

UNIVERSITY OF MICHIGAN
SI 106: Programs, Information, and People

Final Project

Project Plan Due: Thursday 12/3 before your section meets

Demos in section: Thurs. 12/10

Project Due: Tuesday, Dec. 14 Noon

Overview

You have several options for what you can do for your final project:

1. Do some kind of interesting analysis that mashes up two information sources on the Internet (using their REST APIs).
2. Improve the Shannon game guesser.
3. Use a Shannon guesser to estimate the entropy of some interesting texts. For example, you might compare who among your friends is least redundant in their posts or comments (using their previous posts to continuously train the guesser). Or you might compare some liberal and conservative politicians to see who is most redundant (perhaps the word for this is “staying on message”).
4. Create a game with animation (and possibly sound) using the pyglet module.
5. Something else that you propose to the instructional team that meets the minimum requirements below.

We highly recommend that you discuss your project ideas, show off your app, exchange tips, and ask for clarifications on the Facebook group.

Project Plan (250 points) due Thursday, 12/3 before your section meets

What to Hand In

Download the file `project_plan.py` from Canvas. You should modify it and upload it to cTools.

Requirements

`project_plan.py` should contain three elements.

1. A description of your project. The lecture prep for Wed. 12/2 will walk you through this. It should include:
 - a. What data sources you will use
 - b. How you plan to process the data you will get. This should be a high-level description, in English. For example, if you were describing the project that was carried out in Problem Set 12, you might write, “Get the most recent 150 posts from the class Facebook group, calculate the emo score for each and sort the posts by their emo scores.
 - c. How you will present the results. For example, give a sample of what the output might look like, or describe the chart that might be generated in Excel.
2. A description of at least one Class that you will define. You will generate this by filling in a bunch of blanks in `project_plan.py`
3. For each API that you will use, provide answers to the following questions, analogous to the answers provided in the document “Using APIs summary table.pdf”
 - a. Modules
 - b. Developer key

- c. Authorization for particular users
- d. Making requests
- e. Parsing responses

Final Project (1750 points)

What to Hand In

Create a README.txt file that explains what your program does, and any python modules that need to be installed using pip in order for your program to run. If you want to include diagrams and screenshots to illustrate your program running, include them in a PDF file. Upload these to Canvas.

Create one or more .py files with your code in it. Upload these to Canvas.

Requirements

Your application must fulfill the following requirements to get full credit.

1. If you do option 1 (analysis based on external data), then you must get data from *at least* two sources. One may be a source that we've used in the course (FAA, Flickr, iTunes, Facebook, Twitter). One must be something else (a file or an API). At least one must be an Internet API. (The rest of the requirements apply to everyone).
2. Create some output meant for human consumption, either nicely formatted text or a datafile that you import into a spreadsheet or visualization program to produce one or more charts, or something that you show to users interactively. Doesn't have to be fancy, but you should be able to demo your program to someone who is not in the class and they should be able to understand what your program does.
3. Perform the following computations:
 - a. At least one accumulation somewhere
 - b. At least one non-trivial sort (i.e., where you have to provide a key function)
4. Use the following python constructs:
 - a. Define a Class, with an `__init__` method and at least two other methods
 - b. Use at least one python module, either a built-in one or one that you install with pip
 - c. Use at least one list comprehension or calls to map and/or filter
 - d. Define at least five functions or methods
 - e. For at least three functions or methods, create one or more test cases that test whether the functions produce the right outputs on particular inputs
5. Do something creative, something that isn't exactly analogous to what you've already done on a problem set or in class. The bar is low here, but do something that shows you had to go beyond copying and lightly editing existing code snippets.