

# How to Install Modules for Python 3.X Using Pip in Windows 10

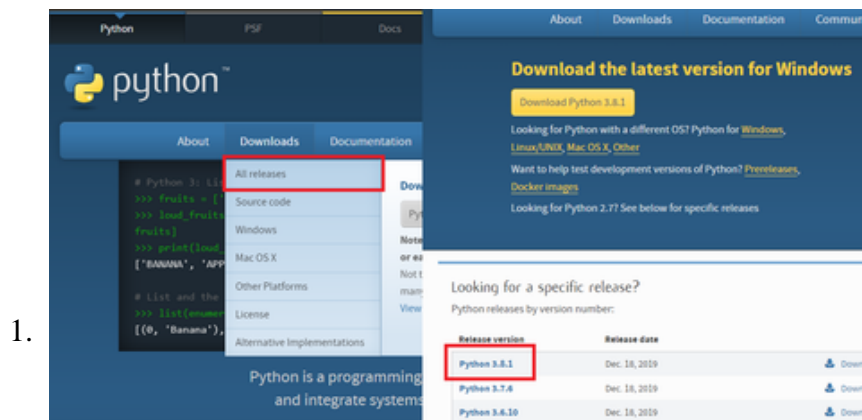
If you're a budding computer scientist working with Python 3 and want to add functionality and power to your projects that doesn't exist in the base built-in Python modules, you may want to try to install external third-party modules and...

Part 1 of 3:

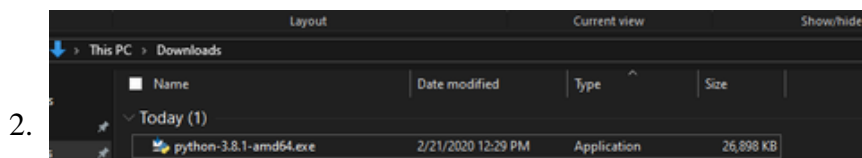
## Getting pip On Your System

1. Depending on whether Python is already on your computer or not, you may have to take different steps. Use Method 1 or Method 2 depending on your circumstance.

### If Python is Not Installed



Go to [python.org](https://python.org) to select the version of Python you wish to download. Typically, the latest version is preferable as it is less bug-prone and has more features, but nearly all the Python 3.X installations should work just fine. Remember to get the 32-bit or 64-bit depending on your system.



Run the installer once it's finished downloading. This is to initiate the installation process. For the most part this is self-explanatory, but there's a crucial check to make.

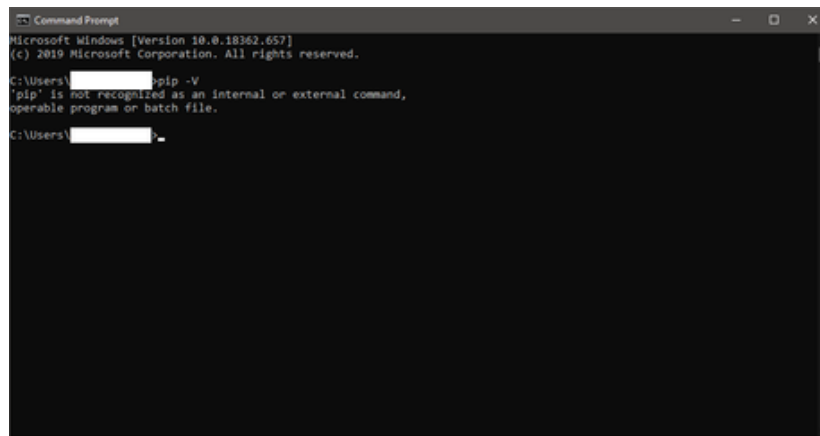
3.



**Check off 'Add Python 3.x to PATH'.** This will allow you to use Python and pip from a command shell easily, which is important for installing packages.

