Code Toolkit: Python

LCST 2790, Fall 2020 (CRN 5525)

Rory Solomon, solomonr@newschool.edu

Wednesday, 12:10 – 2:50am

(Please note: Our class Zoom sessions will always be open 10 minutes before our start time if you wish to arrive early and chat with your classmates or me, as we would do in a physical classroom.)

Office hours: After class for 20 minutes, and by appointment – email me to schedule!

Course Description

This course provides a basic introduction to coding for students with no prior experience using the Python computer language. It will introduce the use of computing and algorithms to web design, data analysis and visualization, and game design. Students will complete integrative projects within each of these areas, and leave the class with a strong foundation in the use of Python across a range of applications. Access to a working laptop will be expected throughout the semester. This course fulfills the Integrative Course requirement of the Culture and Media major.

Class Website & Email

Rather than using Canvas this semester we will be using this website:
http://codeatlang.webfactional.com/code-toolkit/2020-fall/
Please bookmark that URL and refer back to it often. All deadlines, reading materials, project tutorials, and other course materials will be posted here. You will be using Google Docs, Google Drive, and other platforms such as GitHub to submit your project work and reading responses. The website will include links to these resources and instructions on how to submit work.

Please check this site as well as your @newschool.edu email regularly for any course announcements, possibly including last minute updates.

Course Structure

Our course will be organized as a mix of synchronous and asynchronous meetings and materials. I plan to begin our classes each Wednesday with a synchronous Zoom session, which I will record for anyone to view later. I will configure Zoom to provide closed captions (subtitles) with the recording (not live – if this is an issue for you, please let me know.) I will upload these recordings along with the searchable text transcript to our class website.

In general, I plan to begin class by reviewing the previous week's homework, showing examples of student work and addressing any questions that came up. Then I will offer an interactive coding lecture on new topics.

About halfway through, we should take a stretch break, and I plan to use the rest of the class for asynchronous work. I will remain on Zoom and be available for help and questions. I will create breakout groups of 3-5 people, and I would like you, if possible, to try to do some coding work while on Zoom with others. This way you can ask each other questions about bugs. Each week, at least one person should share their screen for some time & show what they're working on. I will visit these breakout groups for help.

For many weeks I will also provide a short video that will supplement our synchronous lecture with an explanation of an additional topic.

Course Requirements & Assignments

You will have weekly homework assignment comprised of coding exercises. You will submit these to our shared Google Drive.

We also have readings about every two weeks. These will be posted in the course schedule. You will be asked to submit short notes as reading responses as part of your homework.

Homeworks and reading responses will be graded as pass/fail, on a scale of 0 or 1, based on effort. You will not be evaluated weekly on whether your code runs perfectly or has bugs, but rather on whether you engaged the topic and made progress. Occasionally I may give a fraction of a point (e.g. .85) to indicate that the level of completeness is not quite enough. I will try my best to keep on posting homework grades throughout the semester, but if I fall behind, at the time of the midterm and final I will get caught up so you know where you stand. If at any time you would like more feedback - about one piece of work specifically or about your standing in the class overall - please do not hesitate to ask.

In addition, there will be two projects to complete: one for the midterm and one for the final. Each project will begin with a 1 page planning document, and will conclude with an accompanying 1-2 page report explaining your work, your intentions, challenges you faced, diagrams and planning material, and explanations of your code. The midterm project will offer you a choice between an interactive non-linear narrative, and a data visualization. The final project will use web protocols (but not HTML pages) to create a networked interactive interface. More specific details about the project assignments will be provided later in the semester.

Projects will be graded based on how well they fulfill the formal requirements stated in the project assignment, and the degree to which they engage with theoretical concepts from readings and discussion. In other words, while the reading discussions are a way to process, digest, and comprehend the readings, the projects are a way to demonstrate an understanding of those readings, and will be graded as such. We won't have any essays or written assignments in this class, in place of this, I would like you to use project work be a product of your engagement with course concepts both technical and theoretical.

Homeworks, and the midterm and final projects will be worked on individually. Your participation grade is based on class discussions, contributions to student work sharing, and questions via office/lab hours and email.

Final Grade Calculation

Weekly homework assignments	30%
Readings and reading discussions	15%
Midterm project	20%
Final project	20%
Participation:	15%

Learning Outcomes

Through this course, students should be able to:

1. think critically about the ways that they and others interact with computation including understanding its socio-cultural and political limits from a hands-on perspective;

2. engage in project-based learning that utilizes computational, algorithmic, and infrastructural thinking;

3. gain a broader understanding of the historical and social factors leading to the increasing presence of computational systems in our lives;

4. work through the social and political implications embedded within computational technologies and develop an accompanying critical / ethical framework;

5. use computation as a tool to enhance their liberal arts education—to better analyze, communicate, create and learn.

6. appreciate the challenges of equity and access posed by increased reliance on computational technologies as well as their potential to reinforce existing inequalities in society; and

Course readings & materials

Lecture notes: I will post weekly lecture notes that give you a recap of all that we talk about in class that week, usually including code examples. These are meant to be very helpful when working on the homework for that week, as homework exercises will directly build on topics discussed in the lesson.

All readings will be provided via the course website. There are no required textbooks to purchase for this course.

Links to other tools, platforms, references, and other online documentation will be posted to the class website.

In addition to these texts, you will need access to a laptop for the duration of the course to complete the projects.

Attendance and Meaningful Participation for an Online Course

Attendance and participation are required. Failing to show up and take part may result in a lowered grade, and excessive lack of engagement may result in being dropped from the course. Of course, we are in strange and challenging times! What is meant by attendance and participation must be rethought during this unusual semester. Participating in a synchronous Zoom session clearly counts as being present. But I realize that due to internet access or other limitations, you may be unable to attend all synchronous sessions. Regardless of whether you can participate synchronously, you are expected to meaningfully participate in the class for 2 hours, 40 minutes each week, plus homework.

Please keep in mind that you are responsible for any assignment deadlines even in the event of an absence. Please check with your peers, the class website, and me regarding anything you miss due to absence. If you miss a Zoom session, you can watch these later when you are able, and I ask you to please by include a few notes in response to the recorded synchronous session as comments in your homework for that week – this will count as your participation.

Please note that you are ***not*** required to turn your camera on during synchronous Zoom sessions. However, a silent gray box for the duration of class may count as a lowered participation grade or even an absence.

I warmly recognize that not everyone feels as comfortable vocally participating in class situations. I implore you to keep in mind that your position and perspectives are unique and valuable, and that the class and I will always benefit from your voice and contributions. If you are unable to participate in a Zoom session, please compensate by adding some reflections on the discussion to your reading response for that week.

Resources

The university provides many resources to help students achieve academic and artistic excellence. These resources include:

- University Libraries: http://library.newschool.edu

- University Learning Center: http://www.newschool.edu/learning-center

- University Disabilities Service: www.newschool.edu/student-disability-services/
  I am more than happy to ensure any required accommodations for any students with accessibility issues. Any student who has needs or concerns about academic accommodations is welcome to meet with me privately. All conversations will be kept confidential. Students requesting any accommodations will also need to contact Student Disability Service (SDS). SDS will conduct an intake and, if appropriate, the Director will provide an academic accommodation notification letter for you to bring to me. At that point, I will review the letter with you and discuss these accommodations in relation to this course.

- Student Ombuds: https://www.newschool.edu/student-advocacy/conflict-resolution/
  The Student Ombuds office provides students assistance in resolving conflicts, disputes or complaints on an informal basis. This office is independent, neutral, and confidential.

- Office of Financial Aid: https://www.newschool.edu/financial-aid/
  During this online semester, financial aid staff remain available by email, phone, and Google Hangout Monday through Friday from 9:00 a.m. to 5:00 p.m. You can also make a virtual appointment with staff via Starfish.

  Eligible students may be considered for different types of financial aid such as scholarships, federal grants, federal work study and federal student loans. To be considered for federal student aid, you will need to complete the Free Application for Federal Student Aid (FAFSA) available online at fafsa.gov. The FAFSA is available starting October 1. The New School's priority deadline to submit the FAFSA is February 15. The FAFSA needs to be submitted annually.

University and College Policies

Academic honesty, plagiarism … and computer programming and open source software
Compromising your academic integrity may lead to serious consequences, including (but not limited to) one or more of the following: failure of the assignment, failure of the course, academic warning, disciplinary probation, suspension from the university, or dismissal from the university.

Students are responsible for understanding the University's policy on academic honesty and integrity and must make use of proper citations of sources for writing papers, creating, presenting, and performing their work, taking examinations, and doing research. It is the responsibility of students to learn the procedures specific to their discipline for correctly and appropriately differentiating their own work from that of others. The full text of the policy, including adjudication procedures, is found at http://www.newschool.edu/policies/

Resources regarding what plagiarism is and how to avoid it can be found on the Learning Center's website: http://www.newschool.edu/university-learning-center/avoiding-plagiarism.pdf

All of that said, software development work and coding almost always involve the appropriation and collaging of other people's code. Learning computer programming often entails modifying working examples rather than starting from scratch. In this class we are participating in communities of shared practices. However, any work you borrow and/or modify *must* be labeled as such. If you find sample code

and integrate it into your work, this must be clearly and obviously indicated as such. Use comments for this purpose (i.e. `# like this` or `""" or like this """`). When attributing code from others in work that you're submitting, clearly indicate the name of the author or a source URL, and make clear which lines of code are not yours.

You will not be graded down for integrating other people's work into your own. In fact, integrating the code of others is often more challenging than writing your own. However, I wish to review and comment on the work that *you* do, so this must be clearly indicated. Failure to do so will be considered the same as any other breach of academic integrity.

Intellectual property rights
http://www.newschool.edu/provost/accreditation-policies/

Grade policies
http://www.newschool.edu/registrar/academic-policies/

Student code of conduct
https://www.newschool.edu/student-conduct/

Student Course Ratings

During the last two weeks of the semester, students are asked to provide feedback for each of their courses through an online survey. They cannot view grades until providing feedback or officially declining to do so. Course evaluations are a vital space where students can speak about the learning experience. It is an important process which provides valuable data about the successful delivery and support of a course or topic to both the faculty and administrators. Instructors rely on course rating surveys for feedback on the course and teaching methods, so they can understand what aspects of the class are most successful in teaching students, and what aspects might be improved or changed in future. Without this information, it can be difficult for an instructor to reflect upon and improve teaching methods and course design. In addition, program/department chairs and other administrators review course surveys. Instructions are available online at

http://www.newschool.edu/provost/course-evaluations-student-instructions.pdf

Course Outline

An outline of our course schedule for the semester will be provided on the class website, reviewed on the first day, and is included below.

Week 1 – Thinking like a computer?
Wednesday, September 2

- Introductions
- About the course
- What is a program?
- The Processing Development Environment ("PDE")
- Drawing with numbers
- The window as a grid of pixels

Part I: Python and the Processing platform

Week 2 – Adding variance
Wednesday, September 9

- Variables
- Arithmetic (+, -, *, /)
- Introduction to `random()`

Week 3 – Adding interactivity
Wednesday, September 16

- Code blocks and frames (`setup()` and `draw()`)
- Debugging (with `println()`)
- Mouse interaction (`mouseX` and `mouseY`, `pmouseX` and `pmouseY`)
- `map()`
- User testing (and *Export Application*)

Reading due for discussion:

- Lev Manovich. *The Language of New Media*, Cambridge, MA: MIT Press, 2002. Chapter 1 (pages 18-55)

Week 4 – Making things move
Wednesday, September 23

- Conditionals `if` and `else`
- Keyboard interaction
- Motion

Midterm project assignment
Assigned Wednesday, September 30;
Due: Monday, October 26, 8pm; for in-class presentation and discussion on Wednesday, October 28.

Week 5 – Adding repetition
Wednesday, September 30

- Loops

Reading due for discussion:

- Selections from Matthew Fuller, *Software Studies: A Lexicon*: Introduction, "Algorithm", "Code", "Programmability", and "Source Code".

Week 6 – Timing and state
Wednesday, October 7

- Timing
- State: a new way to use variables

Week 7 – Working with many things (lists), and modularity (function)
Wednesday, October 14

- Data structures: Lists
- Functions: for project planning, reusability and modularity

Reading due for discussion:

- "Data Visualization", from Matthew Fuller's *Software Studies: A Lexicon*
- Catherine D'Ignazio and Lauren Klein, "Unicorns, Janitors, Ninjas, Wizards, and Rock Stars," from *Data Feminism*, MIT Press, 2020

Week 8 – Midterm project work and review as needed
Wednesday, October 21

- Course material review & midterm project work

Reading due for discussion:

- Alex Galloway, "Gamic Action, Four Moments", chapter 1 (pages 1-38) from *Gaming: Essays on Algorithmic Culture*.
- Claus Pias, “The Game Player’s Duty: The User as the Gestalt of the Ports”, from *Media Archaeology: Approaches, Applications, Implications*, Jussi Parikka and Erkki Huhtamo, eds.

Week 9 – Midterm project presentations and discussion
Wednesday, October 28

- Midterm project presentations

Week 10 – Advising week
Wednesday, November 4

- Course material review & advising

Part II: Standalone Python – Networks and data processing

Week 11 – Data serialization: for storage and communication
Wednesday, November 11

- Introduction to Python outside of Processing with Atom and the command line
- Another data structure: Dictionaries
- Serialization with JSON
- An experiment with networking

Reading due for discussion:

- Paul Dourish, “Protocols, Packets, and Proximity: The Materiality of Internet Routing”, from *Signal Traffic: Critical Studies of Media Infrastructures*, Nicole Starosielski and Lisa Parks, eds.

Final project assignment
Assigned Wednesday, November 18;
Due: Monday, December 14, 8pm; for in-class presentation and discussion on December 16.

Week 12 – Network protocols
Wednesday, November 18

- Networking and the web

Reading due for discussion:

- Alex Galloway and Eugene Thacker, "Protocol and Counter-Protocol", 2003

Wednesday, November 25 – NO CLASS – *Have a restful Thanksgiving break …*

Week 13 – Saving data in a database
Wednesday, November 25

- Working with a simple SQLite database to implement an API
- Final project concepts due for feedback via email

Reading due for discussion

- Excerpt from Victoria Vesna, *Database Aesthetics: Art in the Age of Information Overflow* (2007)

Week 14 – Final project work
Wednesday, December 9

- Course material review & midterm project work

Week 15 – Final projects
Wednesday, December 16

- Final project presentations and discussion