INTRODUCTION TO PROGRAMMING

Section	BST 273, Fall I, 2018
When	Tuesday, Thursday (TR) 11:30am-1:00pm
Where	HSPH FXB G13
Website	https://canvas.harvard.edu/courses/53900
Credits	2.5
Version	2018-09-07

Instructors

Dr. Eric Franzosa
Research Scientist, Biostatistics Department, Harvard Chan School
Email: franzosa@hsph.harvard.edu
Office Hour: Friday 11-12pm
Office Hour Location: SPH2 434

Dr. Kevin Bonham
Postdoctoral Fellow, Biostatistics Department, Harvard Chan School
Email: kbonham@broadinstitute.org
Office Hour: Friday 1-2pm
Office Hour Location: SPH2 434

Teaching Assistants

Marina Cheng
M.S. candidate in Computational Biology and Quantitative Genetics, Harvard Chan School
Email: mcheng@hsph.harvard.edu
Office Hour: Wednesday 3-4pm
Office Hour Location: SPH2 428

Shirley Liao
Ph.D. candidate, Biostatistics Department, Harvard Chan School
Email: shirleyxliao@g.harvard.edu
Office Hour: Thursday 1-2pm
Office Hour Location: FXB G03 by default; Kresge 502 on Sept. 20 and Sept. 27

Emma Thomas
Ph.D. candidate, Biostatistics Department, Harvard Chan School
Email: emmathomas@g.harvard.edu
Office Hour: (*Joint with Shirley, please see above*)

Contacting Course Staff

Please contact course staff at the email addresses above or through the Canvas website. If using email, please include “BST 273” in the subject line.

Course Description

This is an introductory course on computer programming in Python (specifically Python 3). Students will begin by exploring the fundamental elements of a computer program (e.g. variables, data objects, functions/methods, and flow of control) along with their associated Python syntax. In the second part of the course, students will gain familiarity with modules in the Python Standard Library for reading and writing data, interacting with external programs, and writing command-line interfaces. The final part of the course will introduce special topics, including scientific computing and Object-Oriented Design. Interspersed with core programming content, students will be introduced to important related programming skills, including navigating the command line, testing and debugging code, and version control.

The course consists of two 90-minute meetings per week, divided into lecture/discussion sections and hands-on activities. Students will complete weekly assignments to reinforce their new programming skills, starting with basic tests of new syntax and leading toward the completion of functional Python scripts. Each student will complete a final project (of their own design, or chosen from a list of candidates) that involves developing, testing, and documenting a Python script to solve a problem in data analysis.

At the end of the course, students will be able to:

- Design, write, and test Python scripts to solve problems in data analysis.
- Identify and invoke appropriate (existing) software modules to aid in script design.
- Share and borrow code using a version control system.
- Enroll in more advanced courses requiring a programming background.
- Learn additional programming languages through self-guided study.

Course Topics and Schedule

Week	Date	Day	Unit	Lecture topic
0	09/04/2018	T	Fundamentals	Orientation
0	09/06/2018	R	Skills	Working on the command line
1	09/11/2018	T	Fundamentals	Variables, scalar data types and methods
1	09/13/2018	R	Fundamentals	Collection data types and iteration
2	09/18/2018	T	Fundamentals	Conditional logic and flow of control
2	09/20/2018	R	Fundamentals	Working with modules, examples with file I/O
3	09/25/2018	T	Fundamentals	Writing functions, references vs. data
3	09/27/2018	R	Fundamentals	Making an executable script
4	10/02/2018	T	Skills	Version control and intro final projects
4	10/04/2018	R	Skills	Testing, debugging, getting online help
5	10/09/2018	T	Special topics	Interacting with external programs
5	10/11/2018	R	Special topics	Regular expressions
6	10/16/2018	T	Special topics	Scientific computing with Python
6	10/18/2018	R	Special topics	Object-oriented Python
7	10/23/2018	T	Special topics	Parallelism and workflows in Python
7	10/25/2018	R	Special topics	Next steps for developing as a programmer

Prerequisites

This course has no prerequisites and is suitable for students with no programming experience. The course will also be beneficial for students whose exposure to programming has been informal or limited to numerical computing environments (such as R or MATLAB).

