# How to add Python to Windows PATH

There are few ways in which you can add Python to Windows PATH. In this guide, I'll share with you two methods that you can use to add Python to Windows path:

- Via the installation of a recent version of Python
- Manual entry of the paths

But why would you want to add Python to Windows path in the first place?

Well, if you try to install a Python package using PIP for example, you may get the following error in the Windows Command Prompt:

***'pip' is not recognized as an internal or external command, operable program or batch file***

To overcome this error, you may apply any of the two methods described below.

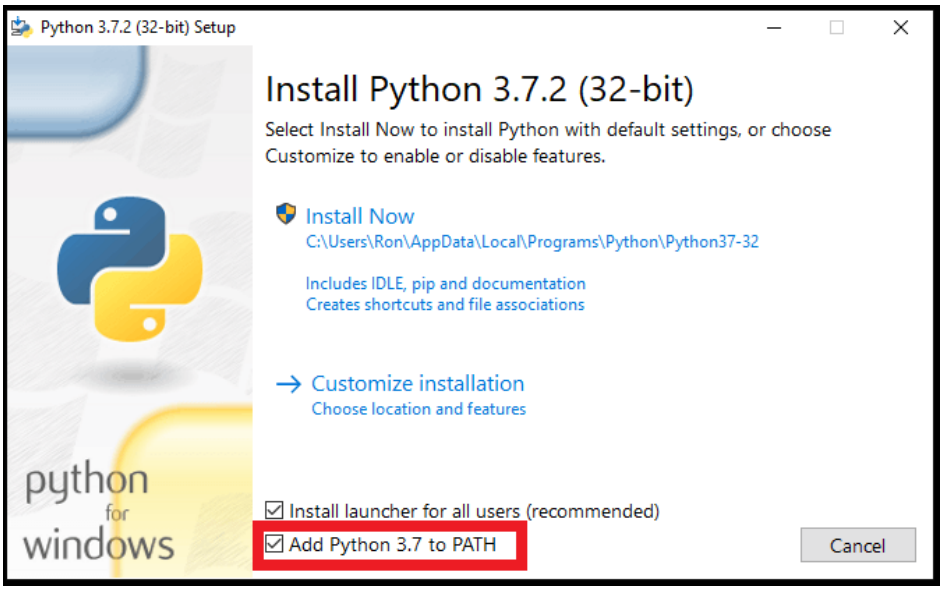
## Method 1: Install a recent version of Python

You can easily add Python to Windows path by downloading a recent version of Python, and then checking the box to **Add Python to PATH** during the installation.

Before you proceed, you may choose to uninstall your previous version of Python if needed.

In my case, the latest version of Python that was available to download was version 3.7.2.

In the Python installation box, just check the box to add Python to PATH as below:




Finish the installation, and you should be good to go.

Alternatively, you may manually add the paths into the Environment variables.

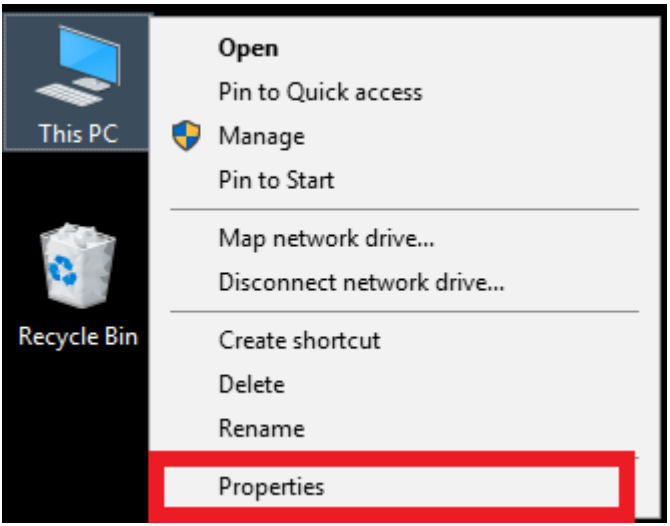
## Method 2: Manually add Python to Windows Path

If you wish to stick with your previous version of Python, you may apply the steps below to manually add Python to Windows path.

