# Background and Scope

Manitoba Agriculture Weather Program operates and maintains a network of 108 weather stations that provides weather variables such as air temperature, wind speed and direction, humidity, precipitation, solar radiation and soil temperature and moisture at 5, 20, 50 and 100 cm depths.

The program supports many partners and the data has a wide variety of applications such as generating current condition maps, crop thermos-physiological growth stage (e.g. Growing Degree Days, Corn Heat Units etc.) and crop disease risk mapping (e.g. Fusarium Head Blight). This job aid shows how to calculate Disease Severity Value (DSV) for Potato Blight using the AgAuto program.

# Software Installation

In order to use the AgAuto program, the necessary software must first be installed. The software that AgAuto needs are Anaconda, and PyCharm. AgAuto also needs the folder containing the AgAuto script and configuration files.

**Installing Anaconda**

Anaconda is an open-source code python software distribution and is how the necessary package(s) will be installed.

1. First navigate to <https://www.anaconda.com/distribution/>
2. As shown in Figure 1, select the windows icon to select Anaconda for windows.

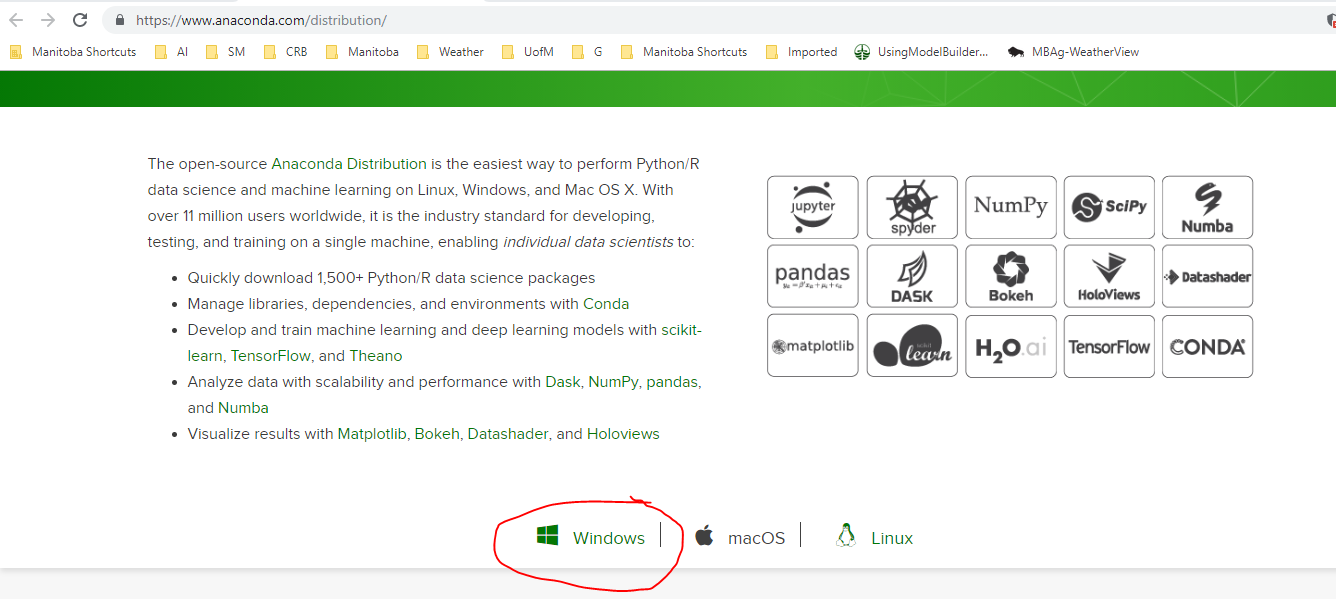


Figure 1: Selecting proper Anaconda version for computer.

