CSC462 Artificial Intelligence

LAB 1: Python Installation

This first Orientation Lab will help you get started by guiding you through the process of installing Python on your computer.

By the end of this lab, you will be able to:

- Describe why Python is a useful computer language for problem solvers
- Describe applications where Python is used
- Detail advantages of Python over other programming languages
- Know the cost of Python
- Know the difference between Python and Anaconda
- Install Python on your computer
- Install Anaconda on your computer

Why Python?

You might be wondering "Why should I solve problems with Python?" There are other programming languages in the world such as MATLAB, LabView, C++ and Java. What makes Python useful for solving problems?

Python is a powerful programming language

Python defines the types of objects you build into your code. Unlike some other languages such as C, you do not need to declare the object type. The object type is also mutable, you can change the type of object easily and on the fly. There is a wide array of object types built into Python. Objects can change in size. Python objects can also contain mixed data types. Strings and floating point numbers can be part of the same list.

Python has an extensive Standard Library. A huge number of object types, functions and methods are available for use without importing any external modules. These include math functions, list methods, and calls to a computer's system. There is a lot that can be done with the Python Standard Library. The first couple of chapters of this book will just use the standard library. It can do a lot.

Python has over 100,000 external packages available for download and use. They are easy to install off of the Python Package Index, commonly called PyPI ("pie pee eye"). There is a Python package for just about everything. There are packages which can help you: interact with the web, make complex computations, calculate unit conversions, plot data, work with .csv, .xls, and .pdf files, manipulate images and video, read data from sensors and test

equipment, train machine learning algorithms, design web apps, work with GIS data, work with astronautical data. There are and many more Python packages added to PyPI every day. In this book, we will use some of the more useful Python packages for problem solvers such as NumPy, Matplotlib, and SymPy.

Python is easy to learn and use

One way Problem solvers code solutions faster in Python faster than coding solutions in other programming languages is that Python is easy to learn and use. Python programs tend to be shorter and quicker to write than a program which completes a similar function in another languages. In the rapid design, prototype, test, iterate cycle programming solutions in Python can be written and tested quickly. Python is also an easy language for fellow problem solvers on your team to learn. Python's language syntax is also quite human readable. While programmers can become preoccupied with a program's runtime, it is development time that takes the longest.

Python is transportable

Python can be installed and run on each of the three major operating systems: Windows, Mac and Linux. On Mac and Linux Python comes installed out of the box. Just open up a terminal in on a MacOS or Linux machine and type `python`. That's it, you are now using Python. On Windows, I recommend downloading and installing the Anaconda distribution of Python. The Anaconda distribution of Python is free and can be installed on all three major operating systems.

Python is free

Some computer languages used for problem solving such as MATLAB and LabView cost money to download and install. Python is free to download and use. Python is also open source and individuals are free to modify, contribute to, and propose improvements to Python. All of the packages available on the Python Package Index are free to download and install. Many more packages, scripts and utilities can be found in open source code repositories on GitHub and BitBucket.

Python is growing

Python is growing in popularity. Python is particularly growing in the data sciences and in use with GIS systems, physical modeling, machine learning and computer vision. These are growing team problem-solving areas for engineers.

The Anaconda Distribution of Python

I recommend to install the *Anaconda distribution of Python*. The following section details the differences between the Anaconda distribution of Python and the version of Python you can download from Python.org

How is Anaconda different from Python?

When you download Python from Python.org, you get the _Python Interpreter_, a little text editing program called **IDLE** and all of the Python Standard Library modules.

The Python Interpreter is an application or program that runs your Python code. A program written in the Python programming language is run with the Python Interpreter. So Python corresponds to both the language that a program is written in as well as the application that runs the program.

When you download the Anaconda distribution of Python from Anaconda.com, you get a Python Interpreter, the **Anaconda Prompt** (a command line program), **Spyder** (a code editor) and about 600 extra Python modules that aren't included in the Python Standard Library. The Anaconda distribution of Python also includes a program called Anaconda Navigator that allows you to launch Jupyter notebooks quickly.

Why download Anaconda, if I want to use is Python?

