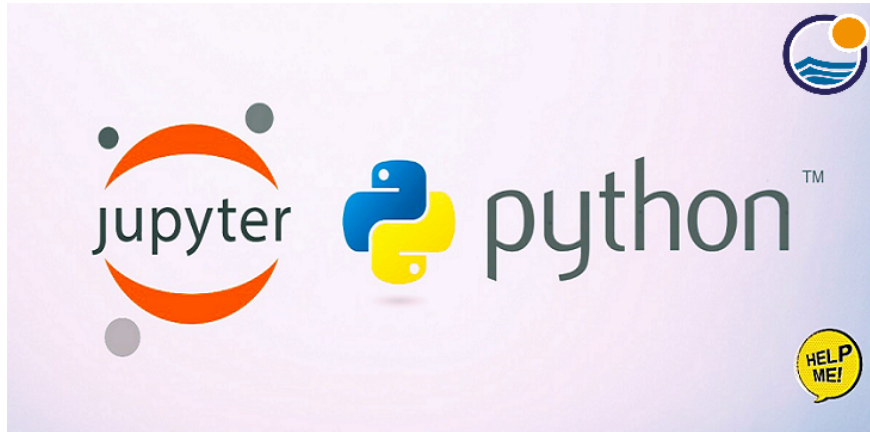


Beginner's Guide for installing Jupyter Notebook using Anaconda Distribution



Random Nerd [Follow](#)

Apr 20, 2018 · 9 min read




Jupyter Notebook (formerly, **IPython Notebook**) is a widely used application in Data Science domain for creating and sharing documents that contain Live code, Equations, Visualizations and Explanatory text. It is a server-client application that allows editing and running notebook documents via any web browser of our choice; and can be executed on a local desktop requiring no internet access or can be installed on a remote server and accessed through the internet. It also has a *Dashboard* (Notebook Dashboard) and a *control panel*, showing local files and allowing to open notebook documents or shutting down their kernels. We may also run Jupyter Notebook without any installation.

Contents

- [1. What is the Jupyter Notebook?](#)
- [2. Installation](#)
- [3. Running the Jupyter Notebook](#)

Jupyter/IPython Notebook Official Quick Start Guide

Setting up Anaconda Distribution:

The weblink associated with above screenshot will direct to Jupyter official webpage where they have exhaustively guided step-by-step for installation. We shall be focusing on getting Jupyter Notebook up and running through Anaconda Distribution from Continuum Analytics. Our foremost agenda (*for beginners and who are very new*) will be to ensure that we uninstall (*How to do help link is at the end of this article*) any other Python IDEs (like PyCharm, Python IDLE, etc.) if we have installed on our computer and ensure that no such related folders are left out. Also ensure that these Python IDEs don't have their PATH added in Environment Variables .

Once we have a clean system, our next step shall be to download Anaconda (being a beginner, we shall avoid Miniconda) distribution. For this process, we need to be aware of 2 critical information about the **Operating System on our computer**: > *Type* (Windows/Macintosh/Linux), and > *Architecture* (32/64 bit). Another important thing to note is the **variety of available packages**: > *Anaconda 2* (Download this only if you intending to use *Python ≤ 2.7*), and > *Anaconda for Python 3* (Recommended | Allows creating virtual environments to use Python 2, if required).

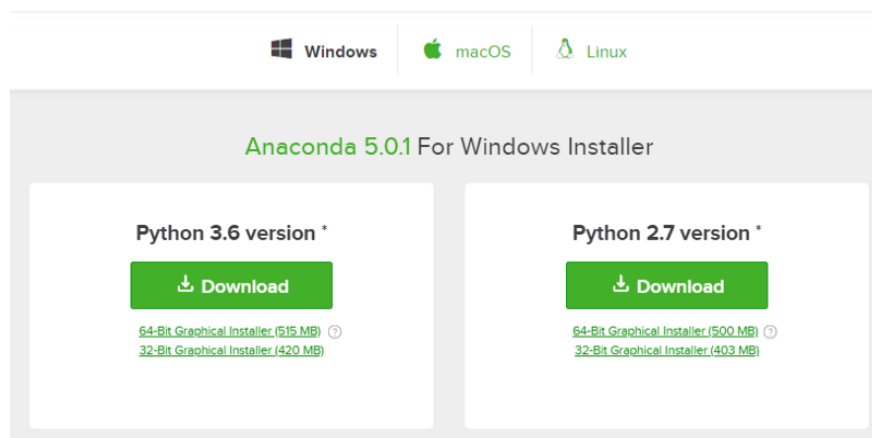


Image 1 | Anaconda for Python 3 Download Link [USE ME]

As we may notice in *Image 1* (above screenshot), this is a large file (500+ MB) so it will *take sometime to download*, depending upon our internet connection speed. Please note if we're trying to download it at our workplace & our corporate security settings do not allow (*Admin/User permission errors*) downloading an executable file, then alternatively we may download their zipped file which contains all the executable files within. Once downloaded we may proceed towards installation (recommended to right click and *initiate as an Administrator*), keeping in mind that if we get any **prompts** like:

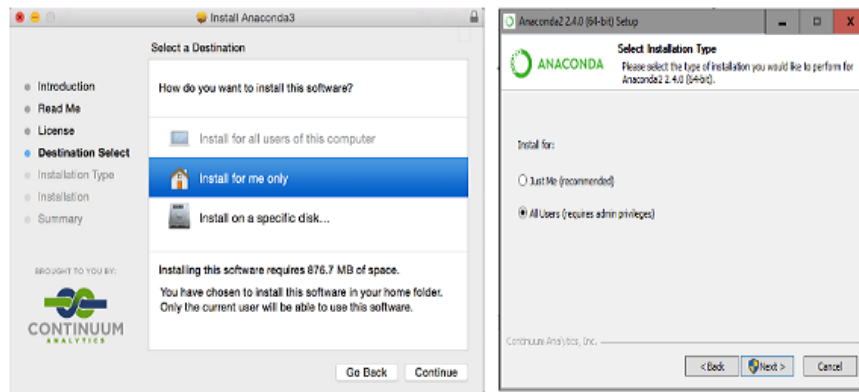


Image 2 | Mac + Windows | Select 'ALL USERS'

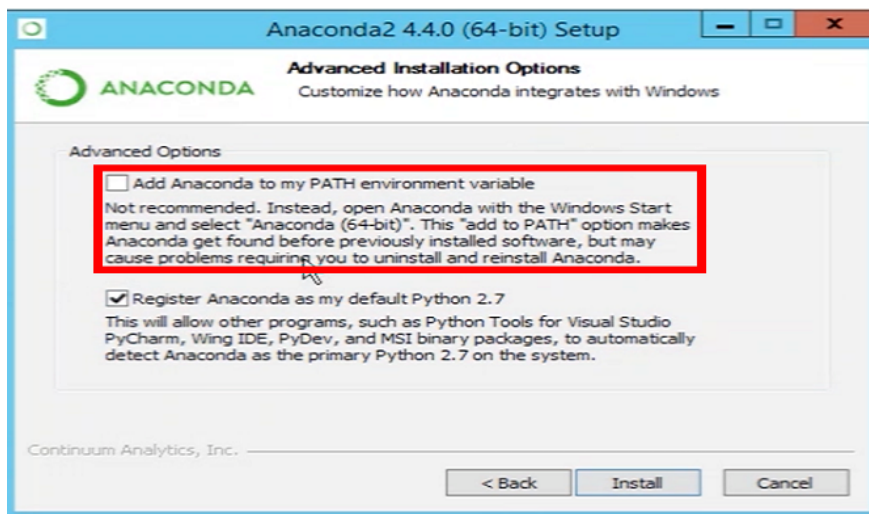


Image 3 | Select both the options to 'Add Anaconda to PATH Variable', although Anaconda recommends otherwise.

In *Image 3*, during installation we ensure that the folder/directory where Anaconda gets installed (*in layman terms*), has that **path added to our computer/system**. Also, please ensure that the *Directory/Folder name* has **no spaces** in it (*not* something like “ C:\Users\My computer\”). Now, when we execute our commands (e.g.- `conda install numpy`), our Windows Command Prompt (on PC) or Terminal (on Mac/Linux) recognizes the actual path of the Anaconda directory and accordingly installs the module (*Numpy* in our example). Later on this will fortify proper execution of commands like `import numpy as np` in our Jupyter Notebook (or, even in *Spyder*) that we would always require in our Python script/code. Here is [Anaconda Quick Start Guide](#) (for installation/running help) & [Anaconda Cheat Sheet](#) (List of navigating commands) in PDF format for anytime assistance. There is a list of frequent [Q&A in their official documentation](#). As a quick note, once we accept the default option to install Anaconda on the default path (as suggested), it gets installed in our user home directory:

- Windows 10: `C:\Users\<your-username>\Anaconda3\`
- MacOS: `/Users/<your-username>/anaconda3`
- Linux: `/home/<your-username>/anaconda3`

If we're unable to find Anaconda Navigator in our Search bar, visit the installation folder and head to `Scripts`. If we don't find `anaconda-navigator.exe` file out there, this suggests that we haven't properly installed Anaconda and hence it is missing.

