UNIVERSITY OF PENNSYLVANIA
SCHOOL OF SOCIAL POLICY & PRACTICE
Data Analytics for Social Policy Certificate

MSSP 607: Practical Programming for Data Science

Course No. MSSP-607-001 (online)

Fall 2021

Instructors
Parijat Dube
TA: Caroline Song
TA: Ronil Synghal

Instructor Contact Information
pdube@upenn.edu
xcsong28@gmail.com
ronils@seas.upenn.edu

Course Description/Purpose
This course familiarizes students with no prior programming experience with the core concepts of programming and the practice of software development for data-intensive applications in industry and government. Handling data and being able to meaningfully investigate it and get answers to quantitative questions is a critical skill and programming should be one tool in your toolbox for gaining that ability. Nevertheless, the goal of this course is not necessarily to be ready to take on a data scientist job yourself, but to be able to engage in meaningful conversation with peers, coworkers, and collaborators about their technical work, to be part of a data-oriented software development team, and to accomplish data analysis and file manipulation goals by yourself with fundamental programming skills in your day-to-day work.

Educational Objectives
After this course, students will be comfortable with the following skills:

- Using appropriate data structures like lists, dictionaries, and multidimensional arrays.
- Coding to perform data cleaning tasks and answer statistical questions about datasets.
- Effectively communicating your findings from interactive, exploratory programming.
- Working within technical teams and using software development best practices.

Course Requirements and Expectations

Class Delivery: The class is fully online delivered over zoom sessions. The scheduled time for synchronous class meetings is Tuesday 8:30 AM till 10:00 AM. The first class is on Tuesday August 31, 2021.

Psychological or Disability Needs: In addition to any questions or needs beyond what is listed in the rest of this syllabus, if you have any guidance or assistance with counseling, accommodations for disabilities, or other services, resources are available through Student Disabilities Services and Counseling and Psychological Services. If these resources are not meeting your needs as they relate to this course, please let the instructors know and we can determine what can be done to help you.

Attendance: This course is fully online and we will be following a flexible "flipped" model, where recorded lectures and materials are made available **before** they are covered in synchronous class periods. Our synchronous class meetings will take place each week at the scheduled time for the course, and they will be recorded.

Watching all the material is vital to learning the overall content of the course, but the specific time and place that you choose to do so is up to you. If you are not able to attend a particular synchronous session, that is okay; not using the materials at all, however, is a serious problem, and may be resolved through individual meetings with the instructors or your educational advisor, and could lead to course failure, depending on circumstances.

Participation: When you do attend synchronous sessions, you are expected to be consistently engaged with what's going on, and to ask questions when you are confused. But if things are going on at home or in your apartment, feel free to step away for a few minutes; the recorded videos will still be there when you come back. Please mute your sound and turn off your video if you think there will be an extended interruption.

We may occasionally post check-in polls and/or requests for feedback that will occur throughout the semester, distributed through Canvas. These will not be for credit but will shape the direction of the course, and we hope you take them seriously and help us improve the course delivery by letting us know your opinions.

Assignments: This course will have weekly quizzes, short take-home activities and five technical assignments, each of which gives you a chance to demonstrate new fundamental programming skills. Each assignment will consist of quantitative questions to be answered with data and code, and you will submit both your answers and your source code for how you answered the questions. Every class (starting from the second week) will have a quiz based on the material taught in the previous class. The quiz will happen towards the end of the lecture and students will take it on canvas. There will be 15 short take-home activities each should not take more than 1 hr.

Later in the semester, you will be working on a final project; most students will be expected to work in pairs. First, you will submit a project proposal where you choose a dataset to work on and investigate in-depth with programming. At the end of the semester, you'll submit a final report, which will have both a code and a written component. This project is the last step of the course; there is no final exam.

The project proposal and project report will be graded primarily on your written report: your ideas, your presentation of data, and the way you integrate your technical results from your coding. However, the proposal and report will also be evaluated for professional presentation, and therefore your grade will also reflect how well you accomplished aspects of publishing your work for a broader audience, and can include mechanical issues like readability and formatting. Projects that were completed as part of a previous course, or that you are already working on for another course this semester, should not be proposed or submitted.

If you need assistance with writing, resources are available at the Weingarten Learning Center, Marks Family Writing Center, and SP2 Academic & Writing Support. If you need further assistance on writing or technical presentation, you should contact the instructors directly and explain what additional help you need.

Collaboration: Programming for technical assignments must be completed individually. You must be the one writing your own code. That being said, getting help from online resources like StackOverflow and Kaggle is a normal part of data science, and students are encouraged to talk with one another about the problems if it is helpful. When you submit homework where you reused code that you found online, you must include links to your sources.