Regardless if you download Python from Python.org or if you download Anaconda (with all the extra stuff it comes with) from Anaconda.com, you will be able to write and execute Python code. However, there are a couple of advantages to using the Anaconda distribution of Python.

Anaconda includes Python plus about 600 additional Python packages

The Anaconda distribution of Python is advantageous because it includes Python as well as about 600 additional Python packages. These additional packages are all free to install. The packages that come with Anaconda includes many of the most common Python packages use to solve problems. If you download Anaconda, you get Python including the Python Standard Library plus about 600 extra packages. If you download Python from Python.org, you just get Python and The Standard Library but no additional modules. You could install the extra modules that come with Anaconda (that don't come with plain old Python), but why not save a step (or about 600 steps) and just install Anaconda instead of installing about 600 different modules?

Anaconda installs without administrator privileges

Even if you don't have the ability to install programs on a computer, like a computer in a school computer lab, you can still download and use Anaconda. The Anaconda distribution of Python will also allow you to install additional modules from the Python package index (PyPI.org) and conda-forge (conda-forge.org), the conda package index.

Anaconda works on MacOS

If you use MacOS, you probably already have Python installed on your computer. Most MacOS installations come with Python included. The problem is that the version of Python that comes with MacOS is old (usually legacy Python, Python 2) and the version of Python that comes with MacOS is locked up behind a set of administrator privileges. Because the pre-installed version of Python included with MacOS can require administrator privileges, you can have trouble with installation and run-time problems. Some things will seem to work fine, and then other things won't run at all, or you will keep getting asked for an administrator password over and over.

Downloading and installing Anaconda (separate from the version of Python that came with MacOS) prevents most of the problems on MacOS caused by using the pre-installed version of Python.

Anaconda makes package management and virtual environments easier

Another advantage of Anaconda is that package management and virtual environments are a lot easier when you have Anaconda. Virtual environments and package handling might not seem to make a huge difference right now. If you just downloaded Anaconda for the first time, you are probably not dealing with package management and virtual environments yet. (It's OK if you don't even know what those two things are yet). After you write a couple of Python programs and start downloading a couple of extra modules from PyPI or conda-forge, dealing with package management and virtual environments becomes more critical.

Installing Anaconda on Windows

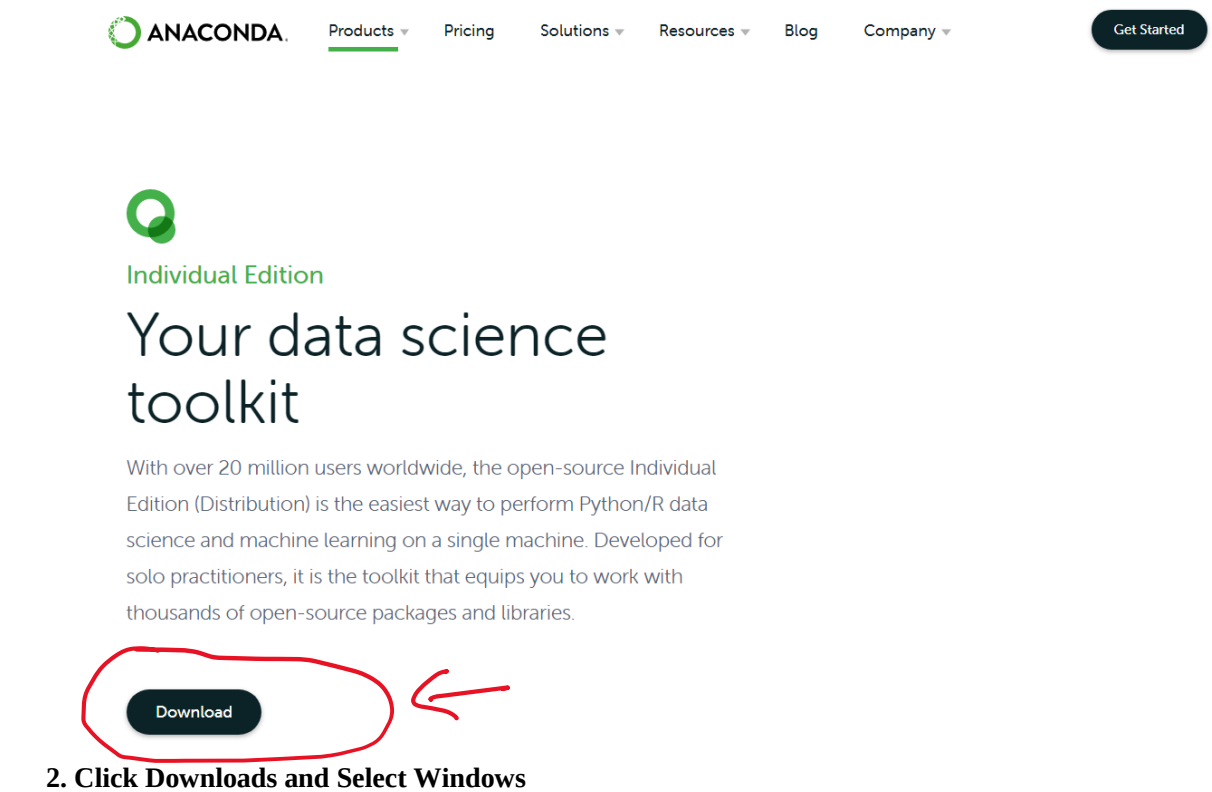
For problem solvers, I recommend installing and using the Anaconda distribution of Python.
This section details the installation of the Anaconda distribution of Python on Windows 10. I think the Anaconda distribution of Python is the best option for problem solvers who want to use Python. Anaconda is free (although the download is large which can take time) and can be installed on school or work computers where you don't have administrator access or the ability to install new programs. Anaconda comes bundled with about 600 packages pre-installed including **NumPy**, **Matplotlib** and **SymPy**. These three packages are very useful for problem solvers and will be discussed in subsequent chapters.