. . .

Launching Jupyter Notebook:

We're done with our successful Anaconda installation, that shall now give us easy access to few amazing tools related to Python development. An effortless way to access all these installed tools is by using **Anaconda Navigator**, which is a desktop graphical user interface (GUI) included in Anaconda® distribution that allows us to launch applications and easily manage *conda* packages, environments and channels without using command-line interface.

To be on a safer side, we shall restart our computer at this stage. Now, use the search bar in your computer to launch this console and note that it takes some time to initiate for the first time, so be patient with it. A simple **click on the Jupyter Notebook icon** shall open up our notebook in the default web browser (like *Google Chrome, Mozilla Firefox, Safari* and *Windows IE/Edge*) of our computer. But, if we're trying to launch Jupyter Notebook for an environment within Anaconda Navigator, then please use drop-down menu next to **Applications on**, and select that particular environment before launching Notebook.

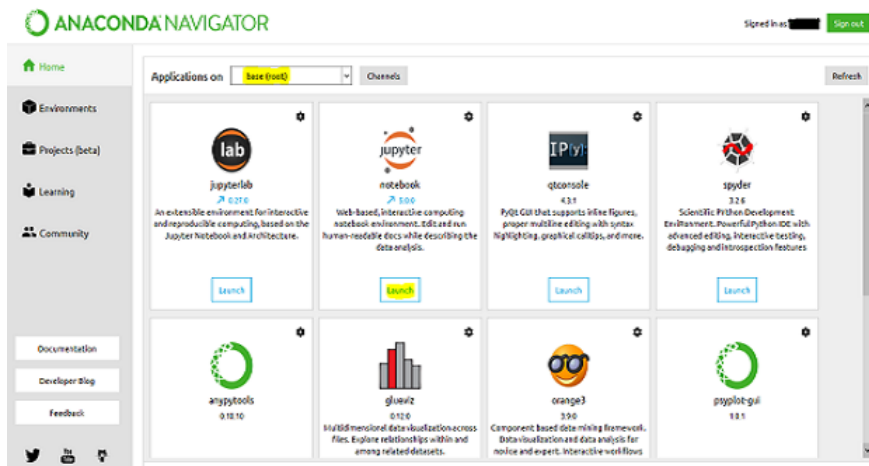


Image 4 | Anaconda Navigator GUI for (highlighted) Jupyter Notebook 'Launch' button

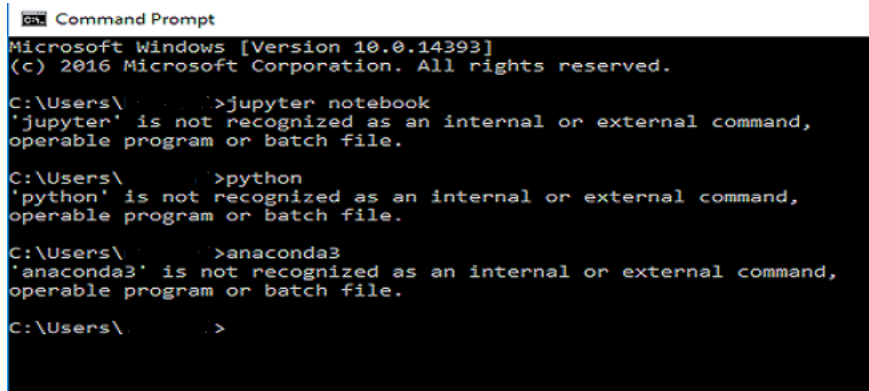
Lets us now focus on **launching Jupyter Notebook via Command line**, and actually that as well involves just one step, i.e. type and run `jupyter notebook` within your Windows Command Prompt or Terminal. This will print some information about the notebook server in our terminal, including URL of the web application (by default, <http://localhost:8888>). Before we run this one line command, it is recommended to double check we have our Anaconda directory PATH properly added to Environment Variables, and if not, we need to locate our Anaconda directory/file path and add it to Environment Variables, or else it might fetch an Error: **“jupyter is not recognized as an internal or external command”** (Image 6). Also note that if you're running this one-line command from **Anaconda Command Prompt** (*Powershell* backed), you might notice two different file paths (as in Image 5), and that isn't a problem. If the launch is successful, the command prompt shall look like Image 5 and **Jupyter notebook will open up** (as in Image 7).

```

Administrator: C:\WINDOWS\system32\cmd.exe - jupyter notebook
(C:\Users\...\Anaconda3) C:\Users\...\Documents>jupyter notebook
[I 15:55:10.189 NotebookApp] JupyterLab alpha preview extension loaded from C:\Users\...\Anaconda3\lib\site-packages\jupyterlab
JupyterLab v0.27.0
Known labextensions:
[I 15:55:10.267 NotebookApp] Running the core application with no additional extensions or settings
[I 15:55:10.438 NotebookApp] Serving notebooks from local directory: C:\Users\alok\Documents
[I 15:55:10.438 NotebookApp] 0 active kernels
[I 15:55:10.438 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/?token=120440b5389f19baf82f5547f62084f10caF331ed012d4f7
[I 15:55:10.438 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).

Copy/paste this URL into your browser when you connect for the first time,
to login with a token:
http://localhost:8888/?token=120440b5389f19baf82f5547f62084f10caF331ed012d4f7
[I 15:55:11.454 NotebookApp] Accepting one-time-token-authenticated connection from ::1
  
