Music All Over

Project Description:

a. Overview: the project is about combining Spotify and Google Map to analyze the preference of the type of music of people in different regions.
b. Motivation: I personally love to listen to music. However, whenever I go to a new place and meet new people. I found that people from different region and age have different preference of type of music. Therefore, I have the inspiration to build Magic All Over, to combine people's preference of the type of music and their location on Google Map. In order to analyze the relation between region and the preference of the type of music being listened. According to the result, the music producers (including singers, composers, lyricist and so on, basically include every person in the music scene) could have a clarify tendency of the group they aim for. Therefore, they can produce more music that matches the demand of the group they aim for, and create maximum

efficiency.

Project Planning:

I will be combining Spotify and Google Map, by analyzing the tendency of music the users from different ages and regions listen to and then point it out on Google Map.

1. Find the website for API:

   I will find the website of the API I need. For now, it will be the API website for Spotify and the API website for Google Map. (But it Spotify is not available, I might change Spotify to KKBox, YouTube, or other music player.)

2. Obtain the API keys:

   I will study the website found above and look through the Tools for Web Developers to try to get the specific API key for the information I need.

3. Analyze the data obtained from Spotify:

   I plan to analyze the information obtained from Spotify. To do so, I will be aiming at high school (or even college) students. (Which is the largest group of music listeners I suppose.) Analyze which type of music they listen to the

most to find their tendency of type of music.

4. Web crawl the IP of users:

   Crawl the user's IP of Spotify, and distinguish the user's location by city. For the user's IP, I will crawl Spotify or IPLoc.(A website which you can find the IP location of the user's data)

5. Combine IP with Google Map:

   Using the IP address received above, point out the location on Google Map. When pointing out, distinguish IP address in the unit of city. And then point out its location on Google Map, print the population of the region at the same time.

6. Combine Spotify with Google Map:

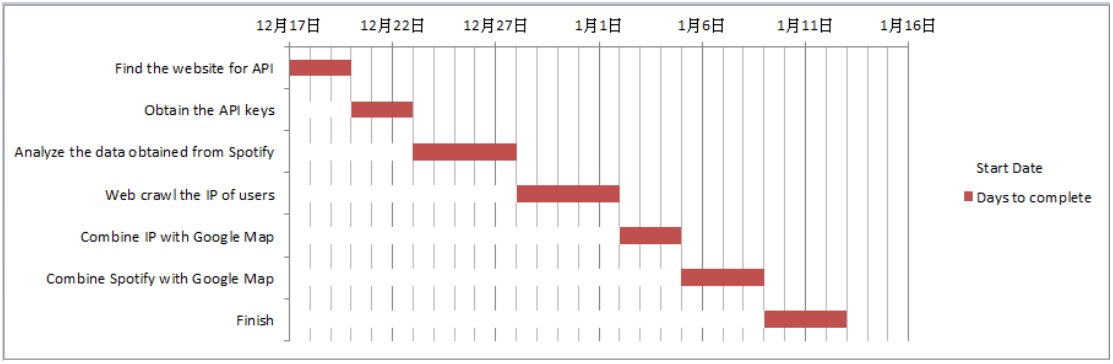
   Using the pie chart acquire above, point the result on Google Map. Print the corresponding pie chart according to each location and age of listeners on Google Map.

7. Finish:

   After pointing out the pie chart and population on its location, we can easily see the tendency of type of music of the listeners. Which can provide the music producers a

clarify direction to aim for when composing.

Timeline:



Update 1

1. What I have done:
    a. I had did some research for the API that I need, and try to use them to get the data that I want. However, I am still trying to put it into my code and make use of it.
    b. I web crawled the country codes (alpha-2) of all countries from Wikipedia since the web of the post office couldn't be crawled. Therefore the web crawl part is finished. However, if time is enough, I will add another crawl for the name of the countries. Since not every people know the country code of the country they wish to search.
2. Any Changes on the final project plan, and why:
    a. Change the API of data from Spotify from top list to new release:

       I searched for many websites and information and found that the API for top list was available until last year, but the policy changed this year. The top list API now is only available for music producers, licensor, label, or music

companies. Therefore I will change the API of the top list to the new released.