1. As shown in Figure 2, select the latest python version (Python 3.7 as of May 2019) for the 64-bit version of windows.

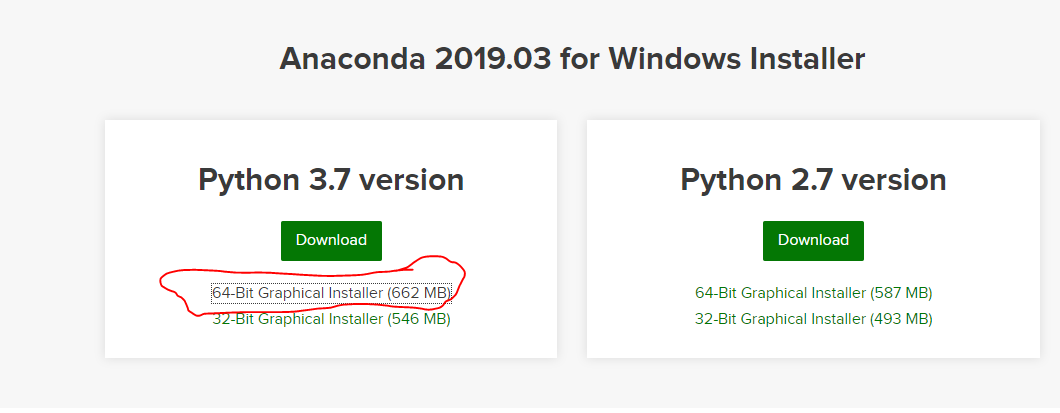


Figure 2: Selecting the latest 64-bit Anaconda version.

1. After the download has completed, go through the installation process. The installation could take up to 30 minutes.
   1. If prompted to select an installation folder, just use the default location.
   2. If prompted, do not check the box ‘add to path variable’.

**Setting Up the Virtual Environment**

Although we have previously installed Anaconda for Python 3.7, the AgAuto program is written on Python 2.7. The newest Anaconda was installed in order to easily install new packages in the future. A future project might be to port the AgAuto script into Python 2.7 but for now, we can create a virtual environment for Python 2.7 to run Python 2.7 code.

1. In Windows search, type ‘Anaconda Prompt’ and run as shown in Figure 3.

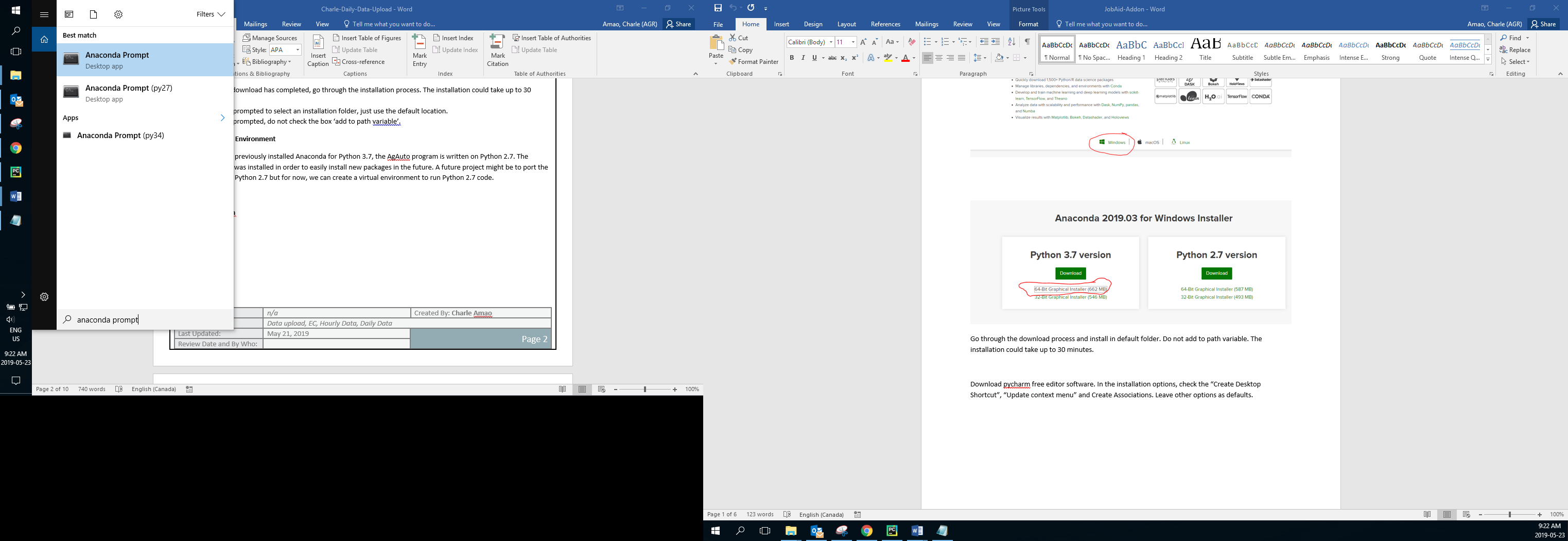


Figure 3: Running the Anaconda prompt.