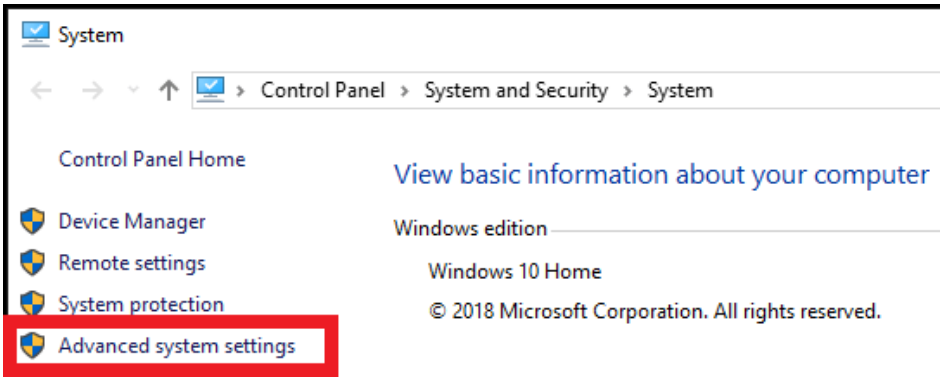
Note that I'll be using Windows 10 to demonstrate the steps, but similar principles would apply for previous versions of Windows.

## Step 1: Get to the Windows Environment Variables screen

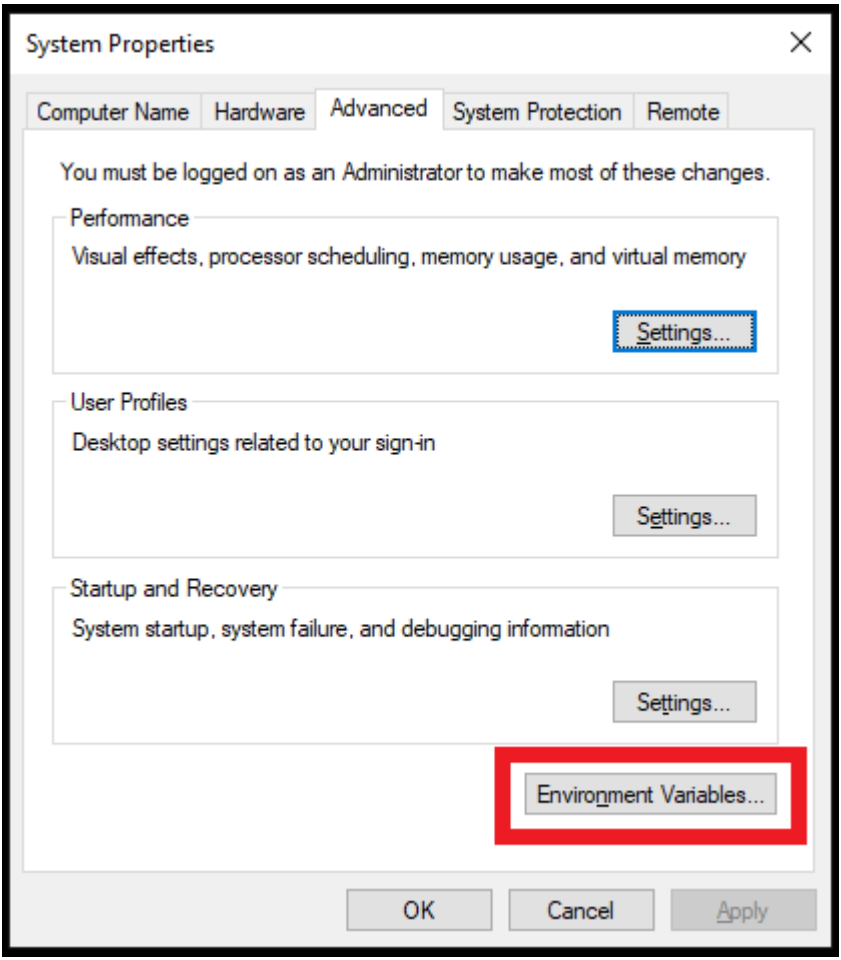
To get to the Windows Environment Variables screen, where you can add new paths, simply right click on the '**This PC**' icon. Then, select '**Properties**.'




