Getting Started with Python

Biochemistry Boot Camp 2023
Session #12
Nick Fitzkee
nfitzkee@chemistry.msstate.edu

What is Python?

- Simple, interpreted programming language
- Guido van Rossum, 1991
- Interpreted, not compiled: A program (*Python interpreter*) parses python commands/code and executes the statements, one at a time
- Compiled languages (like C and FORTRAN) use a *compiler* to convert code into machine language, which is run natively

Why (or Why Not) Python?

Pros	Cons

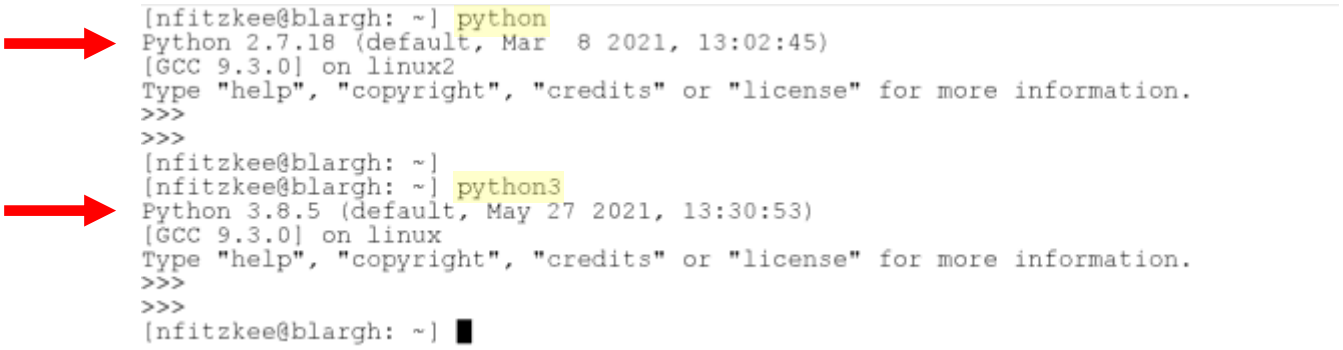
• Very simple syntax, easy to read
• Installed on most Linux, Mac
• Huge library of standard tools
  – Standard functions (file I/O, strings, lists)
  – Math/Science/Bio
• Many complex tasks are simplified (memory management)
• Relatively easy to get working code
	• Not as fast as C/C++/FORTRAN
	• Windows support not as good
	• Some technical tasks are more challenging (e.g., direct memory access)
	• Still requires rigor of learning programming

Python 2 vs. Python 3

- Different *versions* of Python interpreter exist (e.g. 2.7.18, 3.10.5)
  - New versions fix bugs, add features, extend the language
- Many syntax features from 2.x.x work in 3.x.x, but some do not
  - "under the hood" is very different
- Why? Simplify and streamline code, make syntax more consistent
- Python 2 officially discontinued in 2020, but a significant code base still exists

Do I Already Have Python?

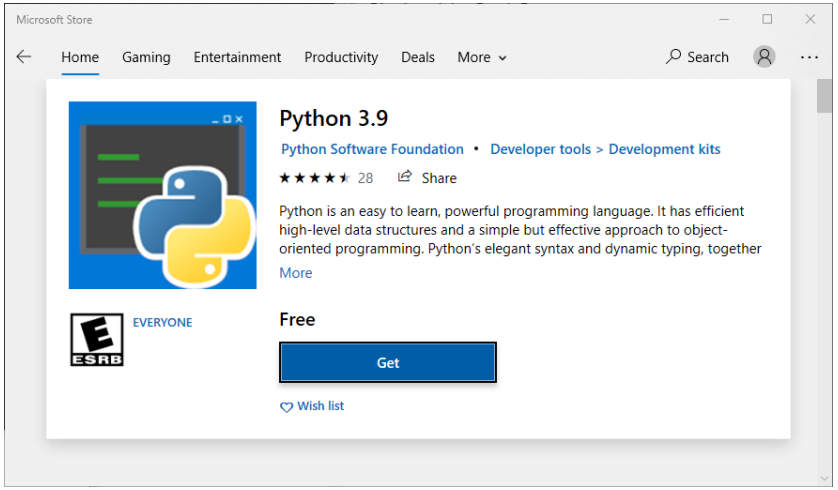
- Starting python at the command line: Simply type "python"
  - May need to type "python3" (Mac, some Linux systems)