1. At the prompt, type *‘conda create -n py27 python=2.7 anaconda’* as shown in Figure 4. This process might take 30-40 minutes.
   1. If prompted to install packages (y/n), type ‘y’ to install the necessary packages.

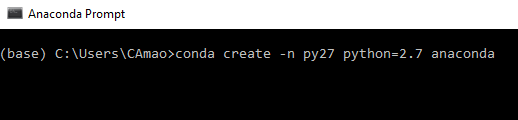


Figure 4: Typing this command creates a virtual Python 2.7 environment named 'py27'. We will use this to run AgAuto in later steps.

1. Once the creation process is complete, type ‘conda activate py27’ and press Enter in order to activate the newly created virtual environment.
2. While inside the prompt, navigate to the AgAuto’s working directory. This is simply inside the folder that contains the AgAuto script and its configuration files. In order to navigate to the directory simply type *cd filepath*, where *filepath* in this case is *C:\Users\CAmao\Documents\Project 1\AgAuto, filepath* will change depending on where you place the AgAuto directory.

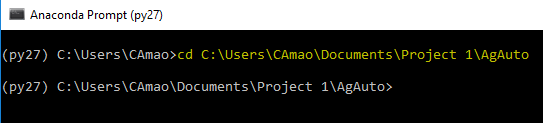


Figure 5: After entering the command, the prompt now sees everything within the AgAuto folder.

1. In the prompt, type *pip install REQUIREMENTS.txt* as shown in Figure 6. This looks through the REQUIREMENTS.txt file within the AgAuto folder to install the necessary packages listed there.

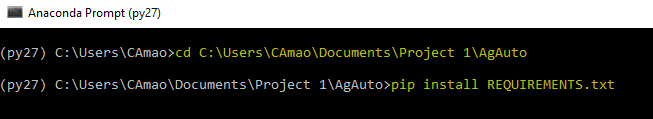
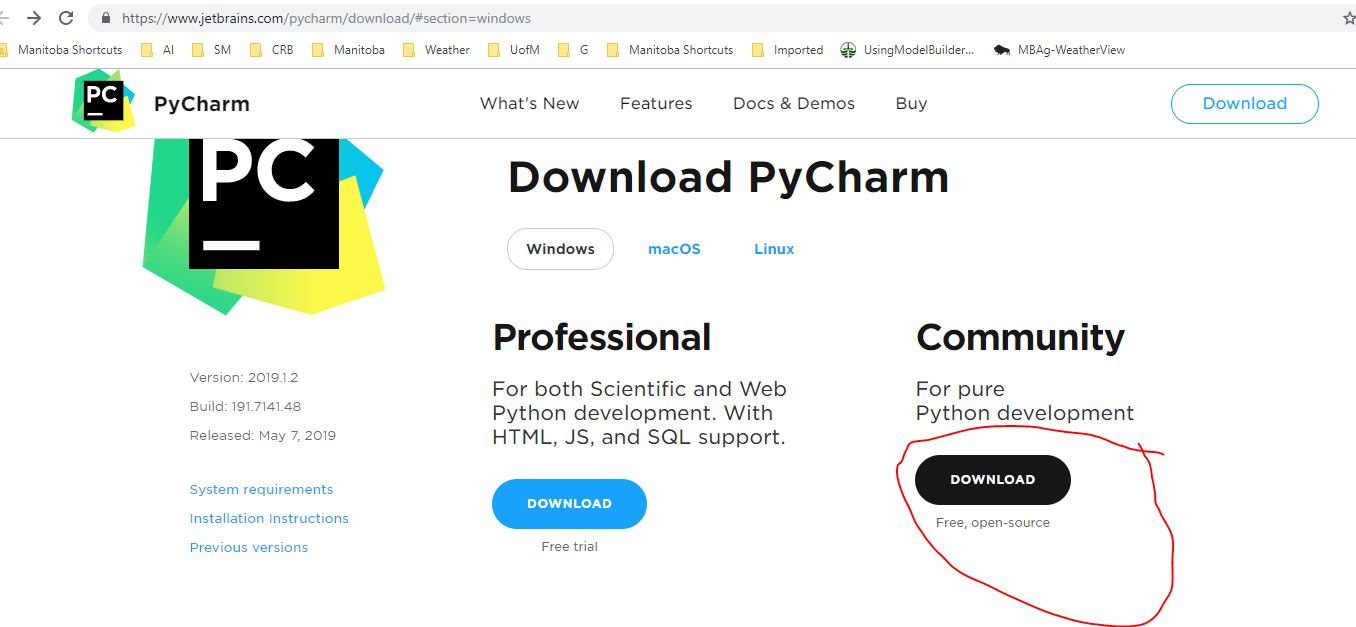