```

Image 5 | Anaconda Prompt to Launch Jupyter Notebook



```

C:\Users\...>jupyter notebook
'jupyter' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\...>python
'python' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\...>anaconda3
'anaconda3' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\...>

```

Image 6 | Win Command Prompt | Error: 'Jupyter' and 'Python' not recognized.

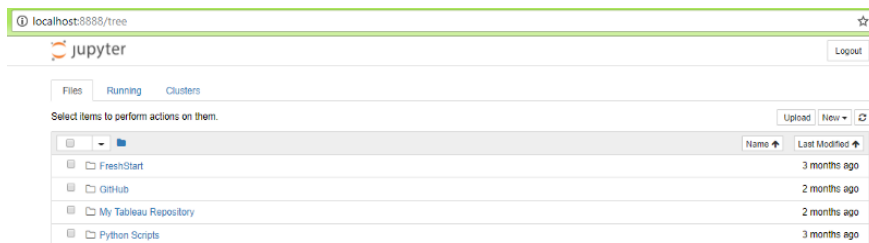
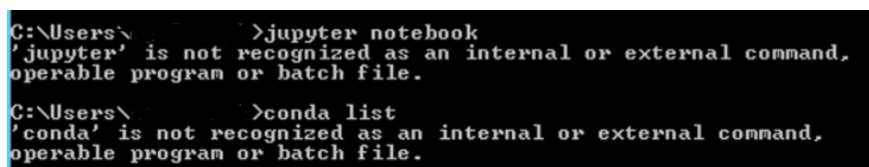


Image 7 | Jupyter Notebook Dashboard on default web browser

There might be a situation where we notice in our Command prompt or Terminal that the File Path (`C:\Users\myusername\Anaconda3`) isn't correct, so in that scenario we will have to navigate to the correct path [on Windows](#) or [on Macintosh](#). Later on, if we find an issue while downloading/installing libraries (`conda install numpy`) within Anaconda where it outputs an **Error: conda-is-not-recognized-as-internal-or-external-command** (as in *Image 8*), again we shall follow the same process of altering the directory path.

If we want to know where is Jupyter installed on our computer, we may run `where jupyter` in our Command prompt. If we wish to know which Python version do we have installed, run `python` or `python -V` or `python --version`. Sometimes, when trying to launch Notebook from Anaconda Navigator, it might fetch a blank page on our web browser: This indicates that system isn't able to read the `jupyter-notebook.exe` file that gets generated by Anaconda & in order to solve it, we may try to directly paste `localhost:8888` in our browser as URL & hit enter.



```

C:\Users\...>jupyter notebook
'jupyter' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\...>conda list
'conda' is not recognized as an internal or external command,
operable program or batch file.

```

Image 8 | Error: conda not recognized

. . .