- Python 3 is strongly preferred over Python 2
  - Most (not all) Linux systems will default to version 3
  - Most Macs use version 2 (legacy reasons)

Installing Python: python.org

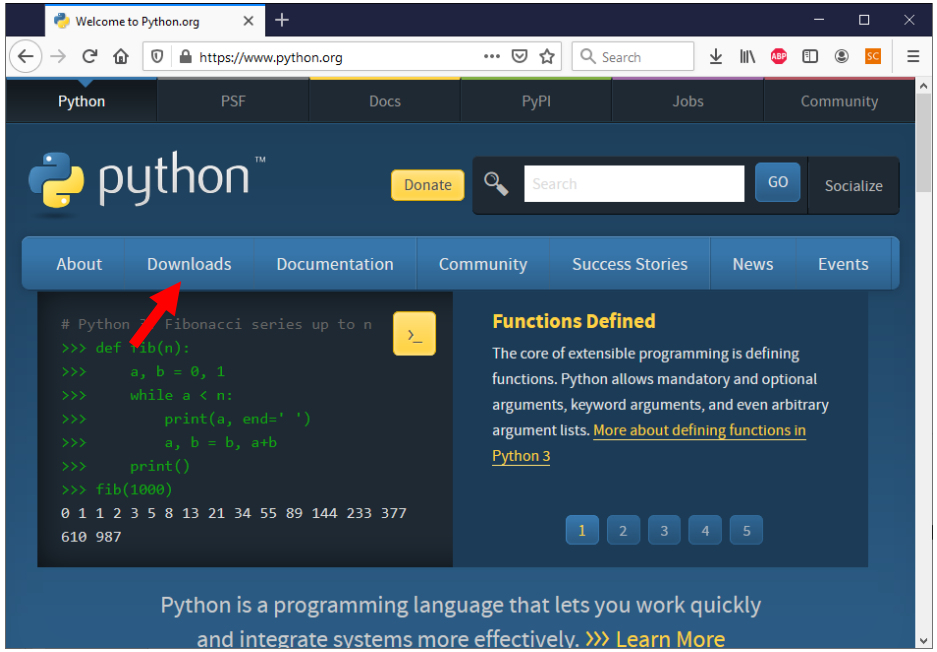
- **DON'T** use the Windows App Store:



- Windows App Store would be great, but:
  - Broken IDLE context menu
  - Asks you to log into Windows store (not really needed)

Installing Python: python.org

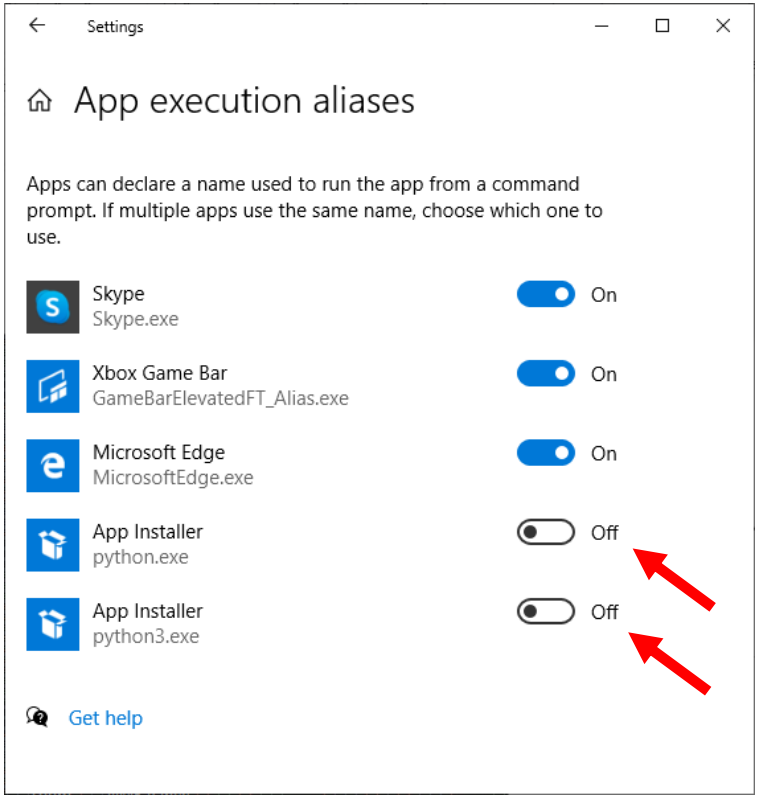
- For Mac and Windows, download the latest version www.python.org:



- Linux will require administrator privileges, but anything ≥ 3.7 should be fine

Disable Windows “Python” Shortcuts

- Typing “python” at the command line on Windows 10 (even if Python is installed) brings up the app store window
- Search “app execution” in the start menu, disable app installers for python and python3

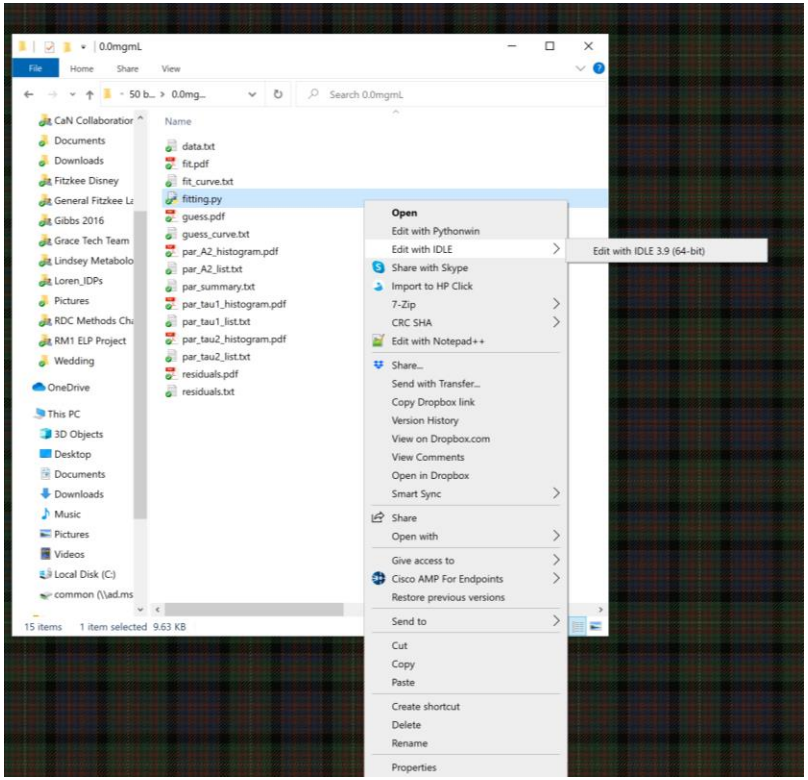


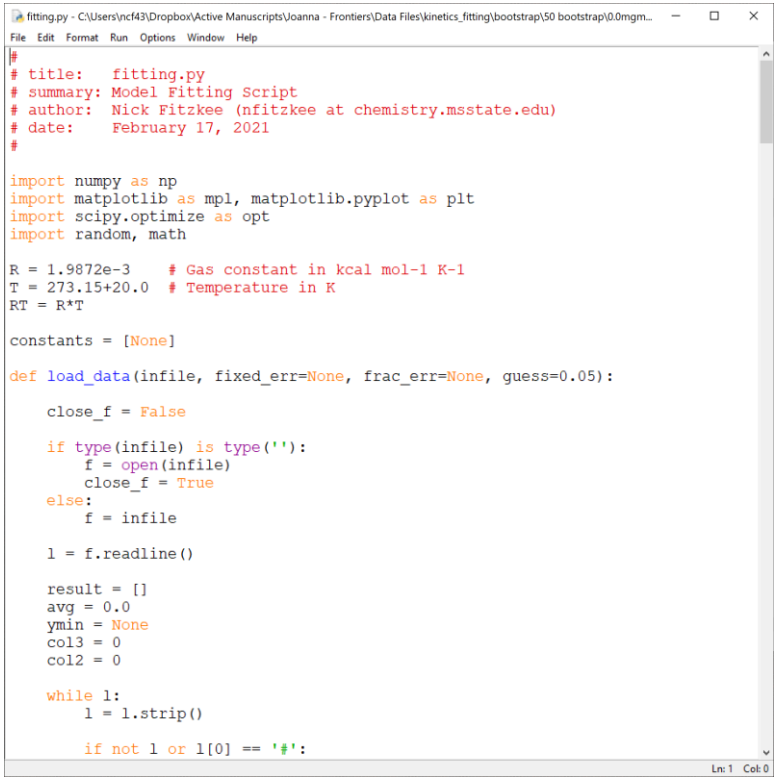
Make Sure Python Is In Your Path

- On Windows 10/11, most defaults are fine, but be sure Python is in your path:

- If you change other defaults, be sure to install pip, IDLE

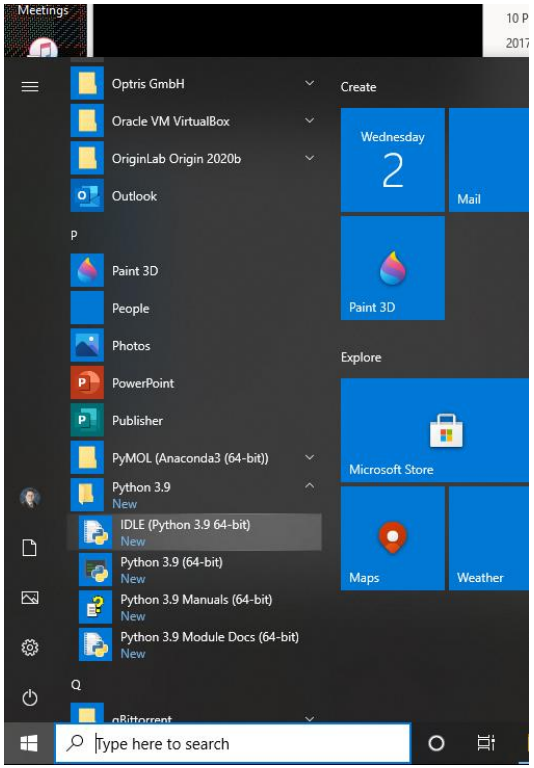
Starting Python on Windows

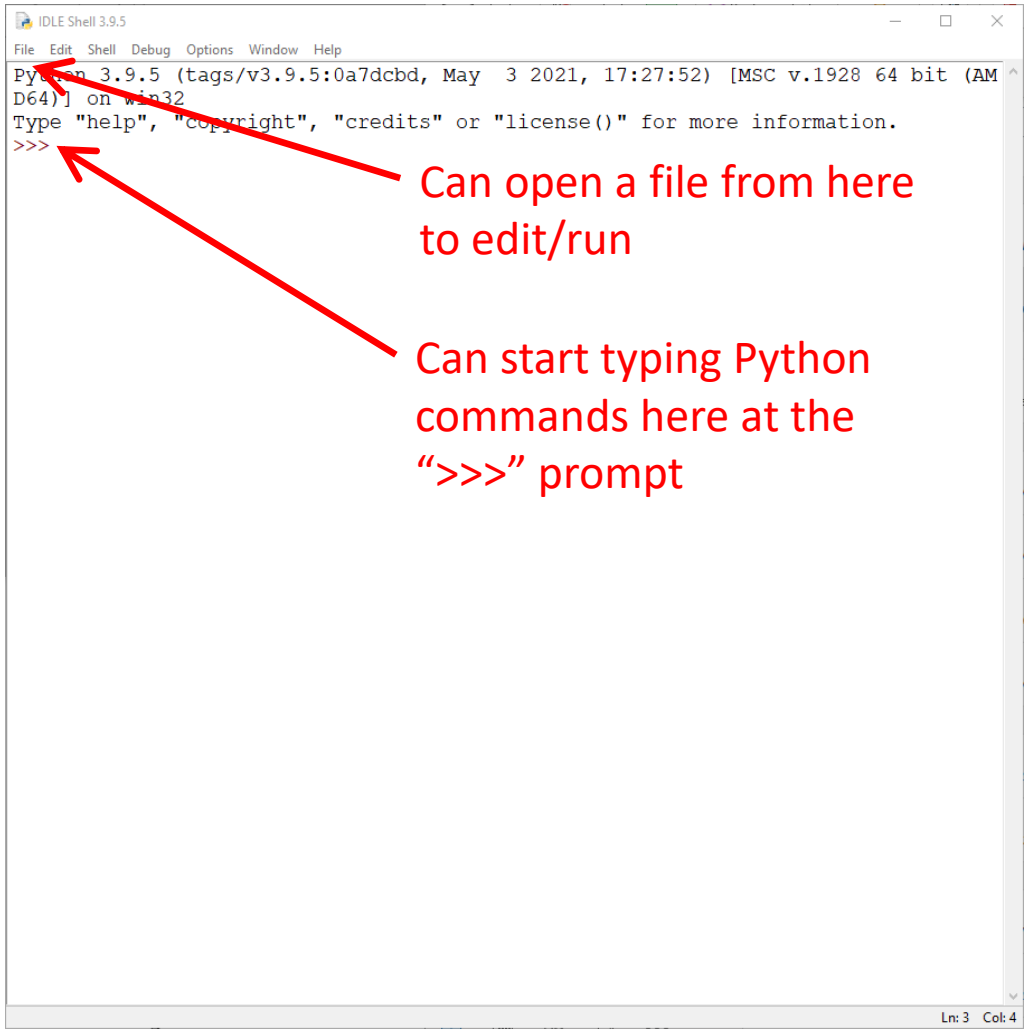




- **Method 1:** Right click on a .py file, then select “edit with IDLE”
  - Can run the file by pressing F5 or shift-F5

Starting Python on Windows





- **Method 2:** Open IDLE from the Start Menu

Python Modules

- The program *pip* is used to add modules & features to Python
  - Extend functionality, re-use code from others

- Key modules:
  - **Numeric Python (numpy, https://numpy.org/):** support for fast matrix/vector calculation
  - **Matplotlib (matplotlib, https://matplotlib.org/):** Create interactive graphs and PDFs from within Python
  - **Scientific Python (scipy, https://scipy.org/):** support for function optimization, numerical integration of differential equations
  - **BioPython (Bio, https://biopython.org/):** Parse PDB files, manipulate DNA/protein sequences

Installing Key Modules

- First, update pip:

```
pip install --upgrade --user pip
```

  - May need to type "`pip3`" instead
  - On Windows, may need to replace "`pip`" with "`python -m pip`" if you receive a warning

- Then, install numpy, scipy, matplotlib, and Bio:

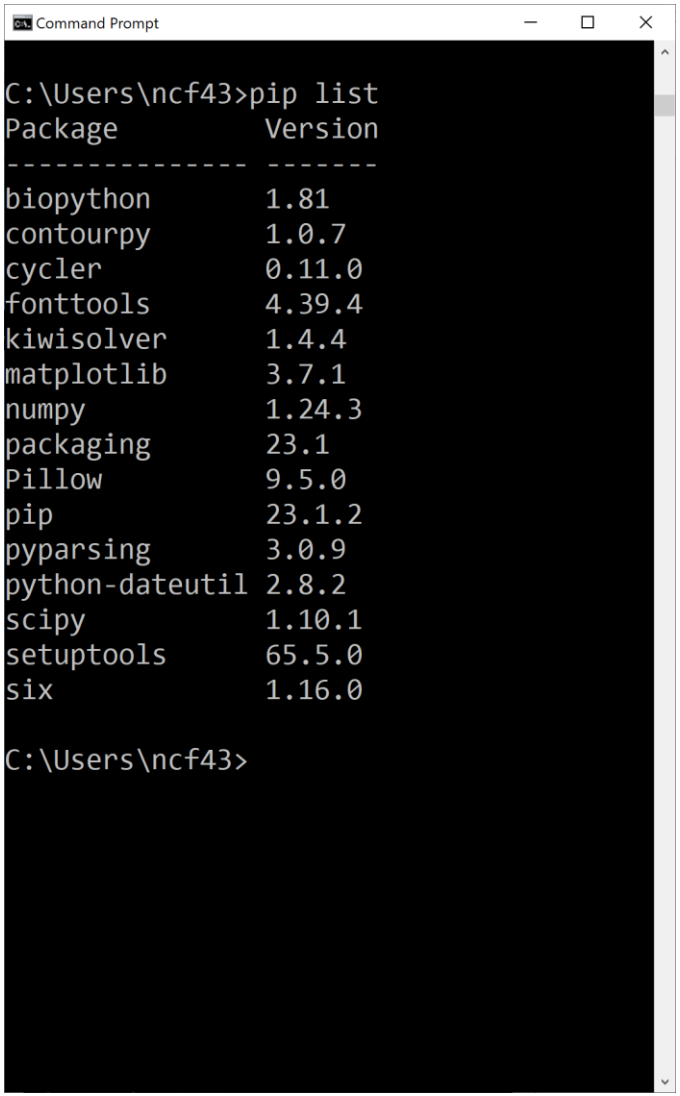
```
pip install --user numpy
pip install --user matplotlib
pip install --user scipy
pip install --user biopython
```

- Some additional packages may be installed; these are updated from web (Internet required!)
  - Can `--upgrade` at any time

Installing Key Modules

- If everything is updated, type "`pip list`" and you should see something like what is shown to the right

- Versions will vary, but notice: biopython, scipy, numpy, and matplotlib



How I Code (Try This First)

- Open a google window; lots of good examples are on the internet (e.g., “how to sort a list python”)
- **Windows:** Open a text file in IDLE, edit the file using IDLE, run using F5
  - Test commands can be run in the Python shell
- **Linux or Mac:** Open a text file in an editor in the background (e.g. Xemacs), run it by repeatedlys aving and typing “`python <file.py>`”
  - Remember to use ampersand (&) when starting the editor!
- Let’s try it both ways!

The Dirty Secret

- I can't teach you how to code in one 75-minute session
  - You must teach yourself or take a class to master coding
  - With experience, you will know syntax and be able to interpret error messages

- Establishing a workflow and having all the tools ("development environment") is the major hurdle for new users

- **Good News:** You now have a development environment and can start playing
  - **Yes, it's play.** Learning starts with play.

Where to Go From Here?

(all of these take *time*; *invest* and *reap dividends*)

- **How to Think Like a Computer Scientist**
  http://openbookproject.net/thinkcs/python/english3e/
  - Focuses on problem solving with Python, very complete list of topics

- **Automate the Boring Stuff Using Python**
  https://automatetheboringstuff.com/
  - Website that focuses on using Python for routine stuff (as scientists often do!)

- **Python Module Index (for Reference)**
  https://docs.python.org/3/py-modindex.html
  - All built-in module documentation, can be very useful for reference, e.g. the `time` and `math` modules contain functions for converting time and performing simple math operations
  - Primarily a last resort, but don't forget that *all* aspects of the core language are documented!

Where to Go From Here?

(all of these take *time*; *invest* and *reap dividends*)

- **NumPy, SciPy, Matplotlib, etc. Websites**
  - These modules may not be documented as well, but they are all documented to a certain extent

- **Internet Forums**
  https://stackoverflow.com/questions/tagged/python and https://www.reddit.com/r/Python/
  - Most people are very friendly and will help if you write a well-phrased question and have done a basic Google search first

- *Success goes to the bold*; if you don't ask questions, you are unlikely to be successful!

A Final Note: Sharing Code

- Many email systems filter/remove .py files from attachments for security

- Rename .py files to .txt (or .py.txt) before sending them as attachments!

The Rest of our Time

- Template for Python
- Basic Python Operations
- **Advanced Examples:**
  - Modify a PDB B-factor Column (BioPython PDB parsing)
  - Protein denaturation fitting script (Numpy, Scipy, Matplotlib)