

SI 106 FINAL PROJECT README

* In ~2-3 sentences, what does your project do?

If you run the program, and input a hashtag search query where the program asks you to input it, the program will: search twitter for the last 200 tweets containing that hashtag, eliminate common vernacular, and create a dictionary of the most popular words that came from that search query ranked in order. Then the program uses Itunes to search that word and returns 50 results for you to use in a playlist if you so desired.

* What files (by name) are included in your submission?

SI 106 FINAL PROJECT README.pdf

FinalProject.ipynb

SAMPLEcreds.txt

SAMPLEsonginfo.csv

SAMPLEenglish (English is a file found nested in nltk_data, which the program will import to use. However, I will attach a sample in the submission with this name.

* What Python modules must be pip installed in order to run your submission?

Json, csv, requests_oauthlib, unittest, webbrowser, requests, nltk

* Explain SPECIFICALLY how to run your code. We should very easily know, after reading this:

Run all of the cells in the file “FinalProject.ipynb”. When you do there will be two areas in which the user is required to input information. The first is to authorize the credentials. A link will appear after you run cell block #4 and you will have to click on it, and select authorize in order to use the application. There are directions in the program that tell you this as well. The second and last time you will need to interact with the program is in cellblock #9, where the program asks you what hashtag you wish to search on Twitter. You can type in any word (without the hashtag) and it will search it on twitter. At the end of the program, there is a summary that will provide information on what you found using the program including: the most popular word on Twitter, one of the tweets results from your search, and the first song that comes up when you search Itunes to make a playlist.

* Where can we find all of the project technical requirements in your code? Fill in with the requirements list below.

REQUIREMENTS LIST (for each, tell us where to find them in your notebook):

- *Get data from 2 different REST APIs*
 - The Twitter REST API uses the first 7 blocks of code to get data. When I specifically get the data is the cell block that starts with “requests any hashtag from user” in lines 9-19.
 - The Itunes REST API gets its data from cell block #17 which starts with “gets info from itunes”. All lines (1-7) are being used to get the data from Itunes here.
- *Define at least 2 classes*
 - Class Tweet() can be found in the block starting with with #definition of class Tweet and includes all lines from 1-15

- Class Song() can be found in block starting with #definition of class Song and includes all lines from 1-24
- Each class must have
 - *A constructor*
 - Constructor for Tweet is the first definition in the class and is in the block starting with #definition of class Tweet (starts at line 3)
 - Constructor for Song is the first definition in the class and is in the block starting with #definition of class Song (starts at line 3)
 - *An __str__ method*
 - For class Tweet(), block starting with #definition of class Tweet (starts at line 10)
 - For class Song(), block starting with #definition of class Song (starts at line 15)
 - *At least one additional method*
 - Class Tweet(), starts with #definition of class Tweet (starts at line 14)
 - Class Song(), starts with #definition of class Song (starts at line 18)
 - *At least 3 instance variables that might be set in each instance*
 - For class Tweet, instance variables are text (which holds the words in the tweet), favorites (which stores the number of favorites the post received), and retweets (which stores the number of retweets the post received). Found in its constructor (look above)
 - For class Song, there are 5 instance variables that store the part of the song that corresponds with the variable name. These are title, length, artist, album, genre. Found in its constructor (look above)
- *You must create at least 1 instance of each class you define*
 - Instance for Tweets can be found in the for loop starting in the cell starting with creates LIST in line 5
 - Additional instance variable “savedTweet”. Found in block starting with #creates list of instances for class tweet (line 9) and used in last cell of program.
 - Instances for Song can be found in the cell block starting with creates list of song instances and is on line 2
 - Additional instance variable “savedSong”. Found in block starting with #creates list of Song() instances (line 8) and is used in last cell of program.
- *Your code must invoke the string method and the additional method for each*
 - Class Tweet(): string method invoked in last cell, other method (stopwords) invoked in cell block starting with #creates LIST of instances of class tweet (line 10)
 - Class Song(): string method invoked in the last cell, other method (converttominutes) used in block starting with #creates CSV file (line 6)
- *You must perform at least one sort with a `key` parameter on data.*
 - Cell block starting with Using those instances), line 16.
- *You must define at least 2 functions outside the class definitions that have more than one line inside the function body.*
 - Cell block 2 (starting with #Gives list of stop words) line 2 is installnlkt()
 - It is invoked in cell block 15, line 2

- Cell block starting with #writes final product) line 2 is wsonginfo(L)
 - It is invoked in cell block 21, line 2
- Second to last cell summary() invoked in last cell (line 3)
- *Your code must create a file that is clearly structured for output.*
 - Found in the file songinfo.csv

END REQUIREMENTS LIST

*** Put any citations you need below.**

- I used the SI 106 textbook to allow me to get the credentials to use Twitter's API.
- I used the Facebook posts to help with the NLTK import

*** Explain in a couple sentences what should happen as a RESULT of your code running:**

- The code will create a few files, but the most important of those is the one named "songinfo.csv". This file will have the top 50 songs from the Itunes search query. It contains the title, artist, length, album, and genre of each song in CSV format.