How to navigate within Jupyter Notebook:

Jupyter Notebook has a very user-friendly Editor with a bundle of easy-to-use navigation/execution buttons right on top (*Image 9*). These notebook documents are stored in the JSON plain text format, which makes it easier for them to be shared with colleagues and controlling versions. They're also available from a publicly accessible URL, and can be **shared using Jupyter Notebook Viewer** with others without requiring others to install Jupyter Notebook on their machines. These notebooks have a `.ipynb` extension (can also be accessed via Enthought Canopy or IPython). Here is a complete Notebook User's manual for quick reference (*bookmark it*)

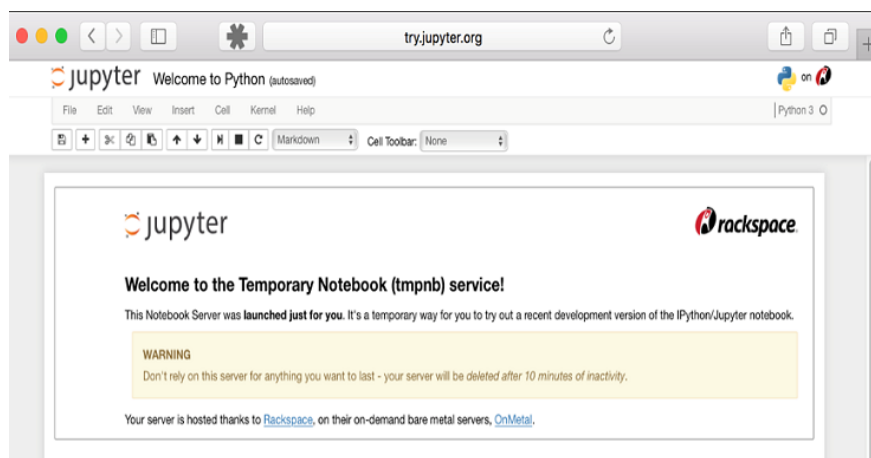


Image 9 | Notebook Editor

Alternatively, we may also use keyboard shortcuts to access files and execute our Python scripts. Here is a Jupyter Notebook Keyboard Shortcut Cheat sheet to refer to in case we forget them. Let us now talk about **handling our Dashboard** which automatically detects all the `.ipynb` files in our directory but in case if our required files are in a separate directory, it also allows us to upload (has an `UPLOAD` button in the Dashboard) a file (Detailed view in *Image 10*). We will also observe a `NEW` button which when clicked, allows us to select a Python kernel (options depend upon what's installed on our local server) from the dropdown menu (*Image 11*). Some kernels in the screenshot below (*Image 11*) may not exist as an option in our case (**Now we know why 'Python 2' or 'R' isn't visible with our installation**).