Figure 6: Using pip install command to install necessary packages within REQUIREMENTS.txt. Anaconda is now properly setup for AgAuto.

**Installing Pycharm**

Once Anaconda is properly set-up, the Pycharm IDE must be installed in order to run the program and edit the code should you wish to change it in the future.

1. Navigate to <https://www.jetbrains.com/pycharm/download/#section=windows> and download the Community version for Windows.



1. Once PyCharm has been downloaded, proceed with the installation.
   1. If prompted, check the “Create Desktop Shortcut”, “Update context menu” and “Create Associations for .py”. Leave other options as defaults.
2. Once PyCharm has been successfully installed, navigate to the folder containing *AgAuto.py* and open *AgAuto.py* using PyCharm.
3. Once AgAuto has been created, press ‘Configure Python Interpreter’ in the top right corner.

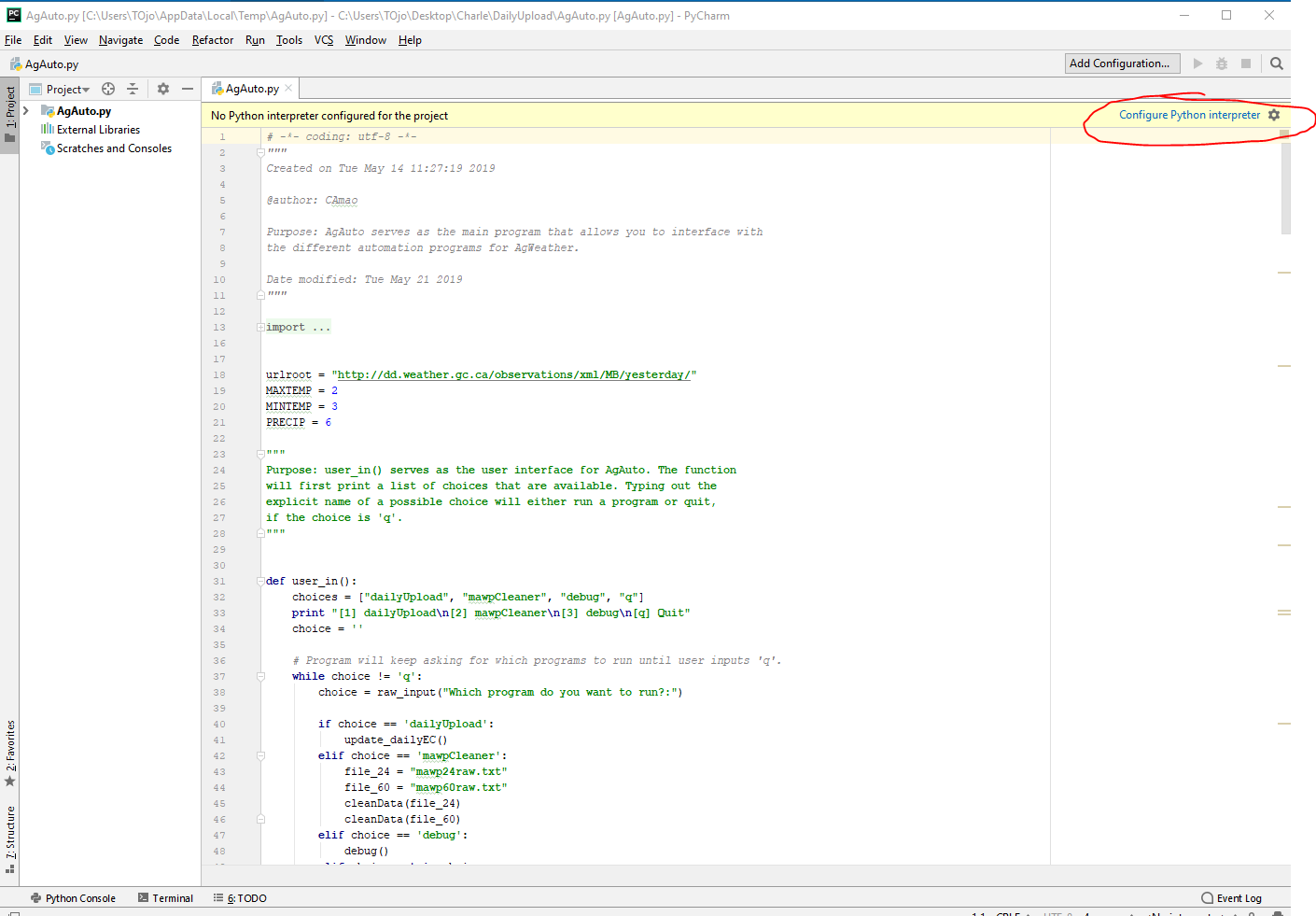


Figure 7: PyCharm needs to know which version of Python to run. In this case, we need to run the py27 virtual environment we created earlier.

1. Select the gear icon in the top right corner. Then select ‘Conda Environment’ as shown in Figure 8.

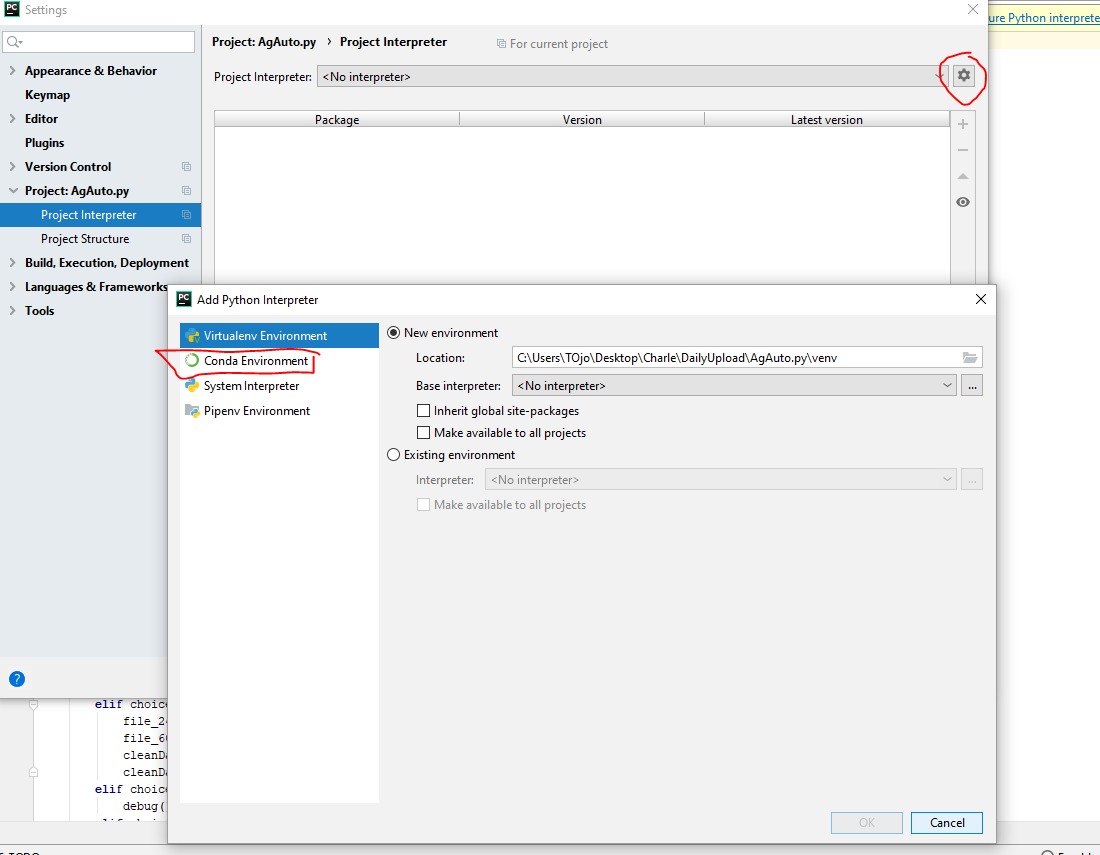


Figure 8: We need to tell PyCharm that we wish to run a conda virtual environment.