Next, click on the '**Advanced system settings**'

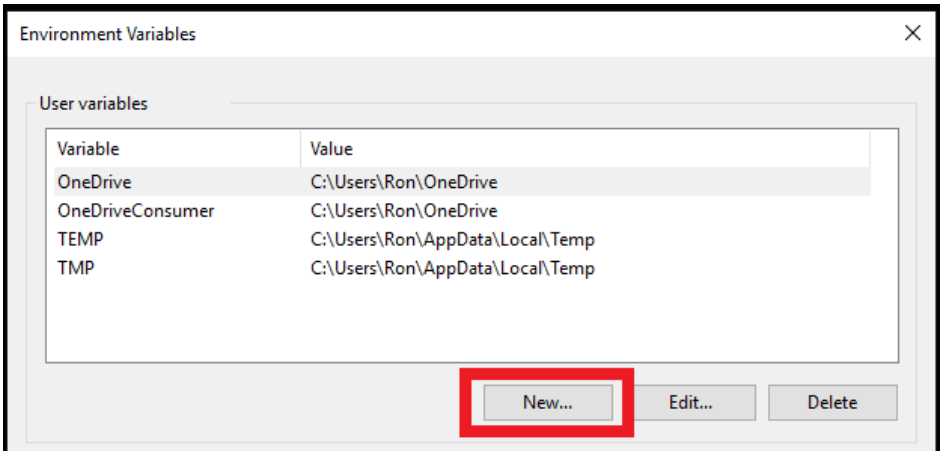



Finally, click on the ‘**Environment variables…**‘

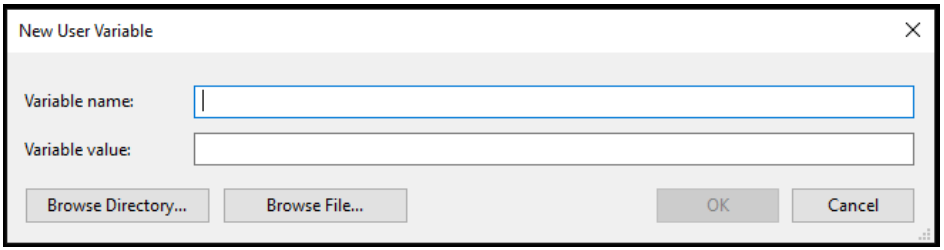



That should take you to the Environment Variables screen, where you can add new paths.

Click on '**New…**' which is located under the *User variables*box.