Students are expected to bring a laptop computer to class to participate in in-class programming activities. Students who do not have access to a laptop should contact the instructors to make alternative arrangements.

Textbooks

- The **required** textbook for this course is Think Python 2e by Allen B. Downey (Green Tea Press). This textbook is available in its entirety online (as HTML or a PDF copy):

  https://greenteapress.com/wp/think-python-2e/

  Make sure you are looking at the second edition (which is updated to Python 3). The website also provides a link to purchase a hard copy of the book (*not required*).

  Author: Allen B. Downey
  Title: Think Python: How to Think Like a Computer Scientist 2nd Edition
  Publisher: O'Reilly Media; 2 edition (December 28, 2015)
  ISBN-10: 1491939362
  ISBN-13: 978-1491939369

- The instructors also **recommend** Get Programming: Learn to code with Python by Ana Bell as an alternative introduction to programming in Python.

  Author: Ana Bell
  Title: Get Programming: Learn to Code with Python
  Publisher: Manning Publications; 1 edition (April 19, 2018)
  ISBN-10: 1617293784
  ISBN-13: 978-1617293788

- As they become more experiences with Python, students will also complete reading assignments from the official Python documentation:

  https://docs.python.org/3/

Course Structure and Assessments

The final grade for this course will be based on:

- Five homework assignments (13% × 5 = 65%)
- Final project (25%)
- Participation (10%)

Five homework assignments (13% × 5 = 65%)

After the first week, but excluding the final project weeks, students will complete weekly homework assignments (five in total). Homework will be assigned on Monday and will be due the following Friday (11:59pm) via Canvas hand-in. Homework assignments are designed to teach Python syntax and encourage computational thinking around algorithmic challenges. Points will be scored based on the syntactic and semantic correctness of the submitted results, with students expected to match pre-set input, output, and behavioral specifications for written code. Especially early on, assignments will also include a small number of written answers to assess conceptual understanding.

Final project (25%)

Students will work on final projects during the final two weeks of the course. Final projects will involve an open-ended implementation of a Python script to solve a problem in data analysis. Students will be able to choose from a number of suggested projects or propose their own. Project ideas will be advertised during the third-to-last week of the course to give students a chance to discuss ideas with the instructors before starting work. (Students proposing their own idea MUST seek and receive instructor approval before proceeding.) In addition to working code, students will turn in sample input and output data along with documentation in the form of a README file (as a single Canvas hand-in). Projects will be graded based on the correctness of the submitted code and completeness of the accompanying documentation.

Participation (10%)

Students are expected to attend and participate in lecture and in-class activities. Participation includes physical presence in class, asking and answering questions, sharing viewpoints in constructive and respectful ways, working diligently on in-class assignments, and otherwise actively engaging with other students and the course instructors.

Attendance will be tracked by the completion of in-class polls on the Canvas site during lecture meetings. *Please notify the instructors of an anticipated absence before class*.

As a guideline, full participation credit (10%) will be awarded to students with good attendance (0-1 unexplained absences) and a strong record of in-class participation. Moderate participation credit (5-9%) will be awarded to students with 2-3 unexplained absences and/or mixed in-class participation (e.g. not participating in in-class activities or rarely contributing to discussions). Low participation credit (0-4%) will be awarded to students with mixed in-class participation (as described above) and 4+ unexplained absences.

Collaboration Policy

We expect students to complete their out-of-class assignments *individually*. While it is tempting to compare code with other students to check correctness and/or move past a stumbling block, within the context of homework assignments and the final project, this is strictly not permitted. Deviation from this policy will be considered a violation of the school's Academic Integrity policy (see below) and treated accordingly.

Naturally, we do not want this policy to be a hindrance to your success in the course. To help with checking code correctness, assignments will generally be bundled with expected inputs and outputs for testing. To help with stumbling blocks, students are encouraged to reach out to

the instructors and/or TAs during the week (preferably during scheduled office hours). Searching for help online (e.g. “what does ‘ZeroDivisionError’ mean?”) is also allowed and encouraged. However, posing specific homework questions online (e.g. via StackExchange) is not permitted, except where specifically requested during the course.