1. Navigate to where the python executable for py27 is located, as shown in Figure 9. Then press OK.

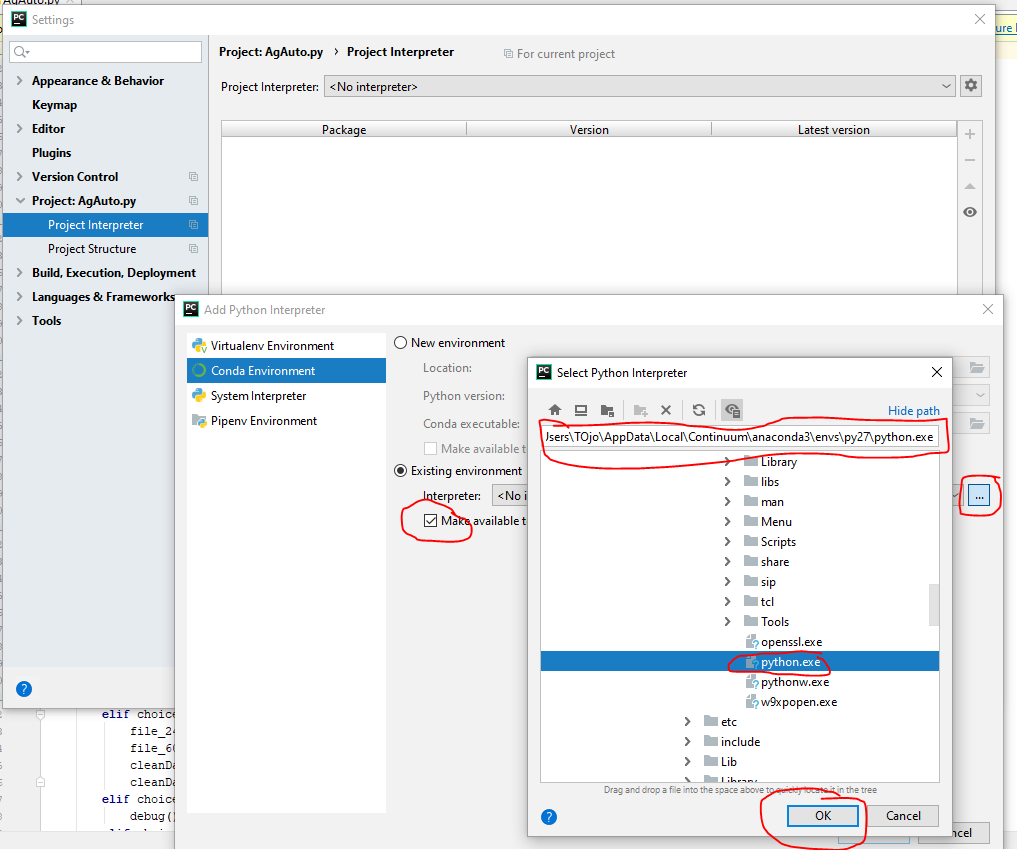


Figure 9: Locate the py27 python executable. Usually it's located in the link shown above.

1. Finally, press Apply and then OK to complete the process. It might take PyCharm 10 minutes to fully load the packages.

