

Introduction to Python Workshop  
GRAD 778 – Module 7  
Preparatory Information and Installation Instructions

**Overview.** Python is a free, open-source, widely-used programming language that includes applications across disciplines, and integrating it into your research workflows is a helpful skill. In this online workshop, we'll focus our attention on learning the fundamentals of the language, understanding the various data types and how they work, learn how to import and work with your own data, and have you work on some applied problems throughout. Each of the twelve (12) individual modules that comprise this workshop give you an overview of a particular topic, outlined in the Workshop orientation file. A disclaimer: No workshop of any kind can cover all of the case studies and applications specific to each discipline, but as you'll see, there are likely packages that you can find and use to help you, no matter what you're trying to do. This should help you feel comfortable getting started on your own, regardless of your discipline or area of study.

We'll work within the Anaconda Distribution, which is tailored to data science use. The target audience of this workshop is those with little to no prior experience with Python, so if you're interested in getting started in working with Python, this workshop is for you! For those with some experience already, you are welcome to review the basics and refresh on some things.

In the workshop, you'll get a chance to work with code that I've provided for you. Throughout, you'll either run code and view output, or modify and experiment with given code. You're also welcome to keep and use these snippets of code for later use – you never know when it may come in handy. Each of these provide an interactive way to help explore the fundamentals of Python.

Finally, one point of clarity right from the beginning: I work almost exclusively in Windows (Windows 10), so all of the tutorials, videos, and screenshots will reflect that. As such, there may be slight variations in how things appear to you than they will to me. Just about everything in this tutorial should work regardless of supported OS (see software instructions below), but this is something to be aware of as you're following along.

**Software Overview.** This first step will involve installing the **Anaconda Distribution**, which we'll use for this workshop. We'll be using the Individual Edition, which is free and open source. We'll also be working in Version 1.9.12, which was released in February 2020.

Within Anaconda, we will be working with the Spyder Editor, which is one of the Integrated Development Environments (IDEs) that comes along with Anaconda and is visible when you first open the Anaconda Navigator. It is not the only one, of course. It is a nice place for beginners to start, though, and you should feel comfortable exploring some of the other editors as your skills grow.

As we'll see later in the workshop, Anaconda Navigator helps quite a bit with environment set-up, and package installation /management. As a bonus, downloading Anaconda 3 means that many of the popular packages are already installed! The benefit – and simultaneous drawback – to installing the full Anaconda Distribution is that you'll get hundreds of packages in the download. As beginners, this is a good environment in which to work, and you should do so for this Workshop. However, as you get more comfortable working in Python later, you may want to explore Miniconda

(<https://docs.conda.io/projects/continuumio-conda/en/latest/user-guide/install/index.html>), which only installs conda, its dependencies, and Python. We do not recommend working just in Miniconda for this workshop, but it's something to keep in mind for later work.

We also realize that some of you may already have Python on your machines due to either prior exploration of the language, or perhaps due to downloading other software that includes Python installation. **To be clear, we will be using Python 3.8 for the workshop, which comes along with downloading the Anaconda Distribution right now (“right now” = September 2020).** To complete this workshop, we encourage you to likewise work in Python 3.8, but you should be able to get through the workshop with earlier versions of Python 3 (e.g., Python 3.6.x, or Python 3.7.x), **but not 2.x.**

A note on Python software (especially older versions): Python 2.x is still fairly common. There are certain software packages that run with 2.x: a prominent example of this is ESRI's ArcGIS Desktop software. You can have both 2.x and 3.x installed on your machine, but it helps to know where they reside, particularly when it comes to installing and managing packages. If you have ArcGIS already installed on your computer, then you already have Python installed, but it is associated with ArcGIS. An IDE even comes along with it (IDLE), which allows you to start working with the standard Python library, along with the `arcpy` module. If this applies to you, then Python is probably installed a location similar to this one:

C:\ProgramData\Microsoft\Windows\Start Menu\Programs\ArcGIS\Python 2.7

*\*Regarding the `arcpy` module, which allows you access to the myriad tools, methods, and properties associated with ArcGIS Desktop in the Python environment: we won't be spending time working with the `arcpy` module in this workshop, but the fundamentals we learn will be essential to effectively using it for your own work. If you're interested specifically in using Python for GIS and/or spatial analysis, we do occasionally offer courses on Geospatial Analysis in Python in the Department of Geography.\**

If you are one of those users with Python 2.x, you should install 3.x (3.8, in this case) separately by downloading the full Anaconda Distribution (Anaconda3), per the instructions below.

### Software installation

Unless: 1) you already have Anaconda3 recently installed on your machine, or 2) you have a working Python 3 environment that you have spent a great deal of time setting up and would like to maintain, the easiest thing to do – and **strongly recommended strategy** – is to install the newest version of the Anaconda Distribution on your machine for this workshop.