Follow the steps below to install the Anaconda distribution of Python on Windows.
Steps:

1. Visit Anaconda.com/downloads
2. Select Windows
3. Download the ***.exe*** installer
4. Open and run the ***.exe*** installer
5. Open the **Anaconda Prompt** and run some Python code

1. Visit the Anaconda downloads page
Go to the following link: Anaconda.com/downloads

The Anaconda Downloads Page will look something like this:



2. Click Downloads and Select Windows

Select Windows (32 or 64 bit according to your windows installation). If you are unsure if your computer is running a 64-bit or 32-bit version of Windows, select 64-bit as 64-bit Windows is most common.

3. Download

Download the most recent Python 3 release. The most recent release was the Python 3.8 Version. Python 2.7 is legacy Python.

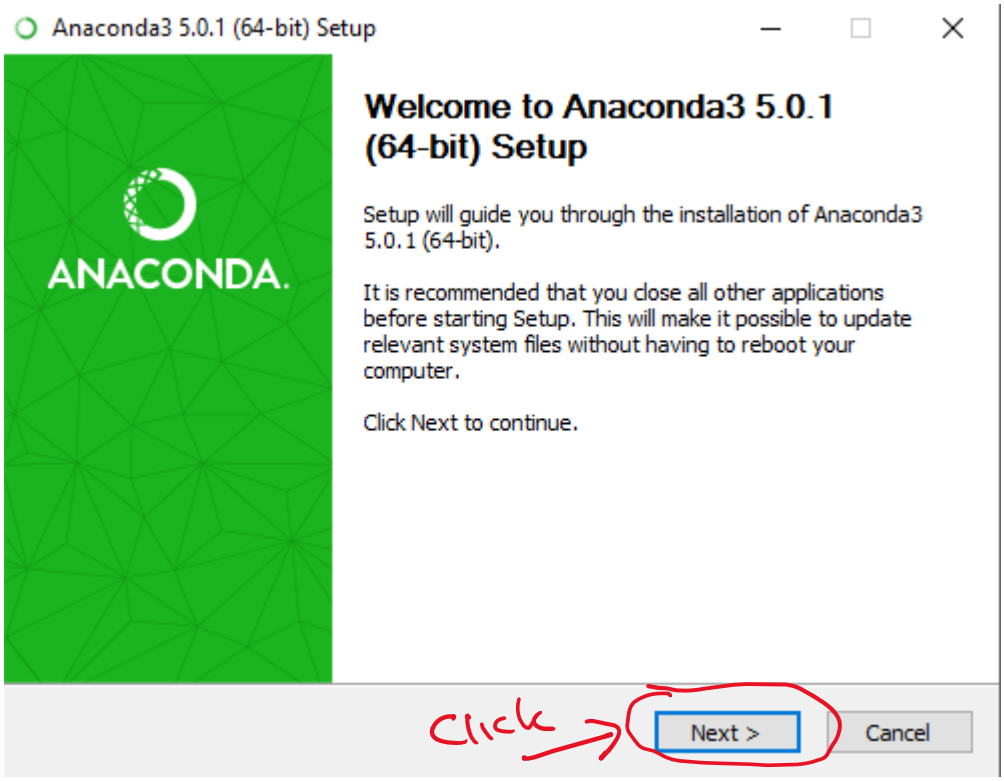
You may be prompted to enter your email. You can still download Anaconda if you click [No Thanks] and don't enter your Work Email address.

