**SI 206 Final Project Report**

1. **Goals**

* My objective for this project was to extrapolate data from social media API’s and use it to determine if the popularity of certain movies on these platforms had any correlation to how much overall revenue the movie accumulated.

1. **Goals Achieved**

* Though difficult, I did achieve my goal in determining if there is any correlation between social media popularity and overall revenue for a few movies. I wound up using the YouTube and Reddit API’s to find the social media data, and the TMDB API to find the current revenues for each movie.

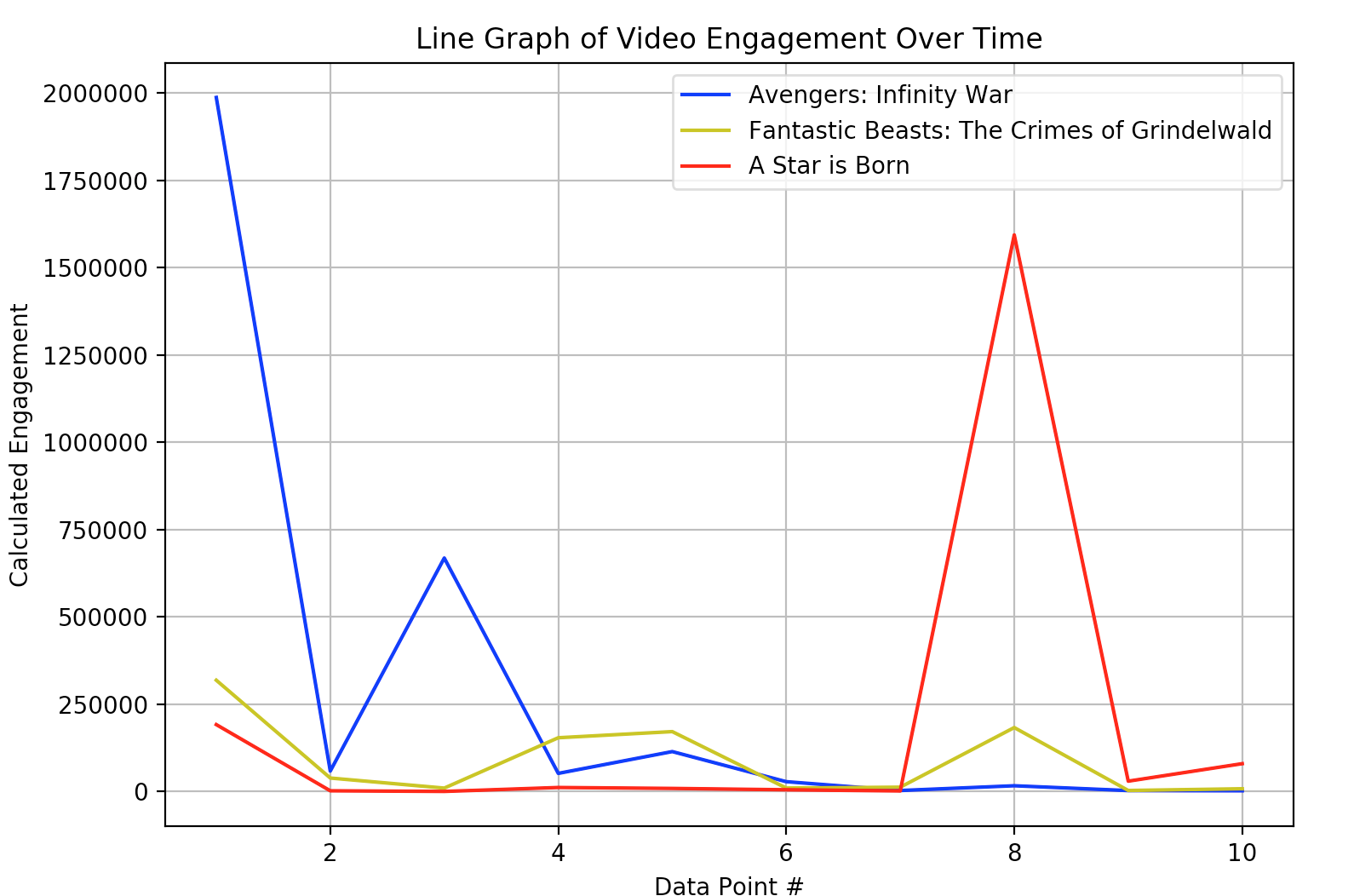
1. **Problems Faced**

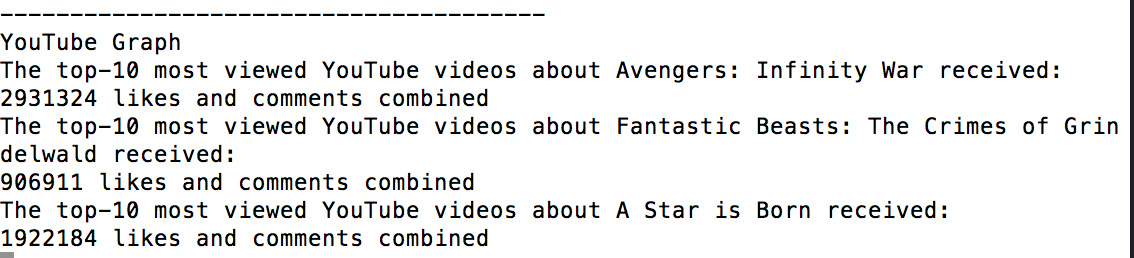
* I found this endeavor to be far more difficult than I expected. I tried to avoid using the Twitter API even though it would have been helpful. I experimented with other API’s (Facebook, Instagram) before I settled on Reddit and YouTube. The API’s I experimented with, though they may have been helpful, I struggled to quickly discover how to access. I was successful with Reddit and YouTube and decided to stick with those platforms.