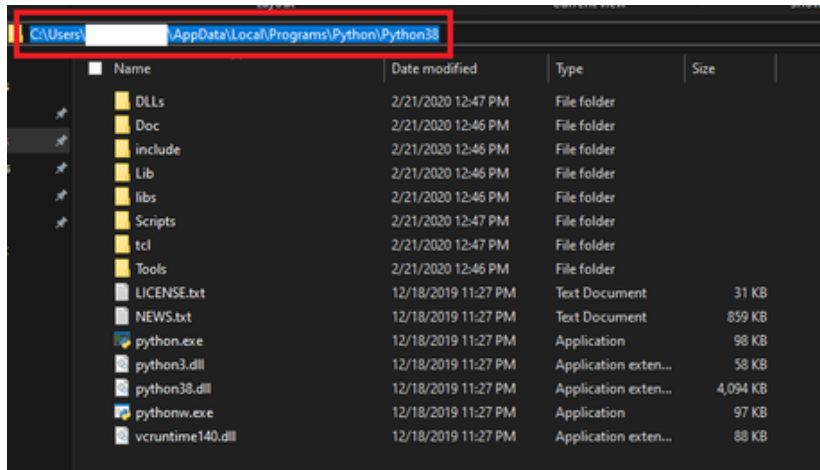
### If Python is Already Installed

1.



**Test to see if pip can be used from the command line already by typing `pip -v` into a command shell.** If this command works and version details for your pip installation are shown, you can skip to Part 2 of this guide. Continue below if the command fails.

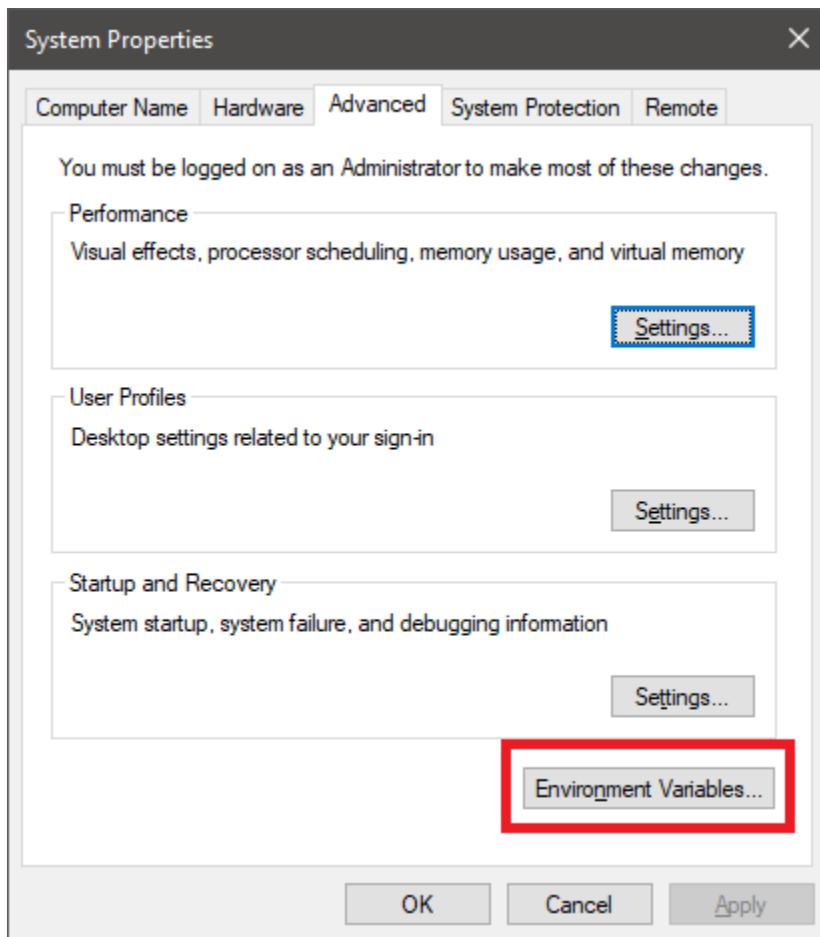
2.