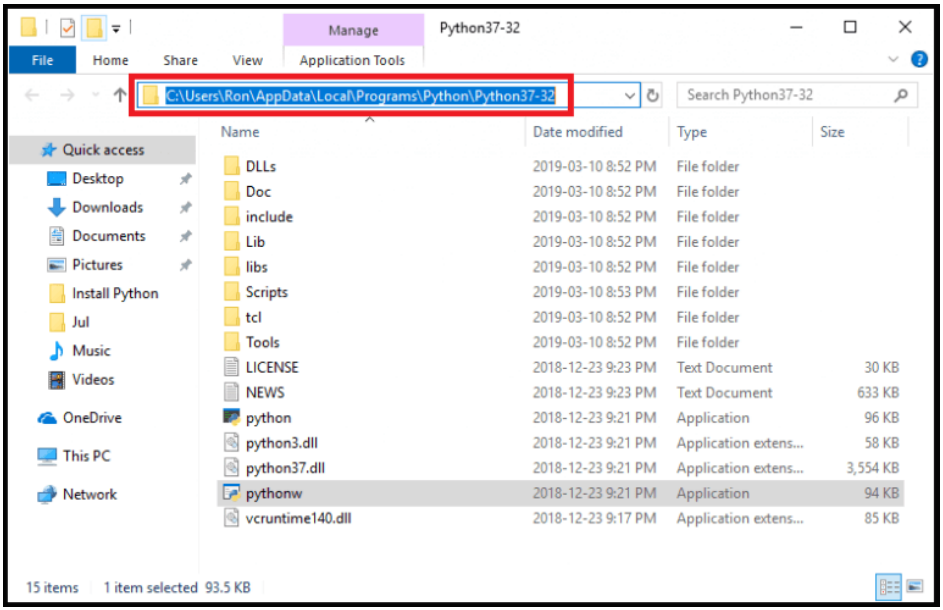

You should then see the following box, which will allow you to add new variables.



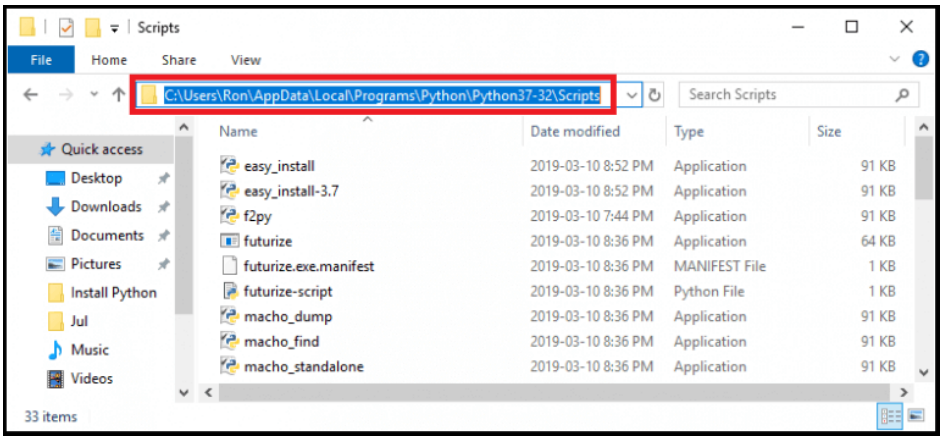

Before you type any values, you'll need to locate the relevant Python paths. The paths that you'll need to get are:

- The Python application path, which is the folder where you originally installed Python; and
- The Python Scripts path. The Scripts folder should be located within the Python application path.

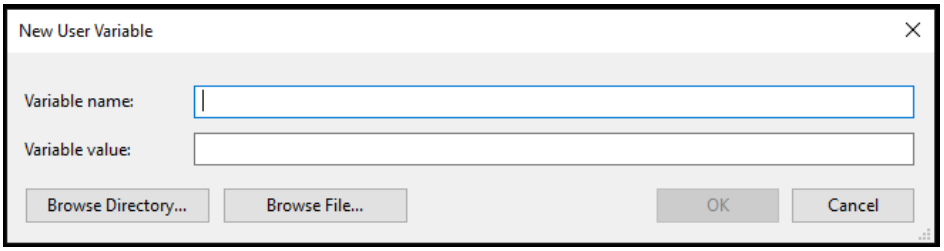
Here is how my Python application path looks like:




And this is how my Python Scripts path looks like:




Now let's fill the **New User Variable** box that you saw earlier:



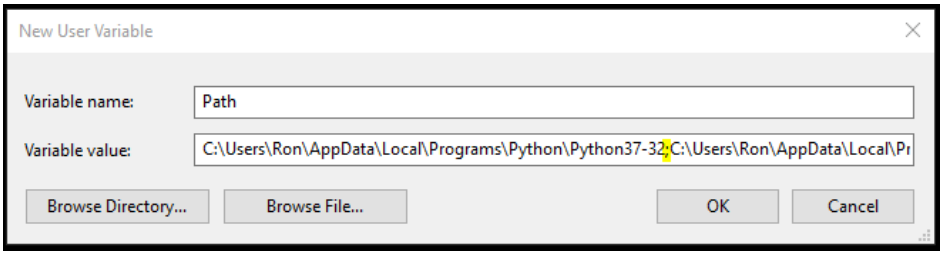

For the **Variable name**, you may type a name, such as ‘Path’ for example.

