# SDS 271: Programming for Data Science in Python

Fall 2023

| | |
|---|---|
| Class Hours: | Tu/Th 2:45 PM -4:00 PM |
| Classroom: | Sabin-Reed 220 |
| Text: | Python for Data Analysis, 3rd Edition (online) |
| Instructor: | Casey Berger |
| E-mail: | cberger@smith.edu |
| Group office hours: | M 10-11AM, Tu 1-2PM<br>and by appointment: **https://calendly.com/caseyberger**<br>* zoom or in-person |
| Group office hours location: | McConnell 214 |
| Office: | McConnell 301A |

## Land Acknowledgment

We acknowledge that we are on Indigenous land: the ancestral territory of the Nonotuck peoples. We are grateful for the opportunity to live, learn, and grow on this sacred land and extend our respect to citizens of this nation who live here today and to their ancestors who have lived here for hundreds of generations. We recognize the repeated violations of sovereignty, territory, and water by invaders who have impacted the original inhabitants of this land for over 400 years. We know this acknowledgement is insufficient, and does not undo the harm that has been done and continues to be perpetrated now against Indigenous people and their land and water.

*Science is not neutral*

## Course Description

This course covers the skills and tools needed to process, analyze, and visualize data in Python and work on collaborative projects. Topics include functional and object-oriented programming in Python, data wrangling in Pandas, visualization in Matplotlib in seaborn, as well as creating a reproducible workflow: debugging, testing, and documenting programs and effectively using version control. The major goal for the course is to create a viable, open-source Python package like those in the Python Package Index (PyPI). Prerequisites: SDS 192 and CSC 110, or equivalent.

# Course Logistics

## Objectives

By the end of this course, you will be able to

- write your own functions and packages in Python
- utilize class structures in Python to create reproducible, organized processes for data manupulation
- understand when to use functions and when to use classes
- submit a pull request to GitHub
- create clear and helpful documentation for your code
- use the Pandas library to organize, manage, and manipulate data
- explore a dataset visually with seaborn
- tell a story with data through effective visualizations
- write a complete Python package

## Topics

This course covers the following content:

- Data workflow in Jupyter Notebooks
- Functional programming in Python
- Important packages in Python for data wrangling: Numpy, Pandas, Matplotlib, and seaborn
- Data visualization
- Object-oriented programming in Python
- Working with version control through GitHub
- Writing a complete package that can be used by others

## Course Components and Grading

The course grading will be distributed in the following way:

- Labs : 20%
- Project 1: write a function: 10%
- Project 2: write a data wrangling class 15%
- Navigating GitHub: 5%
- Final Project: 50%

Many of the course components will be graded S/U, but the completed version of your final project and a group self- and peer-assessment will be graded on a numerical scale. Please see below for details.

### Labs and Reflection (20%)

**due: Tuesdays, recurring**
The first ten weeks of class will focus on skill building. In those weeks, there will be 8 labs, which you will begin during class time but must finish in your own time. In addition, there will be a brief reflection form for each lab that is part of the lab. The labs are graded S/U and are graded on completion.

### Project 1: write a function (10%)

**due: 9/19**
The purpose of this project is to demonstrate an understanding of and the ability to apply functional programming skills in Python. You must write an original function, which can do anything you want it to do. Propose a function (one sentence, must be approved by the instructor), draft the function, and turn in a Jupyter notebook which runs the function and includes at least two tests of the function. This project is graded S/U and you will receive detailed code review. See below for the regrading policy.

### Project 2: write a data wrangling class (15%)

**due: 11/7**
The purpose of this project is to use the Python class structure to create a built-in data cleaning and analysis pipeline for a dataset. The class must include internal functions and routines and must be able to process, clean, analyze, and visualize the data used. This project is graded S/U and you will receive detailed code review. See below for the regrading policy.

### GitHub (5%)

**due: 10/31**
As part of Labs 7 and 8, you must create a repository in GitHub and submit a pull request to another repository. This GitHub portion will be graded separately from the lab, as it's critical to

learn to navigate version control systems to write collaborative code. This will help you as you go into your group projects at the end of the semester. This is graded S/U. If you receive a "U" on this and wish to attempt it a second time, set up a meeting with Casey to discuss.

**Final Project (50%)**

The final project will be to write a Python package like those available in the PyPI library. This is a group project and contains multiple components. Some components are graded S/U, but some are graded with a percent grade.

- Group formation and pitch (5% of the course grade), **due: 10/5**. This phase of the project happens early. You will propose some ideas or topics you are interested in working on, and Casey will put you in groups of three to four people based on your responses. Your group must then come up with and submit a one-sentence pitch for your final project. This is graded S/U, but you will receive feedback on your pitch.

