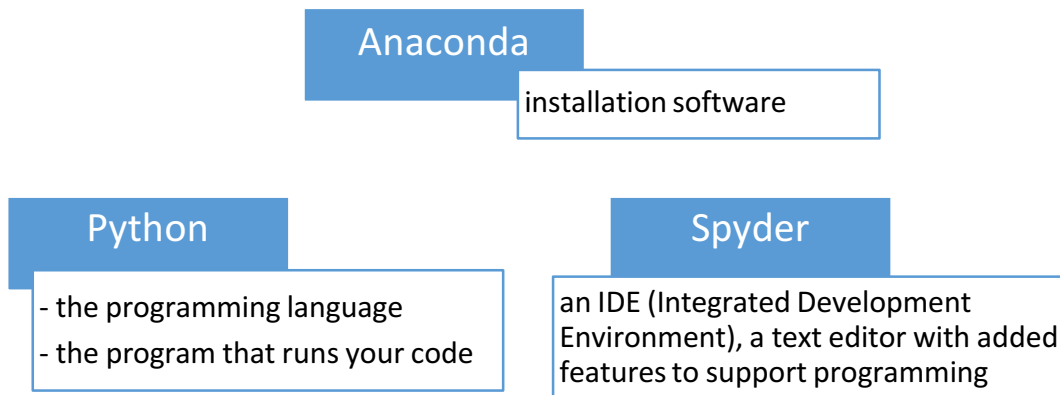# GETTING STARTED WITH PYTHON

Python is a programming language, and you'll be writing your Python code in the programming environment called Spyder. The Anaconda distribution simplifies the installation process by including Python, Spyder, and other packages and tools in one installation file.
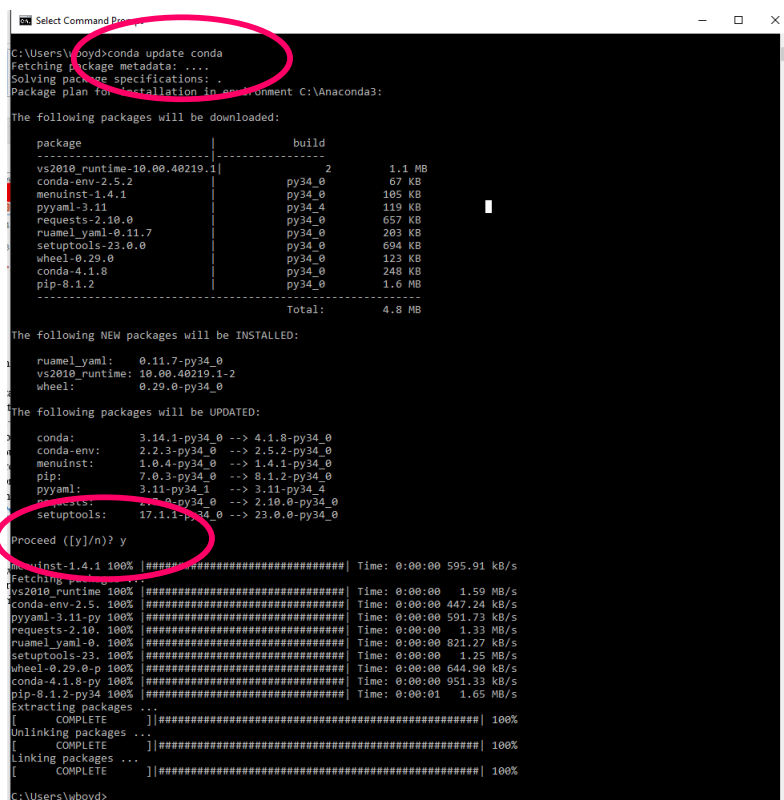
Anaconda

installation software

Python

- the programming language
- the program that runs your code

Spyder

an IDE (Integrated Development Environment), a text editor with added features to support programming