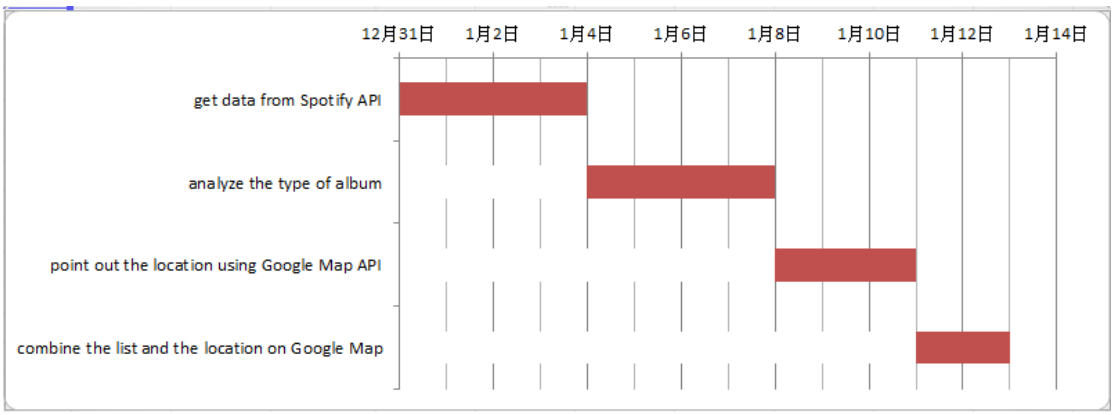
b. Change the users from person to country:

   Since the data I use will change, the user type will be changing simultaneously. According to the API, it requests for country and number of the result you wish to get. Therefore, I changed the input of user from person to country to get the information I need.

c. Analyze data (album type/ pie chart):

   If time is enough and the data I get is able to be analyzed, I will analyze which type of the album is. (single album, video album, live album…..)

3. Timeline for the rest:



Update2

Modifications:

1. Change API from google map to rest countries:

   I looked up for google map API and found that it requires for a billing account. Unfortunately, when I was registering, I found that it doesn't support JCB, which is my only credit card. Therefore, I am forced to abandon google map and look for other API, then I found rest countries. According to the data I can get from rest countries, I change the use of API to requesting the country name and its capital city.

2. Change of crawling:

   a. Change the crawl of country code from wiki to countrycode.org:

      The crawl of country code from wiki is feasible. However, the English name of the country is difficult to obtain in the same order as its Chinese name and country code. Therefore, I decided to abandon the Chinese name of the country. According to this, I

found countrycode.org and do the crawling depending on it.

b. Add another crawl for the countries Spotify is available in:Since Spotify is not available in every country, I crawled for the countries that Spotify is available in as an inspection.

3. Use of PrettyTable:

   Except the analyzation and pie chart of album type, I decided to print out all three of the artist, album name, and album type in a table format. Therefore, I did some research online and found PrettyTable to accomplish my requirement.

4. Add comparison:

   From the inspiration of the TA, I add another input country2 and do the same as the previous input country1. According to the pie chart, the type of the new release albums could easily be compared. You can also see the top 10 latest released albums from the PrettyTable.

Run

Remember to require a new oauth token from https://developer.spotify.com/console/get-new-releases/?country=AX&limit=10&offset=5 and change the auth_token in the code, since the token expires in a couple of hours.

```
curl -X "GET" "https://api.spotify.com/v1/browse/new-
releases?country=AX&limit=10&offset=5" -H "Accept:
application/json" -H "Content-Type: application/json" -H
"Authorization: Bearer BQB_T-
kJ4BMnILMbk2AJFyAH_Y2uuBwQ9sEWRQDb6Kso7rc4Dzr7v0OND21uwfjbOjb
0fbRHzmT5zAJy6W1Bnu2Z9WKW7K-
SvgIzhKFA21RPsiEyTeC2DupAshIgxK7MZRGwdf8BMJi7ydh1kugMPCgWT38t
Hv7Ha3ANqU8"
```

(The oauth token is the highlighting part and auth_token is at line 62 of the code.)

The code explanations are in the codebase.

To run the project, run the project in terminal by pressing run and enter the country code or English name of the countries you wish to compare. The input can be either upper case or lower case, both will work for the project. Do not run the code online, the pie chart will not be showing then.

(Take Taiwan as example, the alpha-2[also known as iso 3166-1]

code for Taiwan is TW. You can enter any combination of upper case and lower case of ‘tw’ or ‘taiwan’, it will all work for the result of taiwan.)

Libraries:

Install libraries under cmd:

For crawling:

- python –m pip install requests
- python –m pip install bs4

For pie chart drawing:

- python –m pip install matplotlib

For output in table format:

- python –m pip install PrettyTable

Title setting for PrettyTable:

- python –m pip install PTable

Take “tw, GB” as an example input, the output will be like this: