

Idealistic Python Course

April 30th to May 8th, 2018
Austin Building 2nd floor, room S8
Department of Zoology
University of Cambridge

This will be an introductory course on Python programming, with no assumed programming or other computing knowledge.

In addition to Python, you'll learn something about using the command line effectively, shell scripting, text editing, source code control (via git & github), and code testing. There's no a priori course agenda. We will work on simple relevant real problems, rather than doing abstract programming exercises. So if you have data and/or problems you'd like us to work on, please bring them. There are many other topics we could cover, depending on interests. E.g., SLURM on the HPC cluster, plotting, packages such as BioPython, numpy, & pandas, how the web works, etc. We'll collectively decide.

When non-programmers working with data run into a problem they cannot solve using pre-existing tools (e.g., excel), the reaction is usually: ask someone else for help, do it manually, or give up. In this course we will add another option: write a program. The course aim is concrete and modest: turn you into someone who, when faced with a data problem, thinks "I can write a quick script to solve that problem" (and then does!). Many people go back to their regular lives after taking courses, don't put into practice what they learned, and eventually forget it all. I don't want that to happen with our course. We'll aim to turn everyone into an active programmer. To get you going on a path that fundamentally changes how you work with data for the rest of your career. It's impossible to learn everything about programming in a week, but it is possible to take the crucial first steps and to make them stick. Although I will formally lead the course, it will only be successful if it is a group effort. Attitude is very important. We'll get past the initial learning hump, teach each other (so if you already know a bit more, please come with the intention of helping others catch you up), learn by doing, learn where to find help, and end up with a small group who can continue to encourage and help each other. Hence "idealistic".

Running the course is a big time commitment for me, and I will expect a similar commitment from you :-). So if you sign up, please plan to dedicate as much time as possible to the course. That means spending your days in the class, and preferably spending your evenings and the weekend reading, programming, learning, doing homework tasks, etc. Make sure you have the time to really throw yourself into the course. If you can't commit this time around, don't worry, we can run the course again.

Some notes:

- You'll need to bring a laptop, ideally running some kind of UNIX (i.e., Linux or Mac OS X). It's not easy (and slows the whole class down) to support people using Windows, but it can be done (probably via Cygwin: <https://cygwin.com>). Let me know in advance if you only have Windows.
- The group size will be small. Hopefully not more than 8.
- The course will run over 7 working days: Mon-Fri in the 1st week, then Mon-Tue in the 2nd. The 1st week should be considered mandatory. If you can attend at the start of the 2nd week as well, great. In the days of the second week, we'll emphasize helping each other as a group, working on code yourselves, specific problems, and less general / formal teaching.

Although you won't be expected to know anything ahead of time, the more you do know, the better. So:

- It would be good to familiarize yourself a little with the command line (also known as the terminal, the shell, or bash).
- It would be good to know how to use a text editor. This is for editing plain text files. You cannot use a tool like Microsoft Word for this because it has its own special and obscure file format. I suggest trying Sublime Text (<http://www.sublimetext.com>), though there are many options - any one should be fine. If unsure, ask.
- You could try learning some Python (version 3, preferably) ahead of time. E.g., at <https://www.codecademy.com/learn/python> or by reading, e.g., <http://learnpythonthehardway.org>

These three things are closely tied together in what you'll be doing. You'll write code in the Python language. You'll do that in a text editor. You'll then run those programs on the command line (using bash). You must become comfortable with those three tools. The more you know about them before we start, the better.