Installing Anaconda

1. Go to http://continuum.io/downloads

2. Scroll down to find your operating system and click on Python 3.5 to download the graphical installer.

    a. For windows, see http://windows.microsoft.com/en-us/windows/32-bit-and-64-bit-windows to find out whether your computer has a 32-bit or 64-bit version of windows. Click on the FAQ "*How can I tell if my computer is running a 32-bit or 64-bit version of windows?*", and follow the instructions to find out. If your computer is running a 32-bit version of Windows, click on "Windows 32-bit Python 3.4 Graphical Installer" under OTHER INSTALLERS, instead of the Windows 64-bit installer, and follow the same directions to install it. Actually, any Python 3.x version from Anaconda should work for this course. Python 2.x will not work for any version x.

    b. If your mac OS is prior to 10.7, see the FAQ. _____

3. Save the file to your computer.

4. Double click on the downloaded file to open it.

5. Follow the on-screen installation instructions, leaving options as they are currently set. This finishes the installation process.

6. (OPTIONAL) Next, check for any updates using Conda. Conda is one of the extras that is installed through the distribution Anaconda. It handles things like updates, set-up, and package installation through a command line interface. If there are many updates this can take 10 minutes or more. It is rare that you will have to use it.
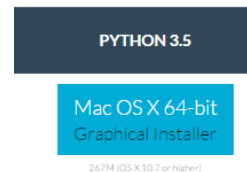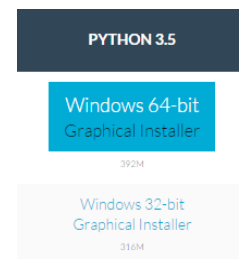
WINDOWS:
  a. Open Anaconda Command Prompt. Start typing Anaconda Command Prompt into the search box in the start menu, and it will show up.
  b. Type `conda update conda` at the command prompt, typing `y` for Yes and then pressing enter when it asks if you want to proceed. Your installation may identify different packages that need updating.
  c. After that completes, type `conda update anaconda` at the command prompt. If it prompts you to proceed with installation or updating, type `y` for Yes and press enter.
  d. After that completes, you can then close the command prompt window.

MAC:
  a. Open Terminal, which you can find by using Spotlight.

  b. Type `conda update conda` at the command prompt in the terminal, typing `y` for Yes and then pressing enter when it asks if you want to proceed. Your installation may identify different packages that need updated (see Windows image above for the text shown in the terminal window).
  c. After that completes, type `conda update anaconda` at the command prompt. If it prompts you to proceed with installation or updating, type `y` for Yes and press enter.
  d. After that completes, you can then close the command prompt window.
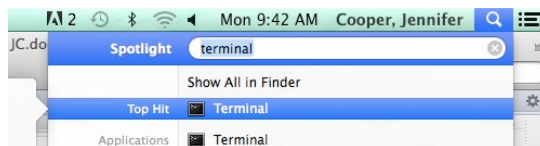
Getting to Know Spyder

You will write your programs and run them inside the Spyder IDE.
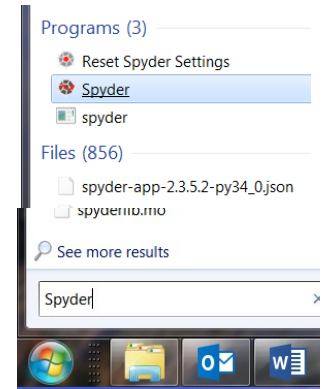
Spyder

The Scientific Python Development Environment

There are various ways to find and open this program – we only describe one way below.

On Windows, in the search box associated with the start menu, type Spyder. Click on the program with its logo. Right clicking on it will give you the option to pin it to your start menu or taskbar for easier access in the future.

