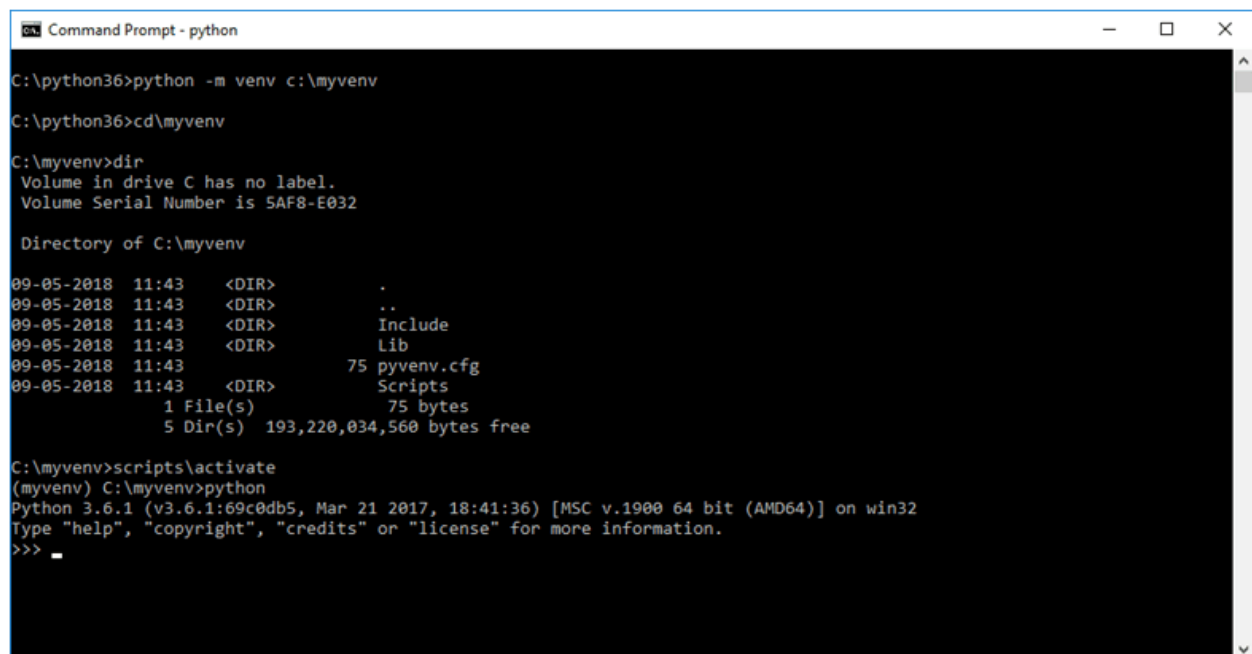
The **venv** module in a standard Python library used to create virtual environments. First run following command

C:\Python36>python -m venv c:\myvenv

This will create **c:\myvenv** and directories inside it containing a copy of the Python interpreter, the standard library, and other supporting files.

To activate the environment run a batch file called activate.bat in the scripts subdirectory. The name of the current environment appears on the left side of the windows command prompt.

Now you can run a local copy of the Python interpreter from the command prompt.

Python Virtual Environment Setup

Run deactivate.bat to terminate the virtual environment

Online Python Shell

Installing Python (or any software) can be a little daunting for a newbie. Fortunately there are many online resources to get familiar with the syntax, features and philosophy of Python before deciding to install Python in the local machine.

You can launch an online Python Shell directly from the official website - https://www.python.org/shell. The Shell terminal shows a Python prompt (>>>) in front of which any valid Python expression can be written, which is executed on pressing 'Enter'.
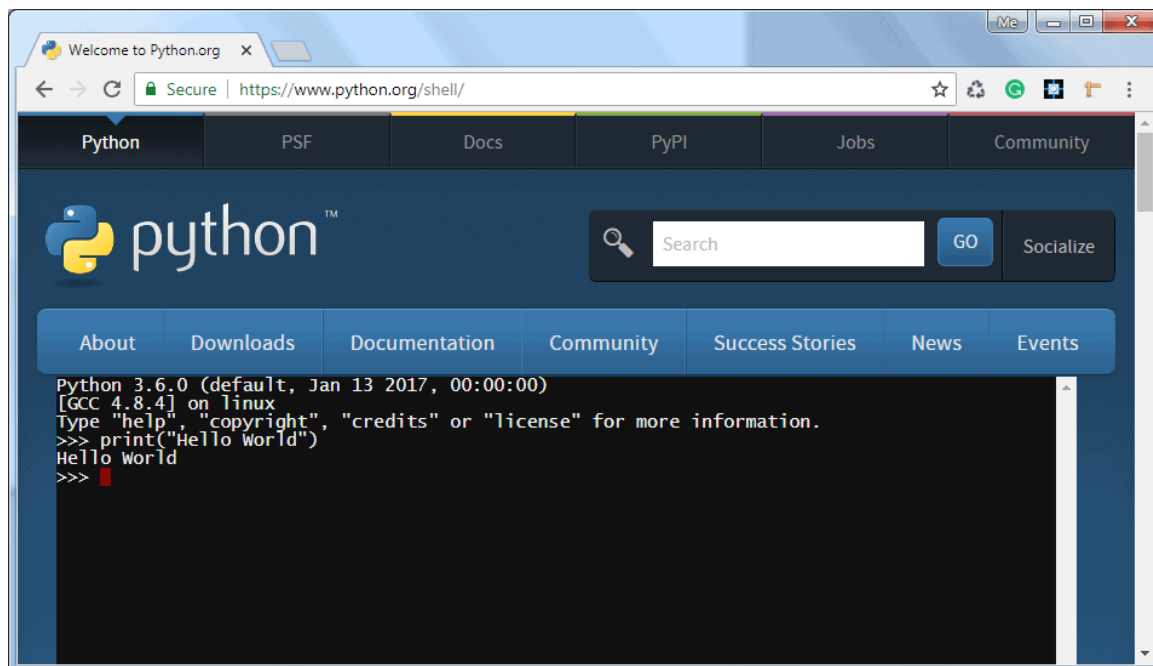
Python Environment Setup

