

1 SIO 113: Python Programming for Earth Science Students

Authors: Lisa Tauxe, ltauxe@ucsd.edu Hanna Asefaw, hasefaw@ucsd.edu,
& Brendan Cych, bcych@ucsd.edu

Instructor: Lisa Tauxe, ltauxe@ucsd.edu

TAs: Brendan Cych, bcych@ucsd.edu; Shelby Jones, saj012@ucsd.edu

- Spring 2020
- Lectures: MWF 1-1:50
- Discussion Section: W 2-2:50
- Lectures and Discussion sessions will be conducted on Zoom:
<https://ucsd.zoom.us/j/629760853>.

1.1 Computers in Earth Science

Computers are essential to all modern Earth Science research. We use them for compiling and analyzing data, preparing illustrations like maps or data plots, writing manuscripts, and so on. In this class, you will learn to write computer programs with special applications useful to Earth Scientists. We will learn Python, an object-oriented programming language, and use Jupyter notebooks to write our Python programs.

1.2 Python

So, why learn Python? Because it is:

- Flexible, freely available, cross platform
- Easier to learn than many other languages
- It has many numerical, statistical and visualization packages
- It is well supported and has lots of online documentation
- The name ‘Python’ refers to ‘Monty Python’ - not the snake - and many examples in the Python documentation use jokes from the old Monty Python skits. If you have never heard of Monty Python, look it up on youtube; you are in for a treat.

Which Python?

- Python is transitioned from 2.7 to 3. The notebooks in this class, apart from a few exceptions, are compatible with both but they have only been tested on Python 3, so that is what you should be using.
- If you decide to use a personal computer, we recommend that you install the most recent version of Anaconda python for your operating system: <https://www.anaconda.com/download/> you will also need a few extra packages (cartopy, version 0.17.0 and some others) which can be installed with little hassle.

1.3 Grading

- 25% Practice Problems for each lecture
- 25% Final Project
- 50% Weekly Homework assignments

1.4 Class Structure

- There will be three lectures a week and one discussion session. These will be conducted on ZOOM: <https://ucsd.zoom.us/j/629760853>
- Students are expected to read the lecture prior to attending class
- Lecture material will be available on <https://datahub.ucsd.edu/>. Go to this website and log in with your UCSD credentials. Put this string (no carriage return) in your browser window and hit 'return':

```
https://datahub.ucsd.edu/hub/user-redirect/git-pull?repo=https%3A%2F%2Fgithub.com%2Ftaux%2FPython-for-Earth-Science-Students&urlpath=tree%2FPython-for-Earth-Science-Students
```

This will refresh the website and allows you to use the lecture material. The lectures are in a special format - jupyter notebooks. This website loads the lecture notebooks, installs all the required software and lets you read the lecture and complete the assignments.

- When you open a lecture, make a COPY as the lectures may be overwritten if you repeat the start up string again. Save the lecture (we will show you how in Lecture 1).
- Each lecture will begin with a quick review (~5 min) and go over the most common problems encountered by students in their Practice Problems. Then we will proceed to the topic of the day. Lecture time

will be devoted to practicing the skills that we covered in the lecture by completing the assigned Practice Problems for that lecture.

- Each lecture has a “Practice Problems” notebook in the “Practice_Problems” folder. These are worth 1 point each of which 25 will “count”. These should be completed on the DataHub site, downloaded to your computer, then uploaded into Canvas for grading. They are due by the close of business (midnight) on the day of the lecture (except for the first one which is due the day after). Code must be fully commented (we will show you how), must run with no errors, and must be turned in on time to receive credit. There is no partial credit for practice problems.
- Read the lecture PRIOR TO the zoom “lecture” session as these will be devoted to helping students complete the practice problems.
- There will be a programming assignment every Friday, due BEFORE CLASS one week after it is assigned. Assignments will count for 50% of the grade. There are nine of them, so you do the math.
- Help with assignments and the solutions will be discussed during weekly discussion section on Wednesdays after class. The TAs will help you both before they are due and after they have been graded.
- In lieu of a final exam, there will be a final project. This is a program of your own design, and a notebook cell of approximately one page length that contains a description of the project. There is a great deal of flexibility in what the program will do, but there are a few minimum requirements. We will discuss the final project in more detail later. Final projects will count for 25% of the final grade. In addition to the project, students will prepare a 5 minute presentation of their work, to be given either on the last day of class or on Thursday June 11th, during the “final exam” slot 11:30-2:30. These are fun and attendance at least one of the sessions is required. But you should plan on attending both as the projects can be quite interesting.

1.5 Class Expectations

- Attendance is strongly suggested but not required. Submission of the practice problems IS required as are the weekly homework assigne-

ments and the final project. None of these will be accepted late.

- You may consult any online resources (Stackoverflow is a wonderful resource) or any of your fellow students or your instructors to help you solve your problems. Although this is encouraged, do NOT copy what you find verbatim. You must re-work the solutions through your own brain and in your own words and style, otherwise you will not learn how to program. Copying programs does not help you learn and in fact it is “cheating”. Cheating will be reported to the authorities.
- The best way to learn how to program is to do it. This is helped by attending the lectures, completing the practice problems and assignments, and attending the discussion section where your TA can help.

1.6 Final project

As you learn new concepts, start thinking about what you’d like to create for your final project.

There is a great deal of flexibility on the exact nature of the final project but it must be related to Earth and Space Science and, at the bare minimum, it must:

- include at least one module with three functions - read in at least one data file - make at least one plot - use at least three markdown blocks:
- a description of what the program does, - instructions on how to use the program,
- a summary of the scientific conclusions

Turn the project in as a zipped directory with all the parts required to run it (the data files, figures, modules, etc.).

Here are a few ideas:

1) Make a movie:

- of lightning strikes across the continental US
- of volcanic eruptions across the western US
- of plate motion over the last 200 Ma

- your choice
- 2) Make a 2D image of the solar system with orbiting planets
 - 3) Recreate your favorite plot from a number of examples
 - 4) Design your own project

In Week 6, you will be asked to turn in a proposal for the final project. In the proposal, you'll describe the final project and how it relates to Earth and Space Science. At that stage, you still may not have all the skills required to complete your project, but we can let you know if it is possible and substantial enough for the final project.