**Locate the directory in which you installed Python originally.** Remember this path; that's the folder where the executables are for running Python and pip. This may be difficult to find, but typical locations that you may find Python installed at are:

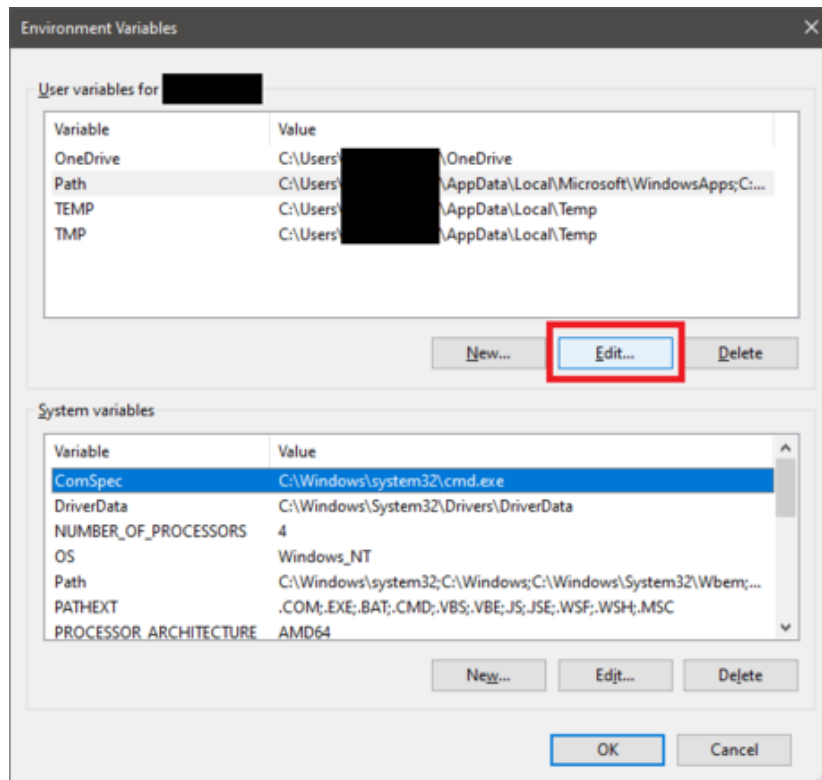
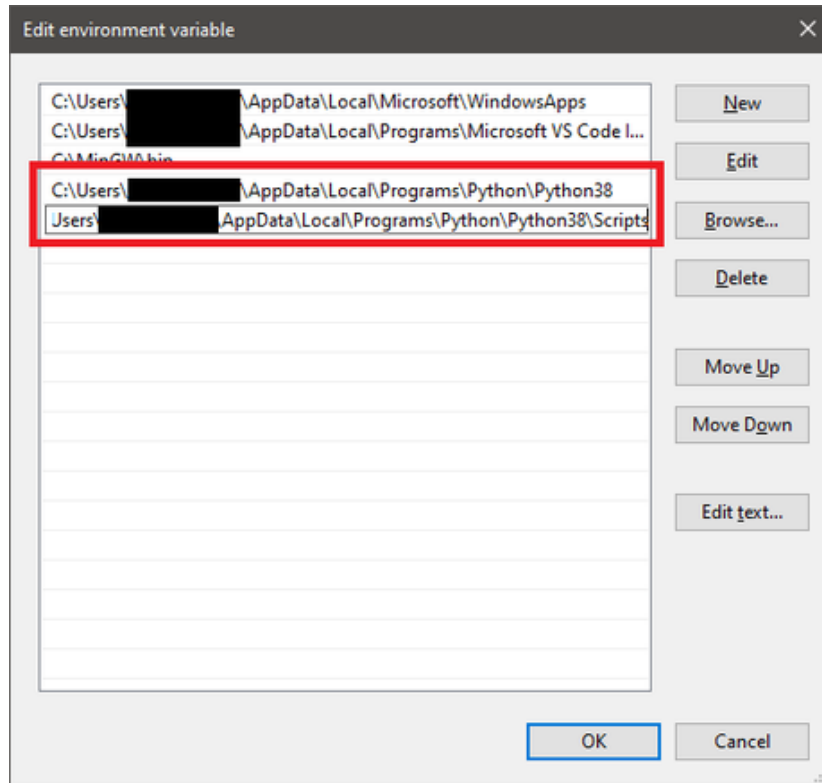
1. *C:\PythonXX*
2. *C:\Users\<username>\AppData\Local\Programs\Python\PythonXX*

3.