The download is quite large (over 500 MB) so it may take a while to for Anaconda to download.

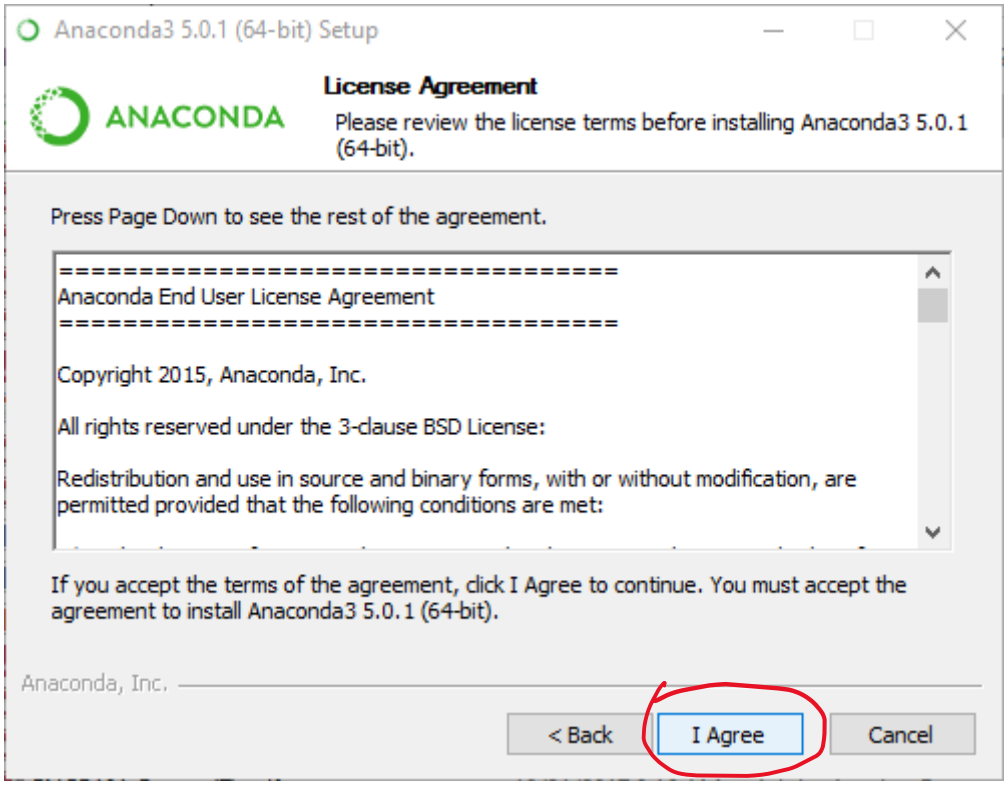
4. Open and run the installer
Once the download completes, open and run the ***.exe*** installer