* After getting access to the correct data, the rest of the project went seamlessly with the exception of manipulating the data to make findings. I wasn’t sure exactly how I wanted to approach determining how popular each movie was on social media platforms. Ultimately, for YouTube, I decided to look at the top 10 most popular videos relating to each movie and based popularity on views, comments, and likes. On Reddit, I looked at a large number of posts relating to the movies, and just summed up the total comments and upvotes for each movie.
* One final problem I encountered is the limits of TMDB. They have an extensive database of movies in their API, however the revenue data was not given for many popular movies. I had to curate the movies I searched to TMDB to make sure the data was available.

1. **Social Media “Report”**

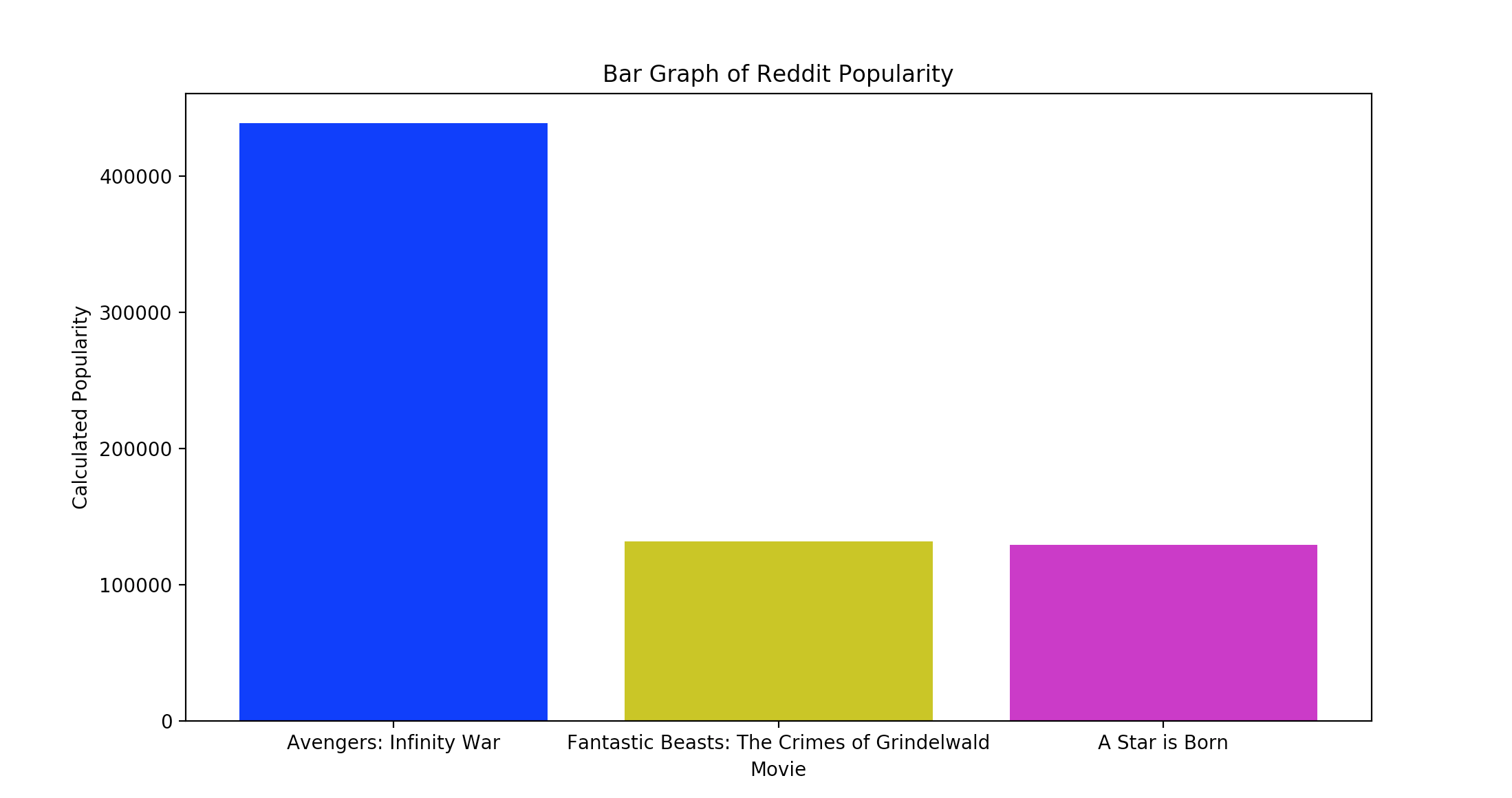
In short, I found that social media popularity does NOT always predict how much revenue a movie will make. The data shows us that “Avengers: Infinity War” had more popular videos and Reddit posts and this was reflected in their impressive revenue. However, the movie “A Star is Born” was MORE popular on Reddit and slightly LESS popular on Reddit than “Fantastic Beasts.” A Star is Born still had a greater overall revenue than Fantastic Beasts at the time the program was run. The data showing this is pictured below.