Outside of comparing assignment code, students are allowed and encouraged to help each other in and outside of class. This includes working together or comparing code during in-class programming assignments. In addition, discussing general concepts (e.g. “could you explain the difference between a list and a set?”) is also allowed, as long as it does not involve specific review of assignment code. (As a general rule, if the people in the discussion are not looking at the current assignment, then you are probably not reviewing assignment code.)

Late Work Policy

The maximum score for late work will fall rapidly: 90% if one day late, 75% if two days late, 50% if three days late, and all credit lost if four or more days late. Extensions may be granted *if requested with reason at least 24 hours in advance of the assignment deadline*.

Final Grades

Final letter grades will be curved based on the percentiles of total scores received by students in the class.

Harvard Chan Policies and Expectations

Inclusivity Statement

Diversity and inclusiveness are fundamental to public health education and practice. We encourage students to have an open mind and respect differences of all kinds. We share responsibility with you for creating a learning climate that is hospitable to all perspectives and cultures; please contact us if you have any concerns or suggestions.

Bias Related Incident Reporting

The Harvard Chan School believes all members of our community should be able to study and work in an environment where they feel safe and respected. As a mechanism to promote an inclusive community, we have created an anonymous bias-related incident reporting system. If you have experienced bias, please submit a report here so that the administration can track and address concerns as they arise and to better support members of the Harvard Chan community.

Title IX

The following policies apply to all Harvard University students, faculty, staff, appointees, or third parties (see links to PDF documents and/or web sites):

- Harvard University Sexual and Gender-Based Harassment Policy
- Procedures for Complaints Against a Faculty Member
- Procedures for Complaints Against Non-Faculty Academic Appointees

Academic Integrity

Each student in this course is expected to abide by the Harvard University and the Harvard. T.H. Chan School of Public Health Codes of Academic Integrity. All work submitted to meet course requirements is expected to be a student's own work. In the preparation of work submitted to meet course requirements, students should always take great care to distinguish their own ideas and knowledge from information derived from sources.

Students must assume that collaboration in the completion of assignments is prohibited unless explicitly specified. Students must acknowledge any collaboration and its extent in all submitted work. This requirement applies to collaboration on editing as well as collaboration on substance.

Should academic misconduct occur, the student(s) may be subject to disciplinary action as outlined in the Student Handbook. See the Student Handbook for additional policies related to academic integrity and disciplinary actions.

Accommodations for Students with Disabilities

Harvard University provides academic accommodations to students with disabilities. Any requests for academic accommodations should ideally be made before the first week of the semester, except for unusual circumstances, so arrangements can be made. Students must register with the Local Disability Coordinator in the Office for Student Affairs to verify their eligibility for appropriate accommodations. Contact Colleen Cronin ccronin@hsph.harvard.edu in all cases, including temporary disabilities.

Religious Holidays, Absence Due to

According to Chapter 151c, Section 2B, of the General Laws of Massachusetts, any student in an educational or vocational training institution, other than a religious or denominational training institution, who is unable, because of his or her religious beliefs, to attend classes or to participate in any examination, study, or work requirement on a particular day shall be excused from any such examination or requirement which he or she may have missed because of such absence on any particular day, provided that such makeup examination or work shall not create an unreasonable burden upon the School. See the Student Handbook for more information.

Grade of Absence from Examination

A student who cannot attend a regularly scheduled examination must request permission for an alternate examination from the instructor in advance of the examination. See the Student Handbook for more information.

Final Examination Policy

No student should be required to take more than two examinations during any one day of finals week. Students who have more than two examinations scheduled during a particular day during the final examination period may take their class schedules to the director for student affairs for assistance in arranging for an alternate time for all exams in excess of two. Please refer to the Student Handbook for the policy.

Course Evaluations

Constructive feedback from students is a valuable resource for improving teaching. The feedback should be specific, focused and respectful. It should also address aspects of the course and teaching that are positive as well as those which need improvement. Completion of the evaluation is a requirement for each course. Your grade will not be available until you submit the evaluation. In addition, registration for future terms will be blocked until you have completed evaluations for courses in prior terms.