Anaconda3-2020.07-Windows-x86_64.exe

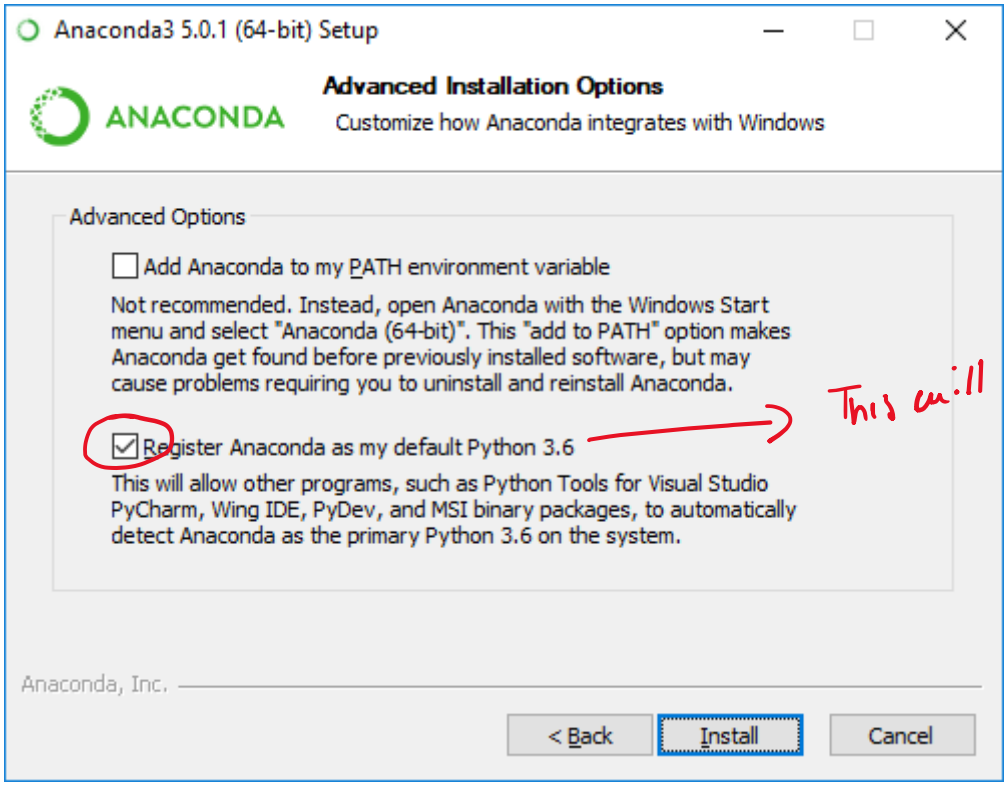
At the beginning of the install, you need to click **Next** to confirm the installation.



Then agree to the license.