Many interactive Python environment shells can be found on the internet. They work based on **REPL** (Read, Evaluate, Print, Loop). Using https://repl.it it is possible to execute Python in interactive as well as in scripting mode.
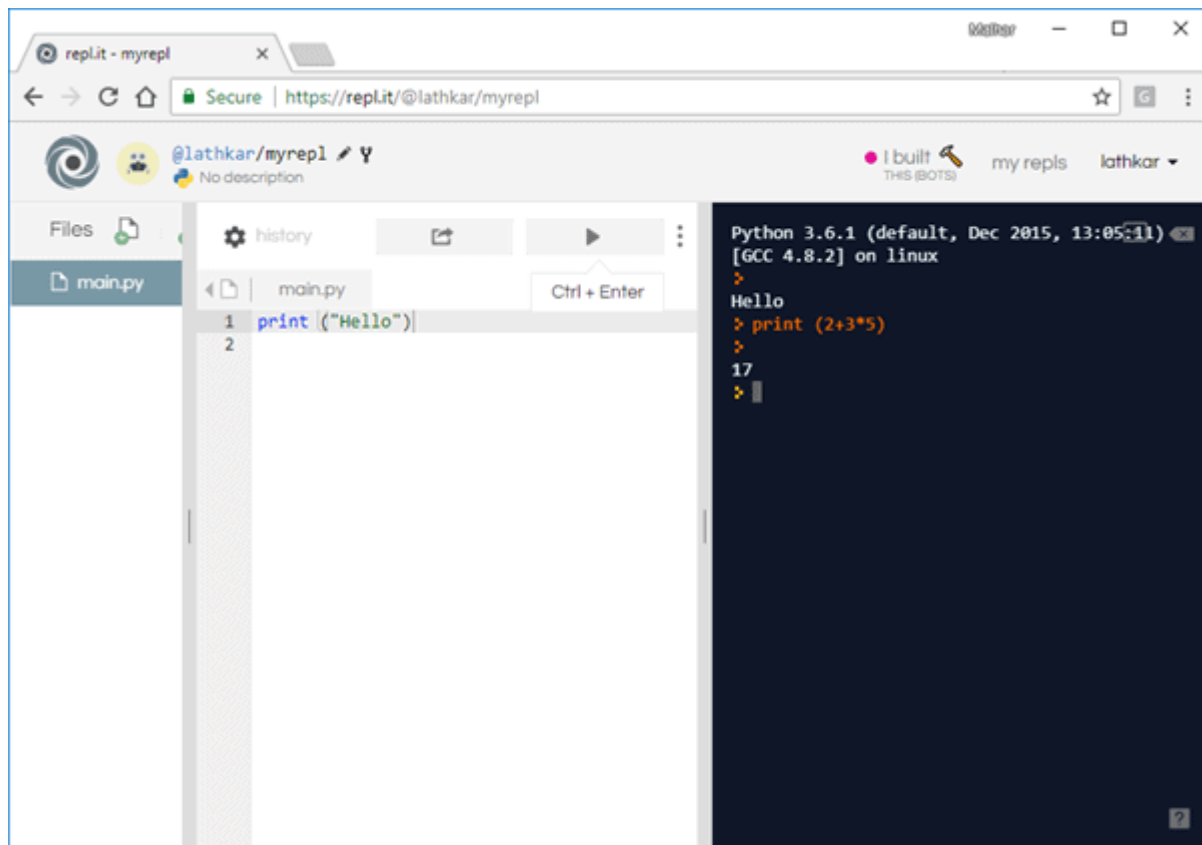
Python - repl.it

The right-hand column in the above diagram is an interactive shell, whereas a Python script can be entered and run in the left pane.