For the **Variable value**, copy the full Python application path, then use *semicolon* (as highlighted in yellow below), and finally copy the Python Scripts path.

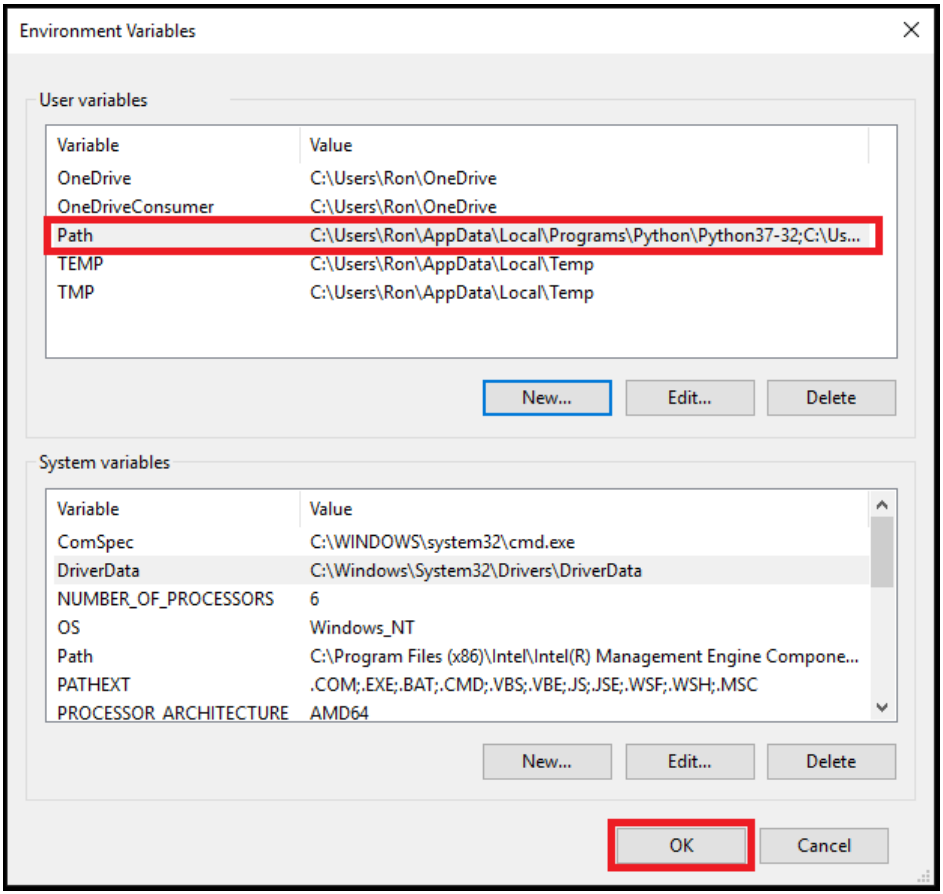
This is how my Variable value would look like:

C:\Users\Ron\AppData\Local\Programs\Python\Python37-32;C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts

Putting all the values together in the New User Variable box:




Press ‘OK’ and you would then see your new Python Path under the ‘User variables’ section. Don’t forget to press ‘OK’ again so that the changes will get implemented.




That's it! You just added Python to the Windows Path.

You'll now be able to install Python packages easily, by opening the Windows Command Prompt and then typing:

**pip install package name**

For example, to install the pandas package, simply type 'pip install pandas' and then press Enter:


Similarly, you may upgrade PIP by typing the following command:

```
python -m pip install --upgrade pip
```