

EN.540.635 “Software Carpentry”

Final Project

Project idea due Fri April 18 at 11:59 PM ET.

Final project submission due on May 6, before class.

Project presentations due on May 6.

Now that you have been exposed to the fundamentals of programming and writing code, it is time to make use of everything you have learned so far to write your own program! This project can be focused on anything you want - the only requirement is that it must be more involved than anything we have done so far in class (including the Lazor project). This means that your project should contain a substantial amount of code and you should be dedicating an appropriate amount of time and effort to complete it - **we should not be able to reproduce your project within a single day**. We do ask that all of the code is written using Python - this will help us with grading. You are allowed to work on this project either individually or in pairs.

First, you must come up with a final project topic and have it approved by either of the co-instructors (and you must also let us know if you are working in a group) by the specified due date. If you are unsure of what topic to do, we have listed a few general ideas below with links:

- Programs around main concepts we’ve built up and covered in class (object-oriented programming for example)
- Programs related to your research (streamlining data collection, data analysis, etc.).
- Programs that use existing APIs to complete some task.
- Programs with GUIs.
- [Web applications using Django](#).
- [Programs using artificial intelligence and machine learning](#).

You are not limited to what is listed here! There are many other ideas that you can find online if you look for them. **If you have trouble finding a project topic, please email the co-instructors for the course and let us know and we will help you figure something out.**

Once you have selected a topic and it has been approved by us, you can get started working on it! **The general rubric (all the things we will consider during grading) is shown on the second page of this handout - if there are any questions, please email us and ask!** This project should be completed in Github (similarly to how the Lazor project is done), where you are constantly committing small changes to keep track of your progress over time and there is appropriate documentation so that anyone would be able to understand how your program works. **All project submissions will be done through Canvas - there will be appropriate assignments where you can submit the appropriate text and links to Github.**

One thing to keep in mind is that it is very important that we adhere to the university’s policies on academic integrity. Your submitted work should be original - you should not copy or edit anyone else’s code and submit it as your own. If you do find relevant code from the Internet and use it in your own work, you **MUST** properly cite it in the README for your project and within your code itself. If we find any violations of this, there will be consequences depending on the severity. For further information, please reference the latest version of the [Graduate Academic Misconduct Policy](#). If there are any questions or concerns about this, please email the instructors and ask.

Because there are no in-person classes for the remainder of the semester, this may have an impact on the work that you submit for this project. If you working with a partner, it is important that you plan accordingly to complete this project by the deadline. We expect that you properly use Github to work on this project in collaboration with your partner - the work should be divided as evenly as possible between the two of you and you both should be making commits to the remote repository where all your code is located on Github. If you are working with a partner, you will be required to complete a peer review form after the project has been submitted. **If there are any issues regarding your partner or should you fall ill during this time, please contact us as soon as possible so we can properly address the issue.**

On May 6 we will have a meeting session where everyone will present their projects to the instructors and other students in the class. This will take place during the reserved exam time period for the class. These will be short presentations (10-minute maximum, but exact timing will be chosen based on number of groups) where you explain your project and your code and demonstrate how it works.

Points Considered During Grading

Project Difficulty

- Was the project appropriately difficult enough?

General Coding

- Does the code have any bugs?
- Are variables named properly?
- Is there good use of appropriate coding structures (functions, classes, data structures)?
- Is the code laid out nicely and properly organized across multiple files?
- Are there unit tests and error handling?
- Is the code written idiomatically?
- Are there docstrings for all functions and classes?
- Are there appropriate comments explaining the code?
- Does the code adhere to PEP8 styling?

Dissemination

- Is the README written well? Does it properly explain the project?
- What are the requirements for running the code (required modules, APIs, etc.)?
- How is a user supposed to run the program?
- Are there any external sources of code that were referenced and are they cited properly?

Github and Peer Review

- Was Github used appropriately?
- Did each student contribute appropriately?

Presentation

- Did you present the general project concept?
- Did you present an outline of your code?
- Did you demonstrate how your program works?