**Open the Environment Variables dialog.** This can be done by searching up "system variables" and clicking on the first option, then clicking on the *Environment Variables...* button.

4.



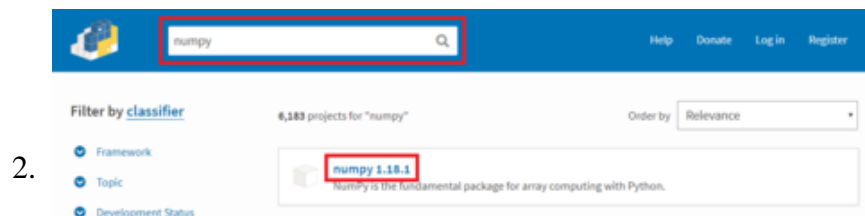
**Add the Python path you just copied to the user 'Path' variable.** You can do this by selecting the 'Path' variable, clicking `Edit...` ? `New` and pasting the path. Paste it again in a new field with "Scripts" at the end of it. *You may need to restart your computer after this step.*

1. After following these steps, you will have a local Python installation with full pip functionality that you can use from a command shell.

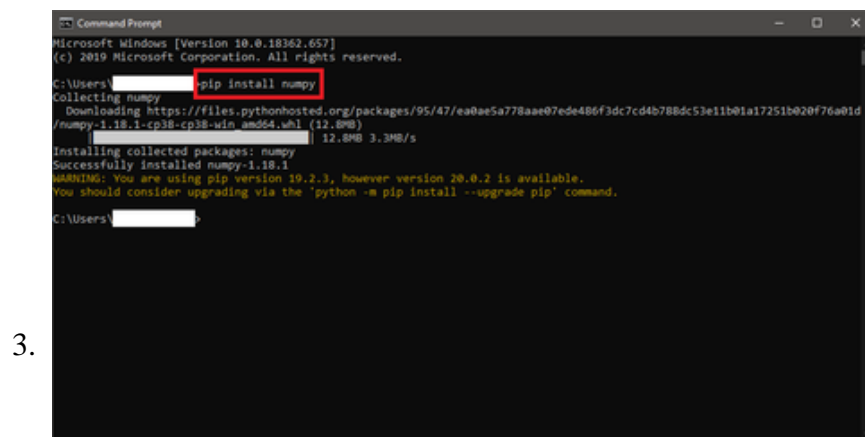
Part 2 of 3:

## Finding and Installing Modules for Python

1. For this example, the module installed will be *numpy* — an extremely rich mathematical library with robust functionality that many other libraries depend on. You can install whatever you'd like however.
1. **Research what kind of modules you want to install on your computer.** There are many, many modules out there that all do great things. In order to deal with the problems you want your program to solve, try looking up modules online by searching "*how to python*"; more often than not, websites can be full of recommendations and helpful advice.



**Look up the package name for the module you want to install.** Go to [pypi.org](https://pypi.org) and look up the module you want. The name of the package that contains it as well as the command needed to install it are at the top of the page. It will look something like `pip install` .



**Open the command shell on your computer and run the command from the PyPI page.** This will initiate the installation process. *Remember to close all instances of Python that are running when you do this.*

After completing this part, the module will be installed and ready for use in your Python projects.

Part 3 of 3:

## Using Newly Installed Modules in Python Code

1. After getting your hands on a new Python module, the next step may seem obvious — use the module! — but it may be likely that you might not know exactly how the module is imported, initiated or otherwise. Here are some steps to help you get started.

```
Python 3.8 (64-bit)
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 23:11:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import numpy
>>> numpy.version.version
'1.18.1'
>>>
```

1.

**Open a fresh Python instance (IDLE or shell) and type in the import statement for your module.**

More often than not, the module name for importing is the same as the package name. You can always use the documentation to verify this. Once you type in the line of code to set up your import, you're good to go. Add some other code as you need to.

2. **Execute your code in your editing environment.** If no errors occur, congratulations! You've managed to install a brand-new third-party Python module.
  1. With this, your foray into Python module installation and usage is complete!

You finished reading the article "**How to Install Modules for Python 3.X Using Pip in Windows 10**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.