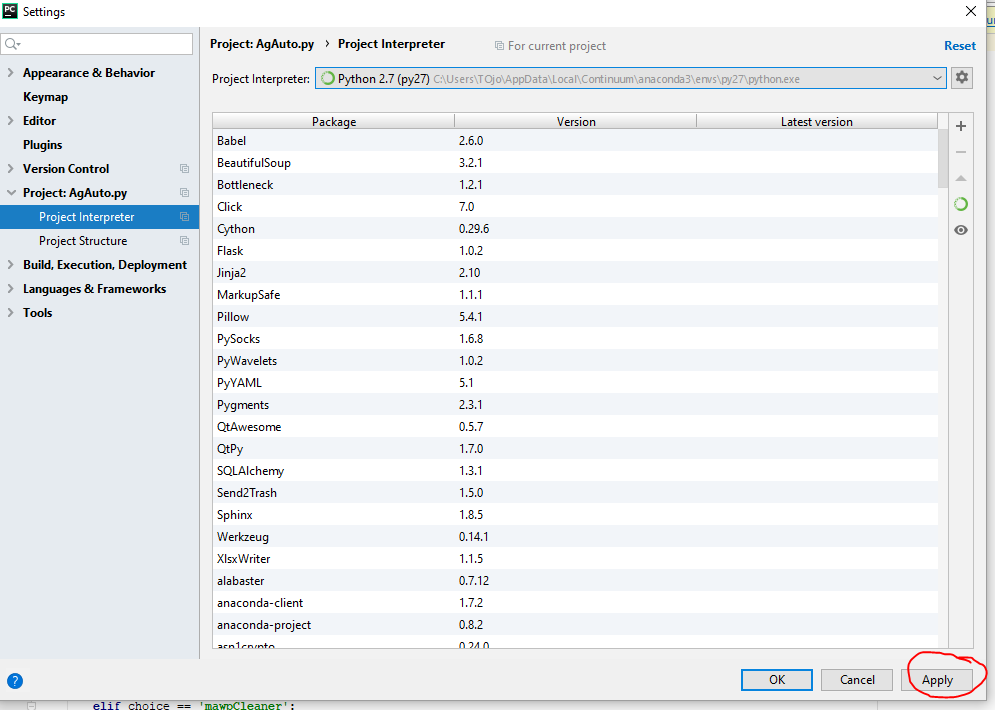


Figure 10: Finalize the interpreter selection process.

# Procedure

NOTE: In order to run the script, you must ensure that your computer is outside of the managed internet environment; connecting to your phones WiFi will suffice. However, in order to place the new files into the upload folder you must switch back to the managed environment.

1. As shown in Figure 11, navigate to the AgAuto working directory and double click AgAuto.py to run open the script using PyCharm.

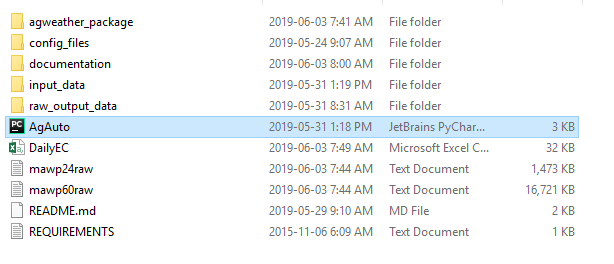


Figure 11: The AgAuto folder containing all the python scripts.