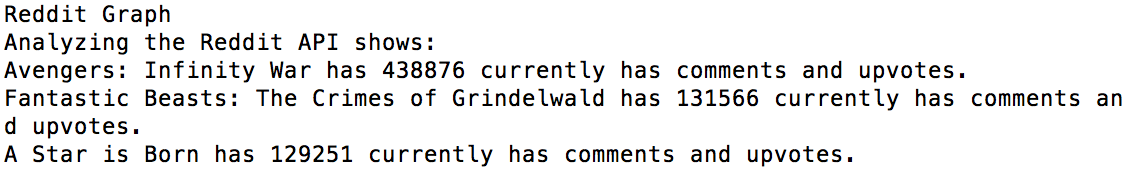
**YouTube Line Graph Data**



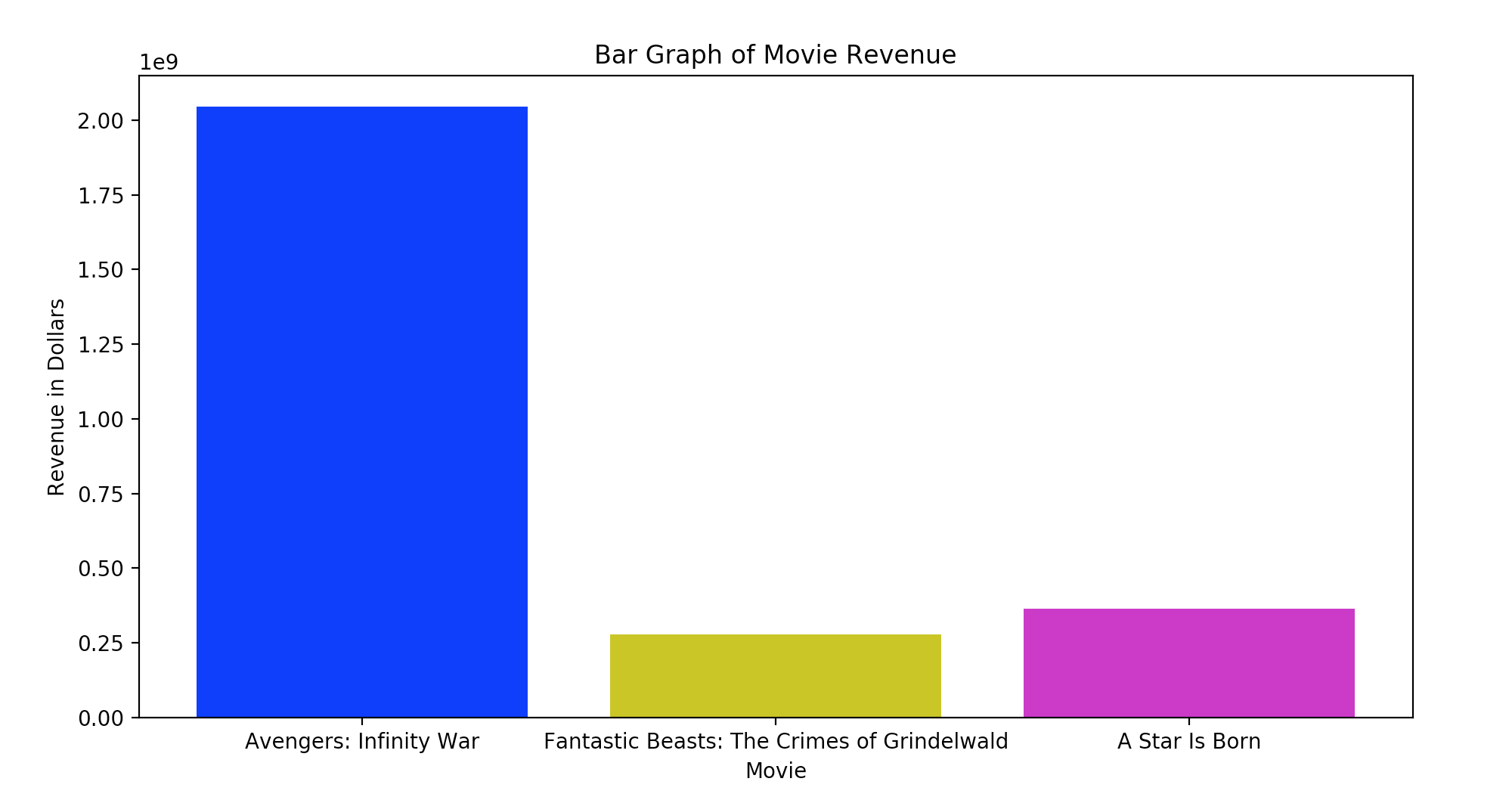


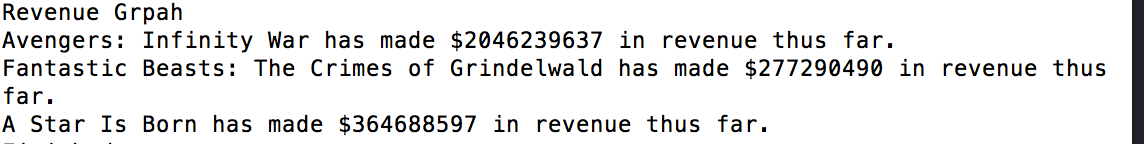
**Reddit Bar Graph Data**





**TMDB Movie Revenue Data**





1. **Instructions for Running Code**

*Note:* These instructions assume the user has a working IDE and has installed the Anaconda python package. The program will also be using my personal credentials to gain access to most APIs.

*Set up:* Many of the necessary imports are already in the program. It’s already called. However, there are a few pip installs that are necessary for the project. They are listed below.

PIP install TMDBsimple

PIP install praw

pip install --upgrade google-api-python-client

pip install --upgrade google-auth google-auth-oauthlib google-auth-httplib2

python -m pip install -U pip

python -m pip install -U matplotlib

*Step 1:* The program runs off of one main function. As mentioned above, there are some limitations with the TMDB database so movie titles have already been added to the function call. With the folder directory open (**final-project-griffpri20)**, type the command **python3 FinalProject.py** (if this doesn’t work try beginning with pythonw or python). This will run the entire program.

*Step 2:* To run the file, it’s necessary to log in to a google account to use the YouTube API. The terminal window will walk the user through this process. A link will pop up. The user must copy and paste this link into a browser of their choosing, and follow the commands. Successful completion of this task will result in a code that the user must then copy and paste back into the terminal window to continue running the program.