- Final project proposal (5% of the course grade), **due: 10/19**. Your group will expand on your one-sentence pitch to write a proposal for your project. This proposal must be two to three pages long and must clearly explain what your package will do and your plan for implementing your idea. This is graded S/U.

- Rough draft (10% of the course grade), **due: 11/16**. Your group must turn in a package through GitHub that contains at least one fully documented function or class. This means a README and a suite of tests for your function. This rough draft is graded S/U.

- Final draft (20% of the course grade), **due: 12/14**. Your group must turn in a complete Python package through GitHub. It must be accessible via an https clone, must contain a README and a tutorial (you may choose how you create that tutorial), and include tests and documentation for how the tests work.

- Self- and peer- assessment (10% of the course grade), **due: 12/14**. You must complete a self-reflection about how you worked as a member of a team and your personal coding development over the semester. You will give yourself a grade for both components. In addition, you must submit an assessment of your teammates' contributions to the team. **If you do not turn in your assessment of your teammates, you will receive a zero for your self-assessment.** Your grade on this part of the assignment will be a weighted average of the scores you received from each member of your team, including from yourself.

You may design this package to do anything you would like, but you must include a testing suite that follows the steps outlined in the PyPI best practices document and detailed documentation for how to use the package.

## Regrading Policy

You will be given the opportunity to retry all projects once except for the final project. This process is described below.

For each project you would like to be considered for regrading, you must

- write up a short explanation of what went wrong the first time, and how your new version differs from your original one
- schedule a 15 minute meeting with Casey to discuss the changes
- submit via email your new version, your old version, and your write-up to your instructor at least 24 hours before your meeting

**This process must be completed within two weeks of your grade being submitted to you. This is subject to the usual extension policies explained below**

## Deadlines and Extensions

You may take an automatic 48-hour extension on any assignment **as long as you send me an email before the deadline telling me you will take the 48 hour extension**.

If you need more than 48 hours, please email me. I may not grant longer extensions – it will depend on the circumstances as well as the type of the project – but I am always willing to consider it.

# Community

## Classroom Culture

We will build this part of the syllabus together, and it will serve as our community expectations. How will we protect and support each other in order to make this classroom environment one where we can all thrive?

- Be respectful of everyone's varied backgrounds (experience level, personal history, physical and mental health challenges)
- Be respectful of different needs in the classroom
  - Please do not wear fragrances to class
  - In group work, be mindful of volume
  - If you need to step out, do so
- Keep each other healthy – don't come to class sick
- Support and help each other – you are in this together
- Establish a culture of communication with each other
- Be kind to yourselves and others. Remember that bringing your best to class might look very different from day to day.

### A general note on emails and availability:

If you email me on a weekday, you can expect a response from me within 24 hours. If you email me on a weekend or holiday, you can expect a response from me by class time on the first day back to class.

### A general note on COVID-19:

If you are feeling at all unwell, please do not come to class. Not only is rest important for your own health, but it's also important to do all you can to protect your community. Stay home, message me to arrange any accommodations (I will be very flexible!), and take a symptomatic test at the self-serve kiosk at the Campus Center. Contact the Schacht Center if you need any assistance with this (413-585-2250).

# College Policies

## Academic Integrity and Honesty

**Honor Code Statement:**
Students and faculty at Smith are part of an academic community defined by its commitment to scholarship, which depends on scrupulous and attentive acknowledgement of all sources of information and honest and respectful use of college resources.

Smith College expects all students to be honest and committed to the principles of academic and intellectual integrity in their preparation and submission of course work and examinations. All submitted work of any kind must be the original work of the student who must cite all the sources used in its preparation.

Students voted to establish the academic honor system in 1944. The basis of the Academic Honor Code is articulated in Article X of the SGA Constitution and Article VII of the SGA Bylaws.

## Accommodations for Disabilities

I **do not** require a letter from ODS to make accommodations for disabilities. If you would like to work through ODS, I am happy to do so, and you can contact them at ods@smith.edu. You may also just let me know what accommodation you need, and we will work to ensure you are properly supported. You do **not** need to tell me your disability.

# Resources

There are tons of resources at Smith to help you succeed in this course and in your college career. If you think of resources I have not included here, please let me know, so I can add them.

- Your instructor! I am here to help you and you are welcome to reach out to me any time.
- Spinelli Center math reviews
- Workshops for time management and managing stress
- Writing help from the Jacobson Center
- Crisis Resources
- Mental Health Crisis Hotline: call or text 988
- Counseling Resources
- Wellness Services
- Gender Identity and Expression
- Where to report sexual misconduct and other forms of discrimination