*\*A note for those users who already has a working Python 3 environment set up on your machine, please feel free to use it for this workshop, but understand that there may some slight changes in how we use certain commands and that the instructions provided here may not always work seamlessly as-is. Of course, if this case applies to you, you are likely not new to Python (the primary target audience of the workshop), and are perhaps instead here to review and refresh.\**

Let's walk through the installation scenarios

1) For the majority of you, you'll need to install the most recent version of the Anaconda Distribution. This will: install Python 3.8, the most recent version of every editor we'll use in the workshop, and newest versions of the packages that come along with the Anaconda Distribution.

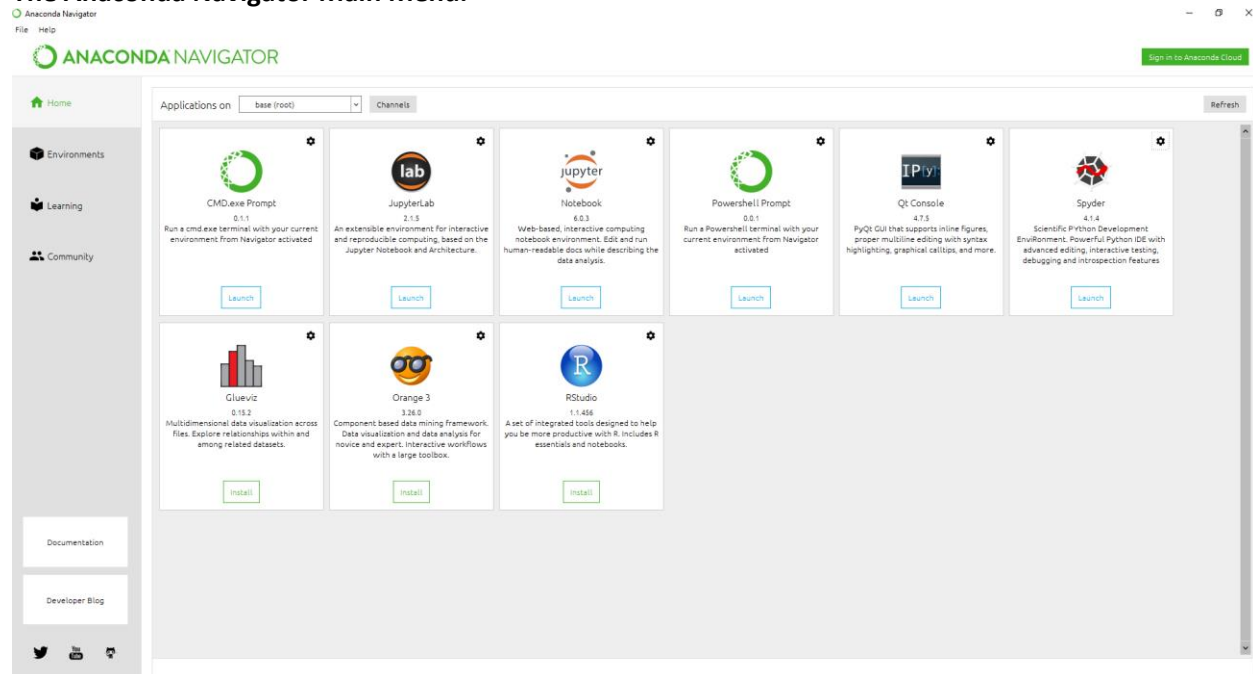
To do so, go to: <https://www.anaconda.com/distribution/#download-section>, and scroll down to select the version for your operating system (Mac, Windows, or Linux). As a side note, just in case you see it,

Anaconda2 includes Python 2.x, so don't install that for this workshop. The Python 3.8 version should be the version you see with the download options for each OS.

Windows 	MacOS 	Linux 
Python 3.8 <a href="#">64-Bit Graphical Installer (466 MB)</a> <a href="#">32-Bit Graphical Installer (397 MB)</a>	Python 3.8 <a href="#">64-Bit Graphical Installer (462 MB)</a> <a href="#">64-Bit Command Line Installer (454 MB)</a>	Python 3.8 <a href="#">64-Bit (x86) Installer (550 MB)</a> <a href="#">64-Bit (Power8 and Power9) Installer (290 MB)</a>

Click on the Installer of your choice, which will initiate the download. Go to this link: <https://docs.anaconda.com/anaconda/install/>, then follow the instructions relevant to your operating system to complete the process: Follow the prompts, and when finished, open Anaconda Navigator. The screen should appear similar to the screenshot included below