*Step 3:* The program will then create two SQL tables; one for YouTube, and one for Reddit. This will take a bit of time. When it completes these tasks, it will display a graph and its findings from the data captured from the APIs.

1. **Documentation for each function**

* Located in code

1. **Resources used**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Issue Description** | **Location of Resource** | **Result** |
| 12/3/18 | Needed help getting YouTube Data | [Documentation](https://developers.google.com/youtube/v3/quickstart/python) | Got YouTube Data |
| 12/6/18 | Wanted easy way to access TMDB data | [Documentation](https://github.com/celiao/tmdbsimple) | Helped get and understand TMDB data |
| 12/4/18 | Wanted package to help get Reddit data from API | [Documentation](https://praw.readthedocs.io/en/latest/index.html) | Got Reddit data |
| 12/8/18 | Needed better understanding of MatPlotLib package | [Documentation](https://matplotlib.org/index.html) | Got help and was able to add legend/color etc… |
| 12/6/18 | SQL error with trying to add external variable to executed code | [Source](https://medium.com/@felipe.f.m/sqlite3-programmingerror-incorrect-number-of-bindings-supplied-2e3583d7e034) | Needed to add comma. Worked after this. |
| 12/6/18 | Reddit used unix time stamps. Needed to convert. | [Source](https://www.w3resource.com/python-exercises/date-time-exercise/python-date-time-exercise-6.php) | Time was converted and I was able to understand it. |