On a Mac, we have two ways of running Spyder. We'll go through the **older** way first using **Launcher**. However, you are likely to have the **new** version that uses **Anaconda-Navigator** and should skip down below to that. What is happening is that sometime in November, 2016, Anaconda changed to the new version. Which you have doesn't matter, but the newest version of Anaconda using Navigator. You'll have to use whichever version your installation provides. If you did the optional step 6 above, you will have the latest version and it will use Navigator, not Launcher.
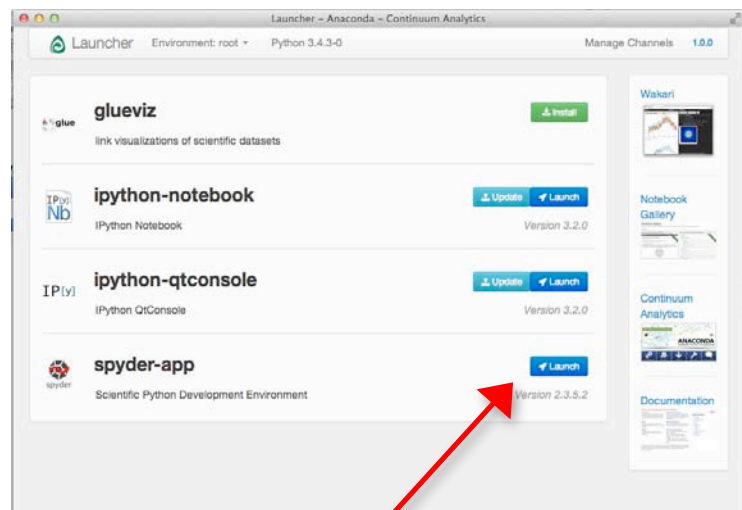
Older way of starting Spyder on a Mac using Launcher.

On a Mac, use Spotlight to find the program **Launcher**. It has a green tear-drop shaped icon associated with it. If you don't find Launcher, then skip to the newer way below using Anaconda-Navigator.
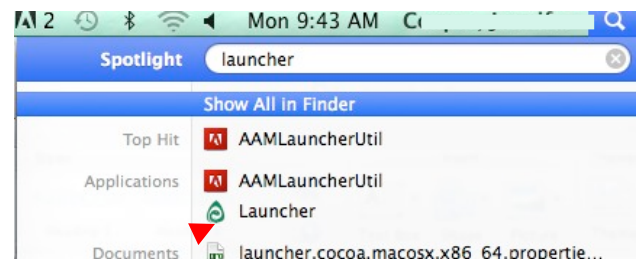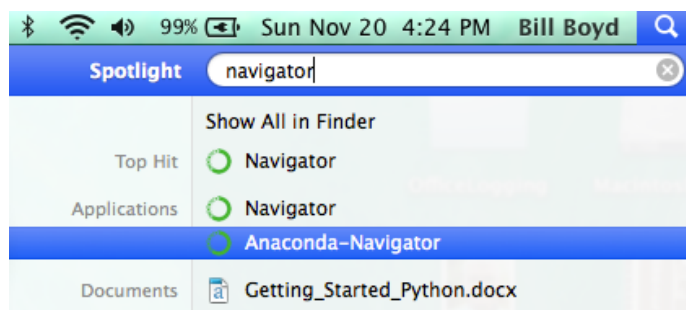
Running Launcher opens the following window, in which you will click Launch, on the last line for **spyder-app**.

FYI: The icon for Python that goes in the OS X dock, once Spyder has started, is generic, and keeping it on the dock will not work.

3

New way of starting Spyder on a Mac using Anaconda-Navigator.

On a Mac, use Spotlight to find the program **Anaconda-Navigator**. It has a green circular-shaped icon associated with it.

Running Anaconda-Navigator opens the following window, in which you will click Launch, underneath the **Spyder** icon.