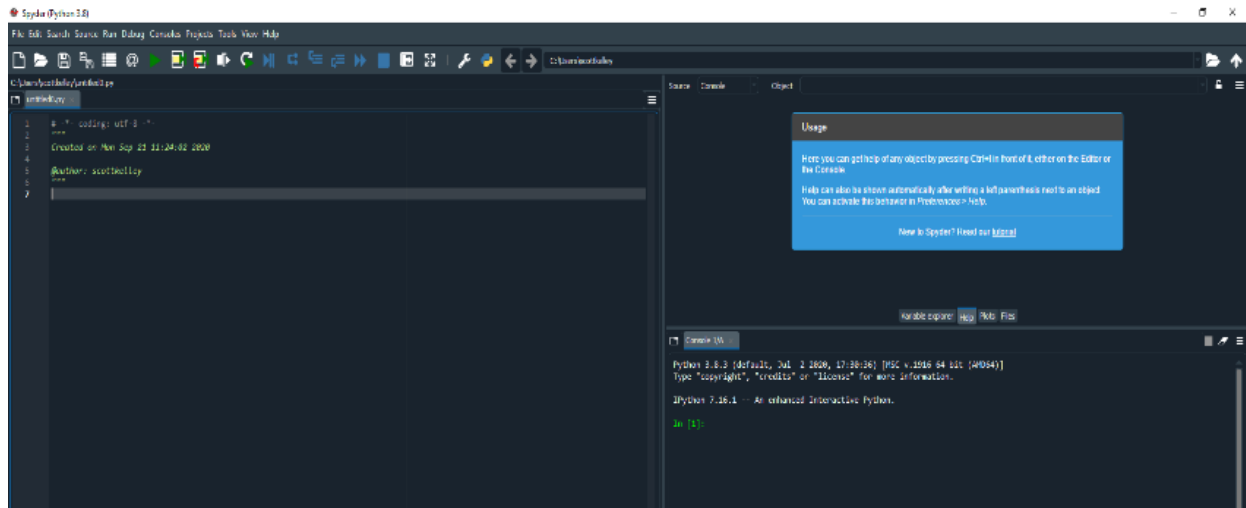
### The Anaconda Navigator Main Menu.



From this screen, next launch Spyder from the Anaconda Navigator window to test to ensure all is well – either 4.1.4 or 4.1.5 are fine. If you see the second screenshot included below, you're set for the workshop!

\*Note on the screenshot: the key things to look for are 1) that “Spyder (Python 3.8)” is at top left, and 2) in the console at right, Python 3.8.3 is the version you see. If those are there, you're set.

## The Spyder IDE environment.



*The dark background you see in this screenshot is set through “Tools – Preferences – Syntax highlighting theme – Spyder Dark”. You can adjust to your liking now or after you begin working.*

**If you have a recent Anaconda3 on your computer already**, you probably are all set to start the workshop modules. Just check to make sure you're running Python 3.8 (either 3.8.1, 3.8.2, or 3.8.3 are all fine). **If you have Python 3.x installed already**, but not 3.8, you can update using conda, if you have Anaconda3 already installed, too. Go to this link to read more about how to update or upgrade using conda, if needed. <https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-python.html>. Scroll to the sections about updating or upgrading Python.

**If you have an older version of Anaconda3 on your computer already**, but you haven't used it in awhile, you should probably upgrade for this workshop. You can uninstall the old version of Anaconda Distribution, then reinstall with a new version if you want, but depending on how old your version is, you may be able to simply update your existing version. See here for details: <https://docs.anaconda.com/anaconda/install/update-version/>.

**If Python 2.x (likely 2.7) is installed on your computer but not 3.x**, you should install a fresh copy of the Anaconda Distribution, following the instructions above. This will give you Anaconda3/Python 3.8. It is possible to download and work with different Anaconda Distributions that manage 2.x and 3.x separately, so if you know you will need to keep working in Python 2.x due to software demands (e.g., ArcGIS), you can install both Anaconda3 and Anaconda2. You can create different environments and manage packages separately for 2.x and 3.x using conda, which of course comes with Anaconda. Read more about it here, if you think you'll want to keep working in 2.x beyond the workshop: <https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-python.html>. Still, for the workshop, we'll be working in Anaconda3.

**If you are unsure/wary about** downloading any new software, you can actually access Anaconda Navigator (and therefore, the editors we'll use, like Spyder) through either the BlueWolf or SilverWolf DataWorks desktops through remote.unr.edu (what you see or have access to will depend on your role/permissions). This will get you access to Python 3.7.7, so it is a slightly earlier version than we'll be working with. Given that we are working largely with fundamentals in this workshop, you should be able to get through the modules just fine, though, except for when you need to set up environments and install packages. As with the other GRAD 778 modules, we recommend that you work on your own machine and local environment for this workshop, but this is an option that is available to you that you can use to get through the workshop if needed.