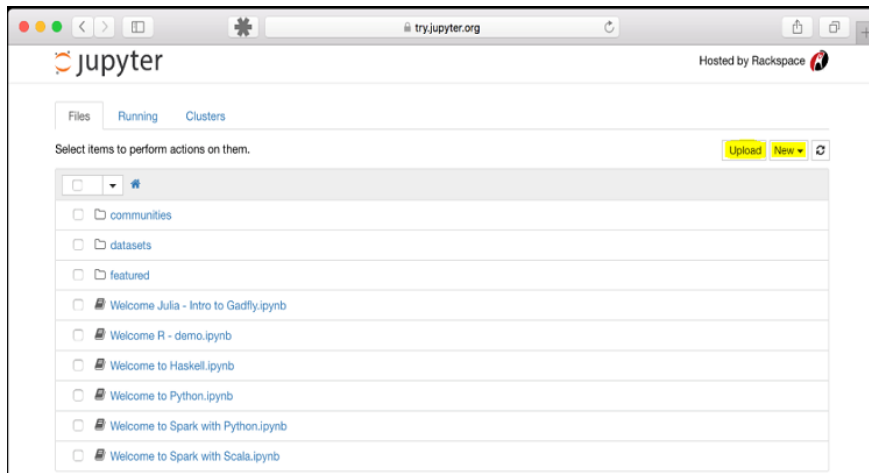


Image 10 | Detailed view of Jupyter Dashboard

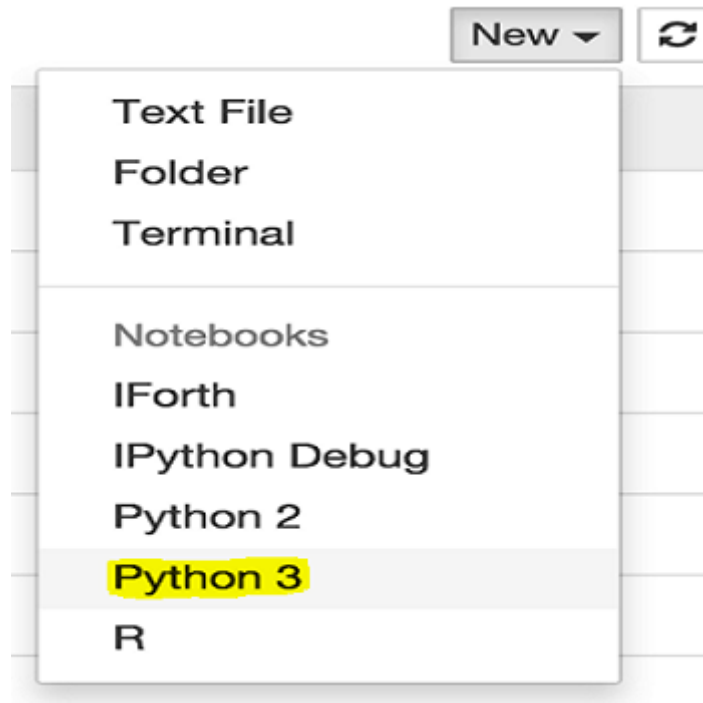


Image 11 | Jupyter 'New' Kernel

Notebooks and files can be uploaded to the current directory by dragging a notebook file onto the notebook list (*Image 10*) or by using 'click here' text above the list. The **notebook list** shows green *Running* text and a green notebook icon next to running notebooks; and remain running until we explicitly shut them down so closing the notebook's page is not sufficient. To **shutdown, delete, duplicate, or rename a notebook**, check/tick the checkbox next to it and an array of controls will appear at the top of the notebook list (as in *Image 12*). We may also use the similar operations on directories and files when applicable. Lastly, to view all of our running notebooks along with their directories, click on the *Running* tab.



Image 12 | Notebook File/Directory Operations

. . .

Common issues with Jupyter Notebook:

Jupyter Notebook team is a bunch of dedicated developers who consistently keep resolving their Open issues. If we wish to take a **look at open issues or report a new issue**, we may visit their [GitHub](#). As a beginner, it is our responsibility to check all the possible online resources like [StackOverflow](#) (currently 6000+ questions have been already answered) for existing solutions before raising a new one. Below listed are **few common/frequently asked Jupyter Notebook issues and their solution**:

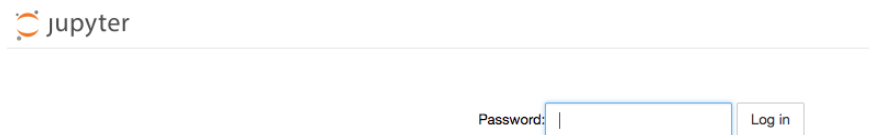



Image 13 | Error: Jupyter Notebook asking for a Password or Token

- > [Launch IPython notebook with selected browser](#)
- > [Change IPython/Jupyter notebook working directory](#)
- > [How to change the Jupyter start-up folder](#)
- > [Jupyter Notebook suddenly started asking for Password or Token](#)
- > [How to disable password request for a Jupyter notebook session?](#)
- > [IOPub data rate exceeded when viewing image in Jupyter notebook](#)
- > [How can I check if code is executed in the IPython notebook?](#)
- > [Link Conda environment with Jupyter Notebook](#)

- > [Error when executing `jupyter notebook` \(No such file or directory\)](#)
- > [Conda environments not showing up in Jupyter Notebook](#)
- > [Jupyter notebook not trusted](#)
- > How to uninstall Python IDLE from [Mac](#) and [Windows](#) ?
- Anaconda uninstallation for [Windows](#) and [Mac](#). [PyCharm removal on Win and Mac](#).
- > [How to make Jupyter Notebook Matplotlib plot inline?](#)
- > [How do I add Python 3 kernel to Jupyter?](#)
- > [Jupyter Notebook timeout waiting for response in Chrome](#)
- > [How do I increase cell width of Jupyter Notebook in my browser?](#)
- > [Running jupyter via command line on Windows](#)
- > [IPython notebook won't read the configuration file](#)
- > ["ImportError: No module named" when trying to run Python script](#)

Number of issues are unlimited and keep varying (like any other application) so enclosing all of them in this brief article isn't possible. If your issue wasn't among listed few, I shall again reiterate the solution process: **Search in [StackOverflow](#) -> [CheckGitHub](#) for Open/Closed issues -> [Check Community Forums](#) -> [Optional is Reddit/Quora](#) -> **Check with your mentor**. For people new to *Medium* platform as well, words or sentences that are underlined in this article are relevant hyperlinks that I have attached for you to refer to. I shall keep updating this list, as and when I get time. Meanwhile, please use the Comments section if you have any doubts or feedback. Also, there is an easy alternative that you might want to look at, [Python Anywhere](#) (if interested). Thank You for your time!**