FYI: The icon for Python that goes in the OS X dock, once Spyder has started, is generic, and keeping it on the dock will not work.
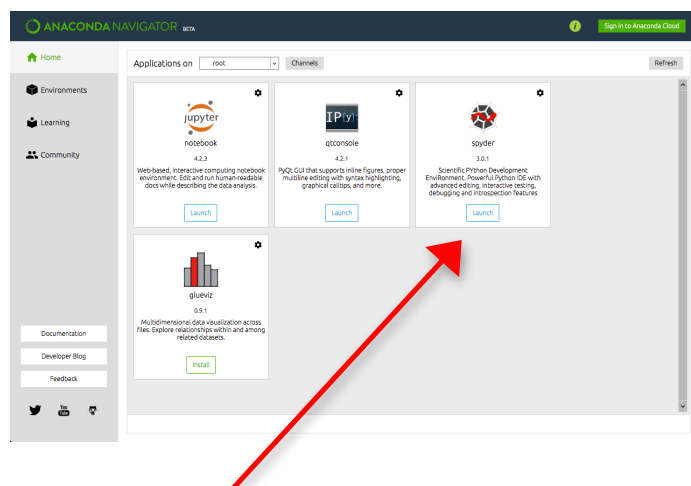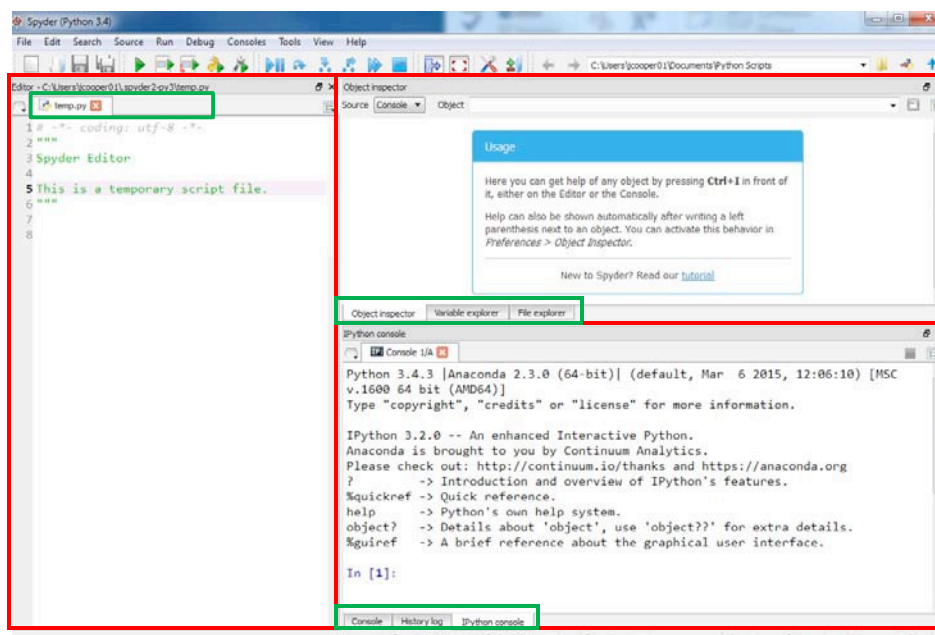
After Spyder has launched, you should see an interface that is organized into multiple windows (marked with red rectangles below) has or could have multiple tabs (marked with green rectangles below).
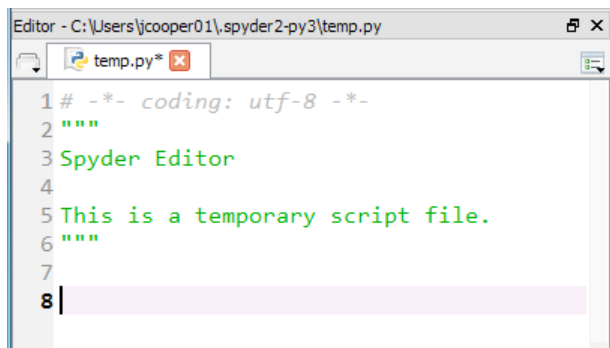
4

The default starting screen has three windows visible: Editor, Object Inspector, and Console.

If you accidentally close a window or change the layout, you can revert to the default by clicking "View" on the menu bar and then clicking "Reset Window Layout".

- **Editor** – where you can write and save sequences of commands – essentially, where you will write your full program
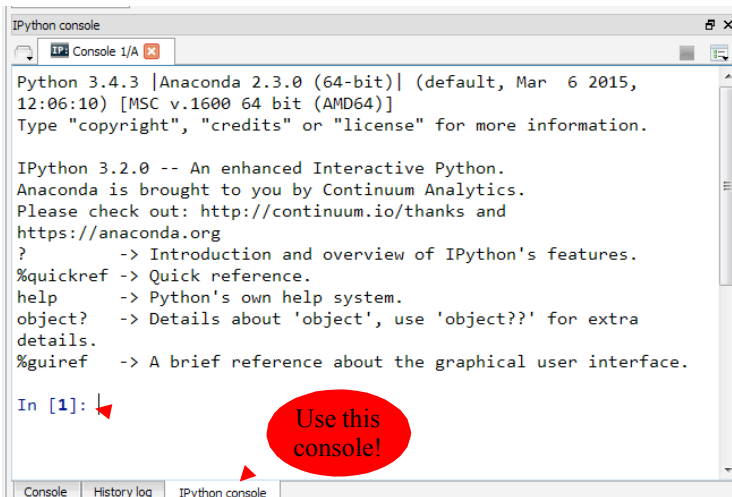

