|  |  |  |
| --- | --- | --- |
|  |  | Create a spotify playlist based on location  SI 507 intermediate programming |

April 23rd, 2020

# Shengnan (Elenore) Duan

**GitHub repo:**

<https://github.com/elenoreD/SI_507_Intermediate_programming>

**Demo video:**

https://drive.google.com/file/d/1bKOcnP0-lY3YX85sP0FmV\_m2DIKbj8pB/view?usp=sharing



## Project code

The project has 2 major functions: a) Create a playlist for user’s Spotify account based on location by scraping website and accessing Spotify API; b) Visually present the artist/tracks result by Flask app.

The 4 authentication information needed for using Spotify API and how to be acquired are as below:

* GET\_TOKENS: <https://developer.spotify.com/console/get-artist-albums/>
* POST\_TOKENS: https://developer.spotify.com/console/post-playlist-tracks/
* USER\_IDS = https://www.spotify.com/us/account/overview/
* COUNTRYS = 'US'

It should be noticed that a) Tokens could be expired, but they can be applied for unlimited times. The program will have notification when they expired; b) By default, the search results are limited in United States.

Packages needed: requests, Flask, sqlite3, pandas, Beautifulsoup

## Data sources

1. Webpage origin: <https://www.ranker.com/list/chicago-bands-and-musical-artists-from-here/reference>

It a webpage about artists in different locations. I start search by changing the ‘location’ part in the URL, and then I scrape the page to get artist name, rankings, artist picture link and the artist description. I will use cache by creating a unique key for each URL of a location. The records amount could be different by locations, since some city have more artists, and some don’t. I will only scrape the first page of the result. For example, Chicago has 50 artists while Ann Arbor has 21 results. This doesn’t matter, I will add everything of the first result page to my database later.

Usually, an artist could have information like name, genre, nationality, description and so on. The records retrieved will be a dictionary. Below is an example of the first element of some keys in the dictionary:

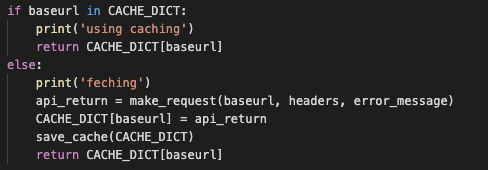


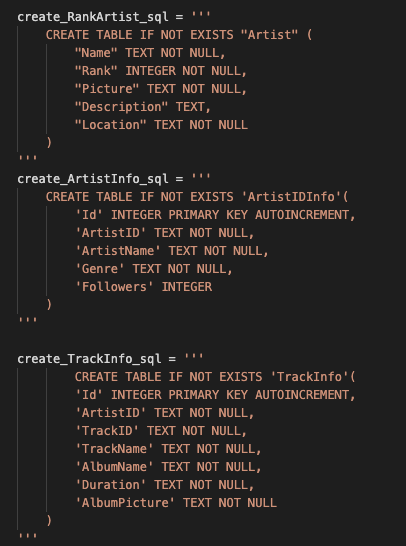
2. API origin: [https://api.spotify.com](https://api.spotify.com/)

It’s the Spotify API. I need to change the endpoint for different request (playlist, artist etc). I mainly use the API to 1) get artist ID by artist name I collected from web page 2) get artist top tracks by artist ID 3) post request to create an empty playlist 4) post request to add the top tracks to the empty playlist. Below is the example of the artist ID of the first Ann Arbor artists:

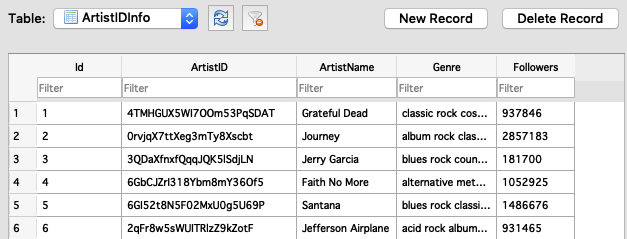


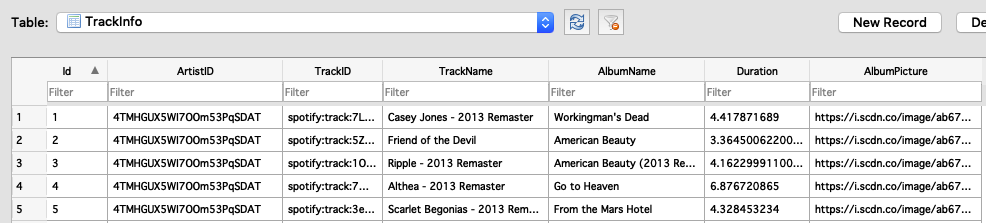
3. evidence of using cache

**Database**



Artist is linked with ArtistIDInfo by ArtistName, ArtistIDInfo is linked with TrackInfo by ArtistID.





## Interaction and Presentation Options

In command line, user can get an automatically generated playlist by simply type the location they interested. An example can be shown below:





Next part is conducted by Flask and Plotly package. User can interact with the app by typing, clicking radio buttons, choosing dropdown list and checkbox as below:

1. Users can check the artist/track results of certain city by inputting a location.
2. Users can choose to sort the results by either artist follower amount or track duration length.
3. The ascending or descending order could also be specified.
4. The search results can be filtered by music genre.
5. Except the general information, users can check the rank of artist’s popularity based on their followers from a bar chart.

