

ENVR 890-001: Python for Environmental Research (1 credit)

Department of Environmental Sciences & Engineering

Gillings School of Global Public Health

University of North Carolina at Chapel Hill

Fall 2020 Information

Instructors: Andrew Hamilton (andrew.hamilton@unc.edu), Dr. Greg Characklis (charack@email.unc.edu)

Class Room: Virtual (Check your email for the Zoom link)

Class Time: Fridays, 10:40-11:55 AM, Eastern time

Office Hours: Wednesdays, 3-5 PM, Eastern time (Check your email for Zoom link)

Course Website: <https://github.com/ENVR-890-001-F20/ENVR-890-001-Python-For-Environmental-Researcher-F20> (Note: *you will need to create an account on GitHub.com and email Andrew your username so that I can grant you access.*)

Prerequisites: None (although ENVR 755 is a suggested co-requisite, see “Course Objectives”)

Text: None. We will draw from a variety of free, online resources.

Software and Hardware: All software used for this course is freely available (e.g., Python, Jupyter Notebooks, GitHub). We will walk through the installations in the first week of class. Due to the nature of this course, all students will need access to a computer on which they can install and run software. However, we recognize that this may not be possible for all students, especially in light of the circumstances surrounding COVID-19, and want to be as inclusive as possible. *If you do not have access to a computer, or if you are unsure whether your device can be used for programming (e.g., a tablet), please email us so we can work to find a solution.*

Syllabus Changes: This syllabus (especially the schedule) is subject to change. Students should refer to the version on the course website, which will be kept up to date.

Course Motivation

Computer programming is an increasingly important tool for researchers interested in environmental science, engineering, and public health. Recent examples from UNC include [modeling the effect of drought on electricity production](#), [studying the factors associated with extreme hurricane storm surge](#), [estimating air quality co-benefits of climate change mitigation](#), [mapping the global extent of rivers using satellite imagery](#), [estimating the premature mortality associated with traffic-related air pollution](#), and analyzing data to look for relationships between [race and community water/sewer service](#) or between [fecal indicator bacteria and illness in swimmers](#). The ability to formulate research questions, build models, analyze data, and visualize results using computer programming is a valuable toolkit for researchers in academia, government, NGOs, and industry.

Course Objectives

This course will serve as an introduction to computer programming in Python. Students will develop a working knowledge of the Python programming language and learn how to formulate research questions as computer code. They will learn about a variety of Python-based packages and techniques that can be used to build models, analyze data, and create visualizations for research in environmental sciences and engineering.

ENVR 890-001 is a companion course to ENVR 755 (Analysis of Water Resource Systems), and as such, an emphasis in the second half of the semester will be placed on applications related to water resources engineering, economics, and management. Students are encouraged to co-enroll in ENVR 755 for context on these applications. However, all are welcome, and many of the tools and techniques learned in this class will be applicable to a variety of research areas beyond water resources management.

Expectations & Approach

This class is intended for students with little or no experience with programming, so we will start at the beginning. However, there will be a fairly steep learning curve, and students will be expected to spend time outside of class practicing and developing programming scripts to complete assignments. Learning a programming language is much like learning any other language - practice and repetition are key.

There will be short homework assignments approximately every other week, meant to gauge understanding and reinforce techniques learned in class. In-class time will be split between lectures, individual exercises, and group work. Students are encouraged to actively participate in class discussions and exercises, and ask questions. Computer programming concepts can take a while to sink in, and if you are confused, chances are high that others are too!

Grading will be based on quality of assigned work (50%) and in-class participation (50%).

Honor Policy

Students are encouraged to communicate and help one another with the assignments. However, each student is responsible for completing their own work, and plagiarism will not be tolerated. Students are also encouraged to consult online resources (Google, Stack Exchange/Overflow, programming blogs, etc.) - learning to ask the right questions and find answers on the internet is a key skill for computer programmers. However, if you copy any code directly from the internet, please cite your source.

For more details, see the [UNC Honor System](#), the [UNC Libraries' plagiarism tutorial](#), or just ask us directly about expectations for the class.

Diversity, Equity, and Inclusion

Diversity of backgrounds and perspectives is vital as we strive for equitable institutions (race, ethnicity, gender identity, sexuality, age, ability, socioeconomic status, etc.). The diversity of researchers in academia, government, NGOs, and industry affects the voices we consider, the

questions we ask, and the answers we find. [Environmental science](#), [computer science](#), and STEM more broadly, have a long way to go on this front. However, changes in recruiting, pedagogy, and work environment can help to encourage students of diverse backgrounds to pursue fields such as [environmental science](#) and [computer science](#). It is our intention to create a (virtual) classroom in which all students feel respected, valued, and encouraged to apply the material to their own interests. Please let us know of any ways to make the course more welcoming and effective for you personally or for other students or student groups.

Schedule (subject to change)

Date	Lecture	Lecture Topic	Homework Assigned	Due
Aug. 14	1	Intro to Jupyter Notebooks, Python, & GitHub	(1) Hello world	–
Aug. 21	2	Variable types & basic data structures	(2) Basic structures	1
Aug. 28	3	Conditions, loops, & functions		2
Sep. 4	3, 4	Advanced data structures & logical indexing	(3) Conditions, loops, & functions	–
Sep. 11	4	More advanced data structures	(4) Advanced structures	3
Sep. 18	5	Solving systems of equations & Optimization	(5) Solving & Optimization	–
Sep. 25	4	More advanced data structures	–	5
Oct. 2	6	Visualization	–	4
Oct. 9	7	Environmental data sources, FAIR data	(6) Data & regression & viz	–
Oct. 16	8	Regression	–	–
Oct. 23	9	Reservoir simulation & Monte Carlo	–	6
Oct. 30	10	Reservoir simulation & time series	(7) Reservoir simulation	–
Nov. 6	10	Reservoir simulation & time series	–	–
Nov. 13	13	Intro to GIS	–	–
Nov. 23 (noon)	–	–	–	7