In the sample script that shows up in the editor, anything after # is a comment, meaning that Python will ignore that text. Three double quotation marks are a special type of comment that spans multiple lines.

Here, you would actually begin writing your syntax on line 8 after the comments that say what your file does.

- **IPython Console or IP Console** – where Python runs your code.

Spyder offers two choices for a console – a basic one and an IPython (Interactive Python) console. It is the IPython console we want to use. We will often refer to it, however, as just the console.

The prompt (where you can type in a command) for an IPython console is
In [1]:

Caution: If your console looks like the one shown to the right, where the prompt is **>>>** then you are in the *wrong* type of console and need to click on the tab for the IPython console.

- **Object Inspector** – On starting Spyder, the object inspector (as shown above) just gives you a link to the tutorial (this can be useful if you want to know more details about using Spyder). The rest of the time, this window will continue to give you information.

    The **Object Inspector** gives you more information about a function – it gives you the documentation from the help files and tells you what parameters (or input) the function takes. **Note**: In the latest version of Spyder this is called **Help**.

    The **Variable Explorer** shows you the value and type of any variables you have created. In the image to the right, we created three variables in the console, and those three variables showed up in the Variable Explorer. This is useful when you are trying to debug your code.
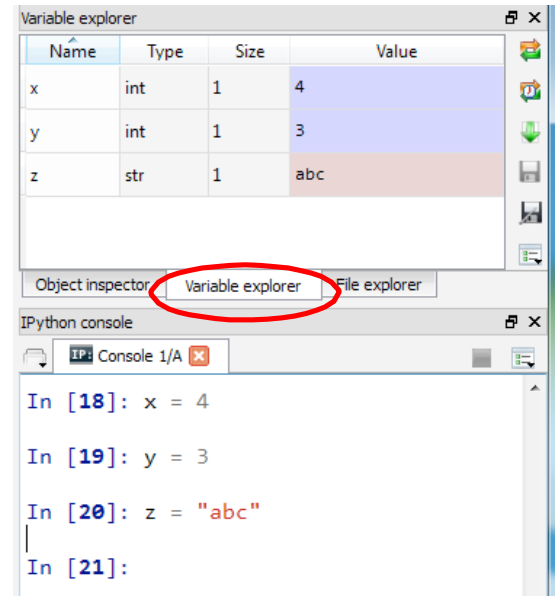
Frequently Asked Questions about Installation

- I already use Python 2.7 or I would rather use Python 2.7. Will that work for this course or do I need to install Python 3.x?
    - No. Python 2.7 will not work. The programs are auto-graded using a Python 3 system and it will not be able to compile Python 2.7 programs, because Python 2.7 is not compatible with Python 3.x. Yes, this is unusual. But the developers of Python decided to change certain features when going to Python 3 that made it impossible for them to maintain complete compatibility with Python 2. However, should you wish to shift to Python 2, later, you will be able to adapt at the expense of very little time and effort. Various websites, including https://wiki.python.org/moin/Python2orPython3, outline some of the syntactic differences.

- I have a different IDE that I like to use. Will that work for this course?
    - You could, but it will take more effort than it is worth. One big benefit of the Anaconda installation is that it includes IPython and IPython notebooks. These allow us to keep a whole lesson in one file, separated into cells. We can easily execute the code in a single cell, so that we don't have to write complete programs to run and test our code. After you finish the course, you can change to a different IDE with little effort, however. Most IDEs are similar to Spyder, but don't include the teaching convenience of the cells.