Student should not collaborate on a homework except when it is explicitly mentioned. When you work collaboratively on any homework, you should clearly state in your submission who you worked with and how the work was separated. For students who submit similar code without acknowledging collaborative work, all students may be penalized, regardless of who wrote the code first. In general, code is subject to the same academic integrity policies as other work.

Students are expected to work in pairs for the final project. Variations from the pair structure may be allowed -- either individual work, or groups of 3 -- but only with the instructor's permission. The project proposal should clearly state the division of labor and the tasks or work that each student will be working on. Each student must complete programming work as part of the final project; student pairs are not permitted to split work responsibilities between a technical (programming) student and non-technical (report writing) student.

Projects completed in larger groups will be held to a higher standard than individual projects; substantially more work will need to be done, with more results, more programming, and a more thorough report, in order to receive the same grade as peers who worked in pairs.

Academic Integrity

Students are expected to adhere to the University's Code of Academic Integrity, available at https://catalog.upenn.edu/pennbook/code-of-academic-integrity/ Care should be taken to avoid academic integrity violations, including plagiarism, fabrication of information, and multiple submissions (see descriptions below).** Students who engage in any of these actions will be referred to the Office of Student Conduct, which investigates and decides on sanctions in cases of academic dishonesty.

1. Plagiarism: using the ideas, data, or language of another person or source without specific or proper acknowledgment. Example: copying, in part or in its entirety, another person's paper, article, or web-based material and submitting it for an assignment; using someone else's ideas without attribution; not using quotation marks where appropriate; etc.

2. Fabrication: submitting contrived or altered information in any academic exercise. Example: making up data or statistics, citing nonexistent articles, contriving sources, etc.

3. Multiple submissions: submitting, without prior permission, any work submitted to fulfill another academic requirement.

**It is students' responsibility to consult the instructor if they are unsure about whether something constitutes a violation of the Code of Academic Integrity.

Recommended Text

Sweigart, A. (2019). *Automate the Boring Stuff with Python, 2nd Edition: Practical Programming for Total Beginners.* No Starch Press.

Moline, S. (2021). *Hands-On Data Analysis with Pandas, 2nd Edition*. Packt Publishing.

The first book is available in full and for free online in HTML format at https://automatetheboringstuff.com/

Paid versions are available:

- In print or ebook form for $30-$45 from Amazon:
  - https://www.amazon.com/Automate-Boring-Stuff-Python-2nd/dp/1593279922
  - https://www.amazon.com/Hands-Data-Analysis-Pandas-visualization-dp-1800563450/dp/1800563450
- Direct from the publisher:
  - https://nostarch.com/automatestuff2 for $39.95 (print) or $31.95 (ebook).
  - https://www.packtpub.com/product/hands-on-data-analysis-with-pandas-second-edition/9781800563452 for $39.99 (print) or $27.99 (ebook)

Getting Help / Office Hours

We will have a course Piazza for asking questions that can be answered for the whole class to see. You can also address private questions directly to the instructors. Use this link to access Piazza for the course: http://piazza.com/upenn/fall2021/mssp6070012021c

The instructors will do their best to respond promptly by Piazza or by email, if necessary. Depending on your question or concern, the instructors may schedule individual phone or Zoom calls to resolve more complex requests.

All the instructors (the professor and the two TAs) will be holding weekly office hour. The hours will be scheduled in the first week of class based on a student survey of available times. Office hours with each TA will begin in week 2.

Grading

15% Flipped Activities

15% Weekly Quizzes

50% Programming Assignments (5 assignments each 10%)

20% Final Project

Class Schedule

Week 1 (August 31)

- Overview of course topics and goals, grading, online schedule and structure.
- Landscape of data science: Computer Science vs. Statistics, Python vs. R, etc.
- Introduction to Piazza, Stack Overflow, Kaggle, and Canvas
- Programming environments: Google Colab or a local Jupyter Notebook
- **PARTICIPATION ACTIVITY: Basic Python activity on Google Colab.**

Week 2 (September 7): Basics
SOURCE TEXT: *Sweigart* Chapters 1, 4, 6

- Programming fundamentals: Variables and numbers, running Jupyter cells for I/O
- Introduction to functions: String manipulation (startswith(), replace(), etc.)
- Introduction to basic data structures (using lists)
- **PARTICIPATION ACTIVITY: Store values in a list and manipulate the list**

Week 3 (September 14): Structure
SOURCE TEXT: *Sweigart* Chapters 2, 3, and 5

- Introduction to flow control: whitespace, if/else statements, for vs. while loops.
- More advanced data structures: dictionaries, nested data structures
- Writing your own functions (parameters, scope, and return values)
- Writing good comments and documentation of code

- More Python functionality: exception handling, libraries
- **PARTICIPATION ACTIVITY:**
    - **Writing and documenting a function.**
    - **Handling exceptions in a function using try and except block**

ASSIGNMENT 1: Asking questions about data
OUT: September 15, DUE: September 27

Week 4 (September 21): Files and Data
SOURCE TEXT: *Sweigart* Chapters 9, 11, 16

- Opening and closing files, reading and writing to files
- Introduction to structured data files (CSV and JSON file formats)
- Working with CSV and JSON files
- Introduction to debugging and refactoring your code.
- Additional practice with function-writing.
- **PARTICIPATION ACTIVITY: Open a CSV file and modify the contents.**

Week 5 (September 28): Pandas
SOURCE TEXT: *Molin* Chapter 2

- Introduction to Pandas - history, context, and comparison to alternatives
- Introduction to multi-dimensional arrays, Series, Index, DataFrame
- Pandas DataFrame: common attributes and functions
- Joining, merging, and filtering tables; renaming and reordering columns
- **PARTICIPATION ACTIVITY: Merge two Pandas dataframes together.**

ASSIGNMENT 2: Using Pandas for data analysis
OUT: September 29, DUE: October 11

Week 6 (October 5): Data Wrangling and Aggregation
SOURCE TEXT: *Molin* Chapter 3 and 4

- Data cleaning: string replacement and variable type conversion
- Data reshaping: transposing, pivoting, melting
- Introduction to lambda functions and .apply() functions in Pandas
- Dealing with messy data - dropping rows and columns, filling in missing values
- Restructuring Pandas dataframes with .groupby()
- **PARTICIPATION ACTIVITY: Write a lambda function for a dataframe.**

Week 7 (October 12): Visualization
SOURCE TEXT: *Molin* Chapter 5 and 6

- Simple data visualization: Bar charts, line charts, and scatter plots with Matplotlib
- More advanced data visualization: Using seaborn for clean presentable visualizations
- Visualizing grouped data, side-by-side bar charts, and combined plots
- Axis labeling, scaling, legends, and exporting figures to image files
- **PARTICIPATION ACTIVITY: Create a bar chart using dataframe and export to a PDF.**

ASSIGNMENT 3: Visualizing a dataset
OUT: October 13, DUE: October 25

Week 8 (October 19): Time Series Data
SOURCE TEXT: *Molin* Chapter 4

- Working with time series data and the datetime objects
- Visualizing time series data and smoothed trendlines
- Asking questions about time series data
- Projecting trends into the future and visualizing forecasts
- **PARTICIPATION ACTIVITY: Plot a trendline over a time series dataset.**

Week 9 (October 26): **Web Data**
SOURCE TEXT: *Sweigart* Chapter 12

- Working with data only available as webpages; introduction to API calls.
- Introduction to curl for fetching HTML
- Navigating HTML files with BeautifulSoup
- **PARTICIPATION ACTIVITY: Fetch a website's content and extract text from it.**

ASSIGNMENT 4: Data Analysis from HTML Sources
OUT: October 27, DUE: November 8

Week 10 (November 2): More Automation
SOURCE TEXT: *Sweigart* Chapter 7, 10, and 13

- Pattern matching using regular expressions
- Operations involving files and folders
- Reading and Writing Excel spreadsheets
- Transforming spreadsheets: adding formula and charts, adjusting rows and columns,

Week 11 (November 9): Financial Data Analysis
SOURCE TEXT: *Molin* Chapter 7

- Python for retrieving, analyzing, and comparing stock prices
- Common financial metrics for a single asset and a group of assets
- Modeling financial performance using historical data and stock price prediction
- **PARTICIPATION ACTIVITY: Retrieve stock prices from Yahoo! Finance**

ASSIGNMENT 5: Financial data analysis with web APIs
OUT: November 10, DUE: November 22

Week 12 (November 16): Project Guidance

- Guidelines for final project and options for final presentation format
- Example projects walkthrough
- Basics of Github and its common commands
- Documenting your code for external readers
- Avoiding common errors in public communication of research

- **PARTICIPATION ACTIVITY: Create a Github Page for your final project d make your first commit**

Project Proposal
OUT: November 16, DUE: November 30

November 23: Thanksgiving week, no class

Week 13 (November 30): Location Data

- Working with geospatial data: Introduction to GPS coordinates
- Common questions you should ask about spatial data
- Introduction to GeoPandas and GeoDataFrame
- Visualizing spatial data and interacting with maps.
- Interaction of spatial data and time series information
- **PARTICIPATION ACTIVITY: Plotting a GPS coordinate dataset on a graph.**

Week 14 (December 7): Statistics

- Visualizing uncertainty: box-and-whisker plots, error bars, time series with uncertainty
- Statistical hypothesis testing: sign tests, t-tests, chi-square tests
- Setting significance targets and interpreting p-values
- Avoiding common errors in statistical significance testing
- **PARTICIPATION ACTIVITY: Measure a p-value from a t-test on a dataset.**

Final Project DUE: Dec 15