1. Once Pycharm is opened, right click on the AgAuto tab and click *Run ‘AgAuto’*.

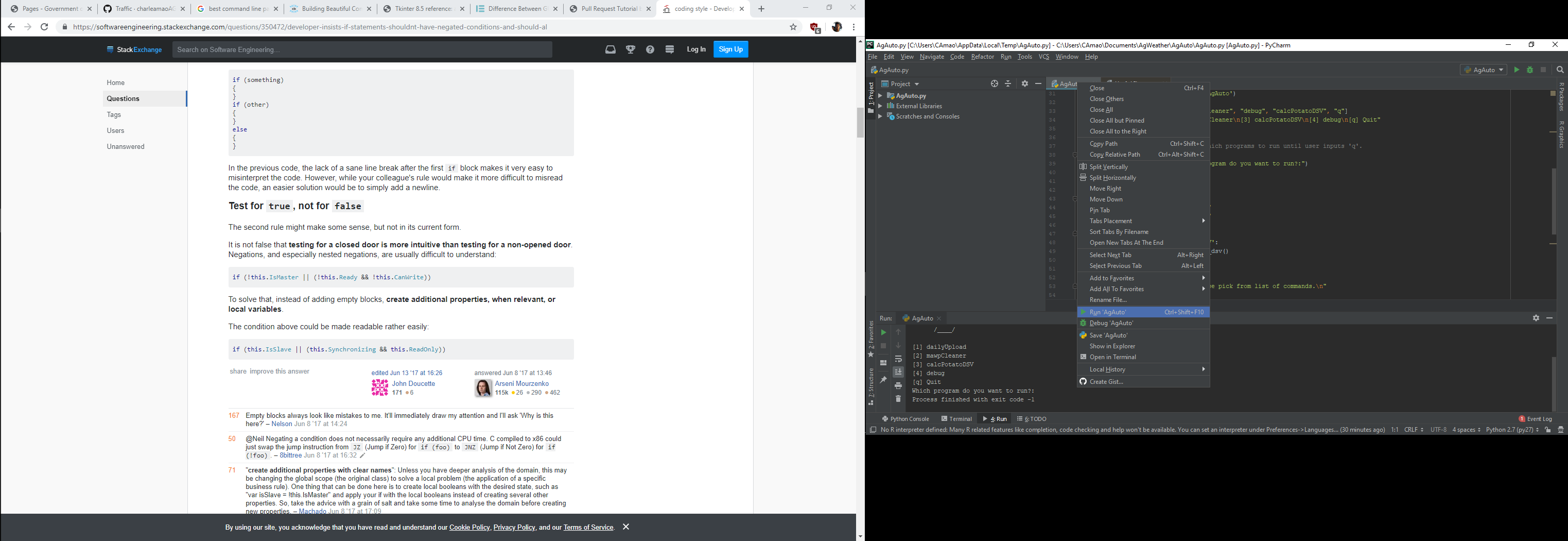


Figure 12: One of two ways to run the program, this is the easiest and most consistent.

1. After a few moments, you should see an output in the console similar to the one shown in Figure 13. Once prompted, type *calcPotatoDSV* and press ENTER.

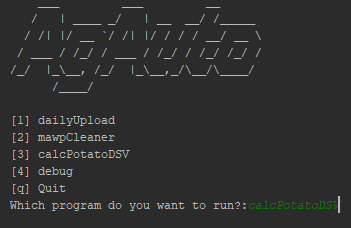


Figure 13: Although there are other choices, calculating DSV only requires calcPotatoDSV.

1. After pressing ENTER, you should see the prompt shown in Figure 14. Enter the date with the format YYYY-MM-DD. For example, if you wish to enter May 2, 2019, then type in *2019-05-02*. Also, ensure that the specified seed date is within the valid data range from *mawp15.txt*. If the specified seed date is not valid, then the program will either error out or output an empty *station\_dsv.csv* file.

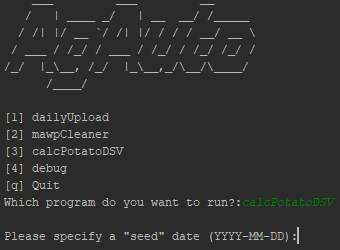


Figure 14: Ensure that the format is exactly YYYY-MM-DD or else the program will fail.

1. After running the command you should see two progress bars appear, as shown in Figure 14. Wait for ‘Calculating station DSVs’ to reach 100%.



Figure 15: The progress bars you see after running the command.

1. If the program displays a warning flag as shown in Figure 15, then review the output csv and the *mawp15.txt* file from <https://mbagweather.ca/partners/win/mawp15.txt>. A warning flag almost always means that the station contains ‘7999’ (a sensor-specific error) in one or more of the data entries. If there are 10 or more such errors in one day, then the program will not consider this day in its calculations and will display the message shown below.

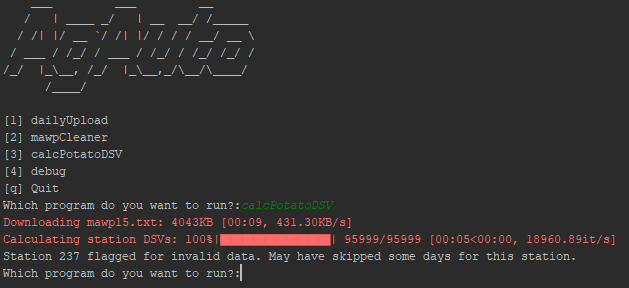


Figure 16: AgAuto may sometimes generate warning flags for stations with invalid data.

1. Once finished, navigate to the *raw\_output\_data* folder and review the *station­\_dsv.csv* file.

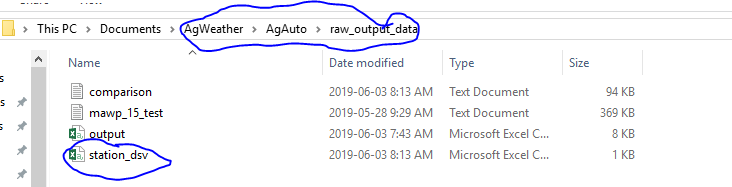


Figure 17: raw\_output\_data contains all output files as a consequence of running AgAuto.