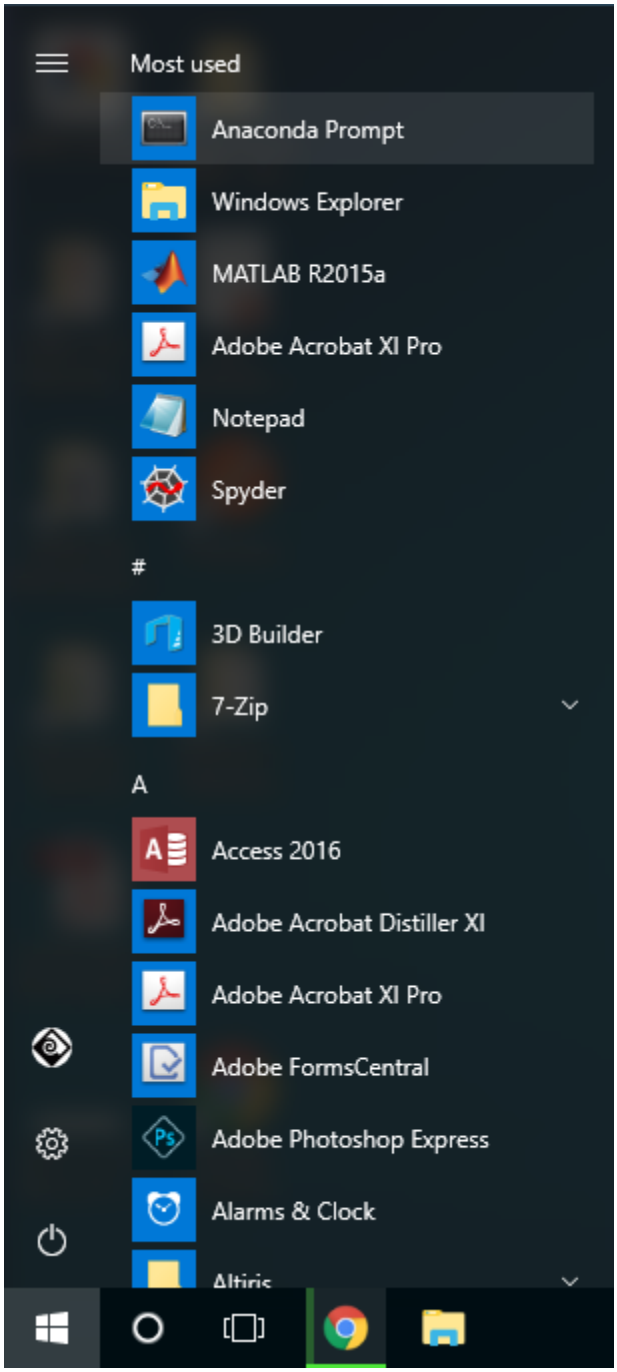


At the Advanced Installation Options screen, I recommend that you **do not check** "Add Anaconda to my PATH environment variable"



5. Open the Anaconda Prompt from the Windows start menu

After the installation of Anaconda is complete, you can go to the Windows start menu and select the Anaconda Prompt.

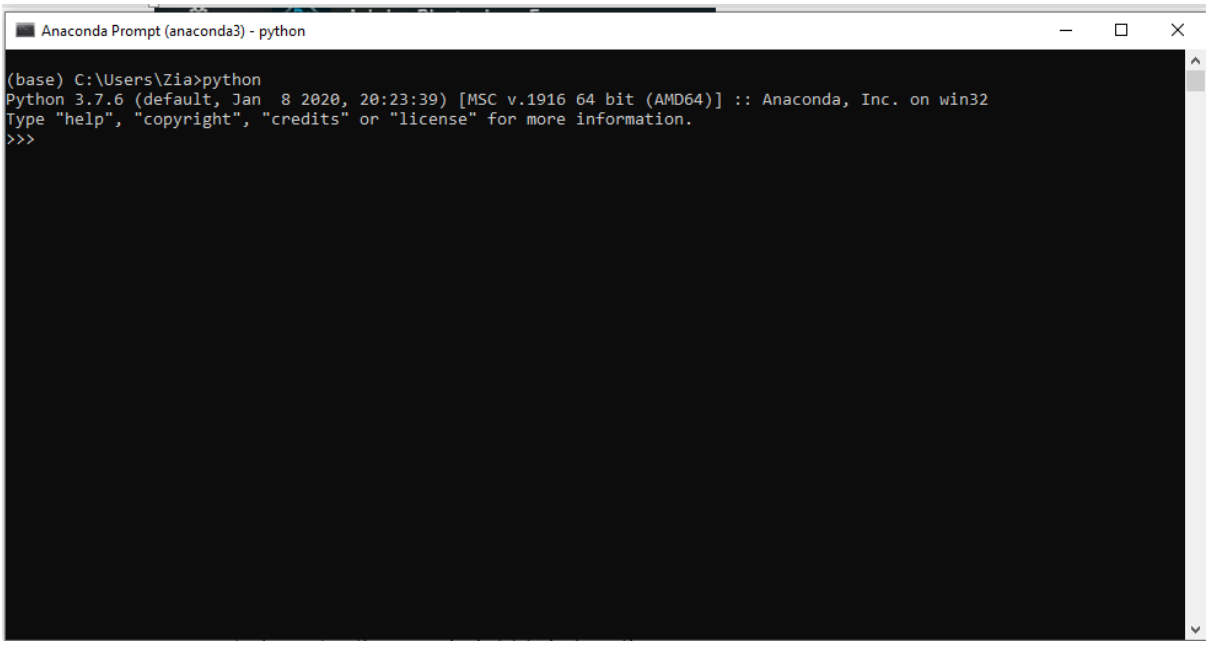


Anaconda prompt

This opens the **Anaconda Prompt**. **Anaconda** is the Python distribution and the **Anaconda Prompt** is a command line shell (a program where you type in commands instead of using a mouse). The black screen and text that makes up the **Anaconda Prompt** doesn't look like much, but it is really helpful for problem solvers using Python.

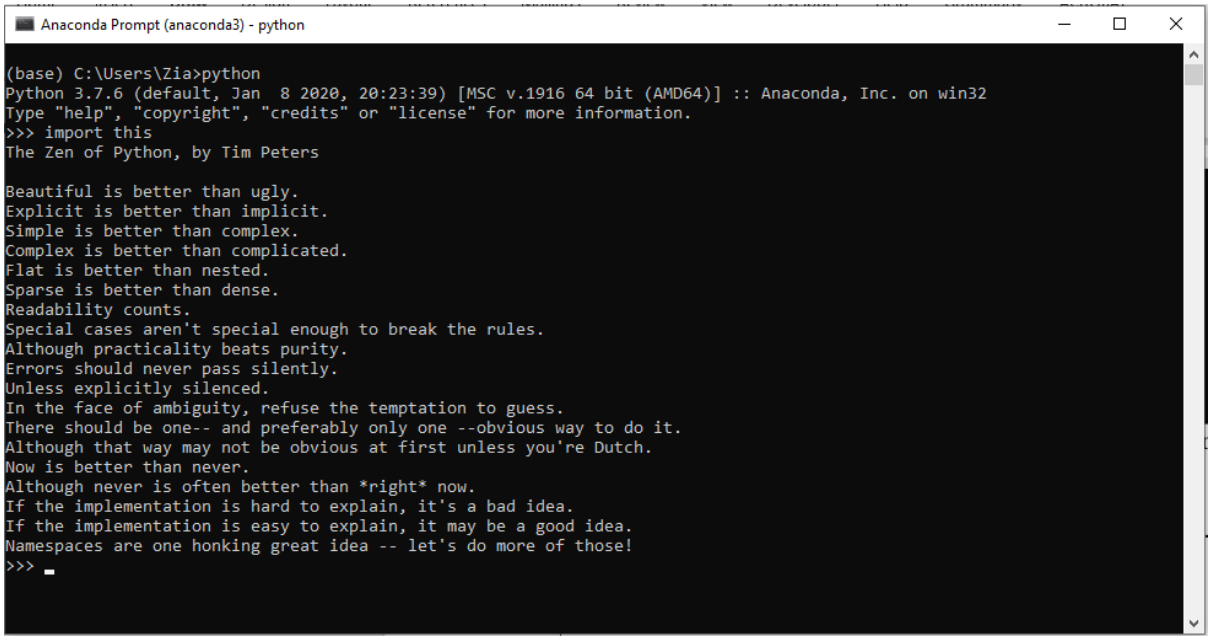
At the Anaconda prompt, type python and hit [Enter]. The python command starts the Python interpreter, also called the Python REPL (for Read Evaluate Print Loop).

```
> python
```



Note the Python version. You should see something like Python 3.8.1. With the interpreter running, you will see a set of greater-than symbols >>> before the cursor.

Now you can type Python commands. Try typing import this. You should see the ***Zen of Python*** by Tim Peters



To close the Python interpreter, type exit() at the prompt >>>. Note the double parenthesis at the end of the exit() command. The () is needed to stop the Python interpreter and get back out to the **Anaconda Prompt**.

To close the **Anaconda Prompt**, you can either close the window with the mouse, or type exit, no parenthesis necessary.

When you want to use the Python interpreter again, just click the Windows Start button and select the **Anaconda Prompt** and type python.

Summary

You learned about the Anaconda distribution of Python and how the Anaconda distribution of Python compares the version of Python at Python.org. The Anaconda distribution of Python comes with about 600 packages pre-installed as well as Jupyter notebooks and the Anaconda Prompt. Jupyter notebooks and some of the pre-installed packages that come with Anaconda will be used later.

Key Terms and Concepts

- Anaconda
- Anaconda Prompt
- Legacy Python
- Python Interpreter
- Python REPL
- package
- operating system
- Windows
- MacOS
- Linux
- terminal
- PATH