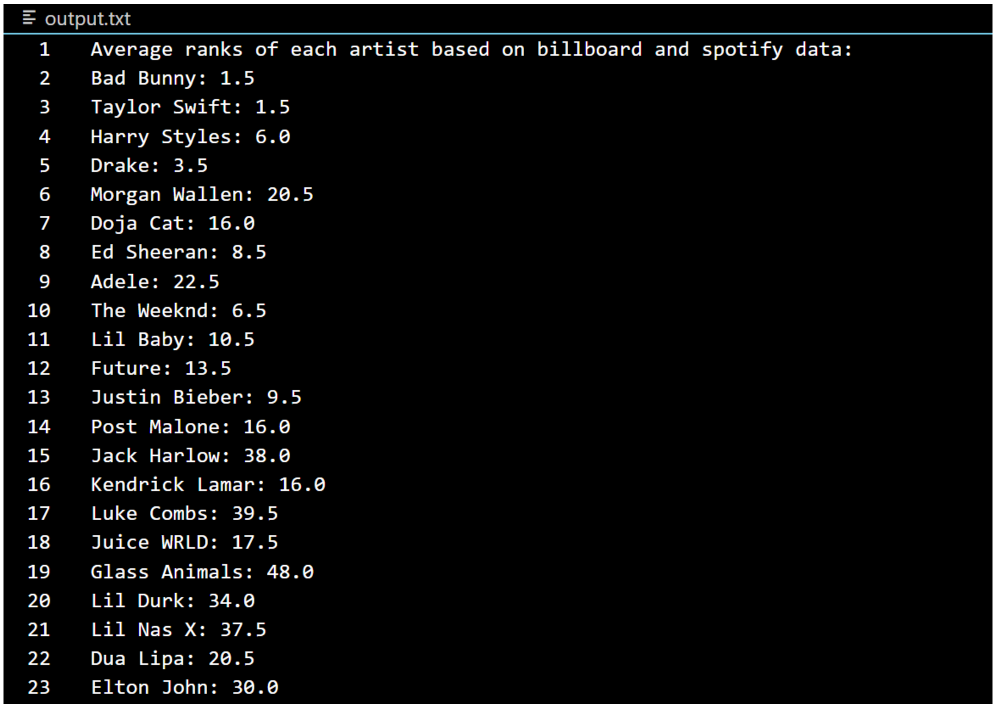
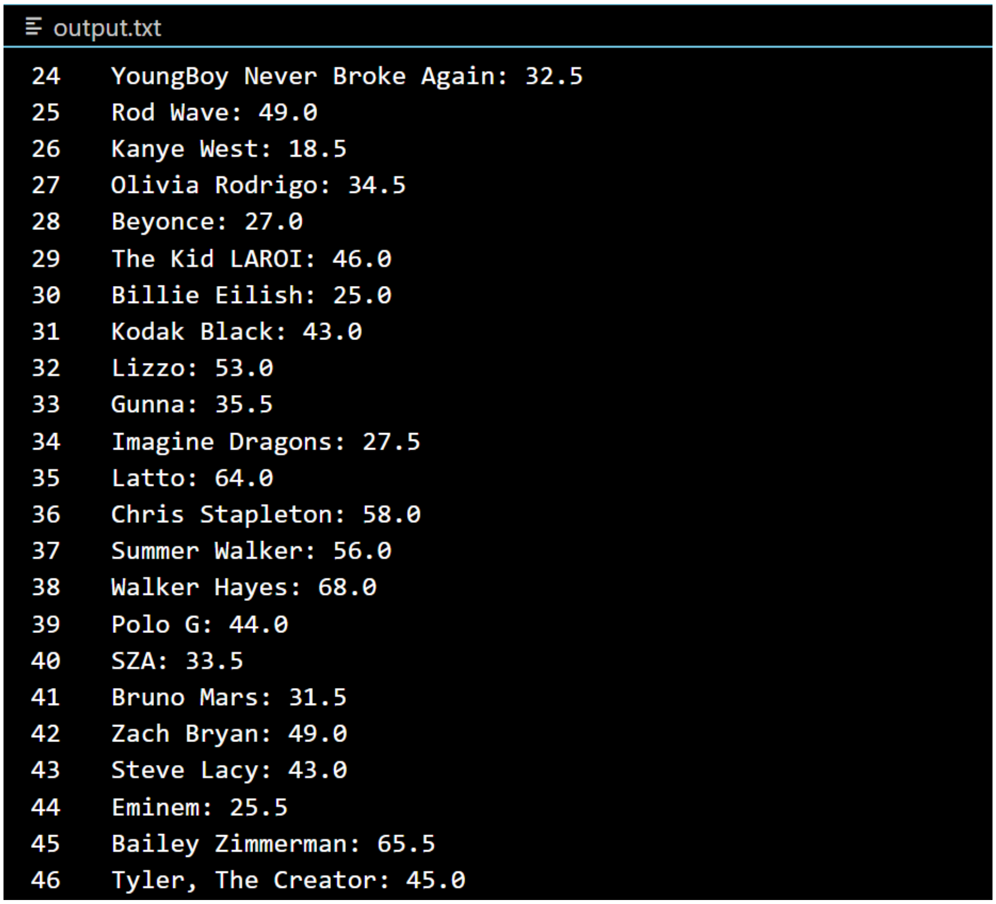
# Final Project Output

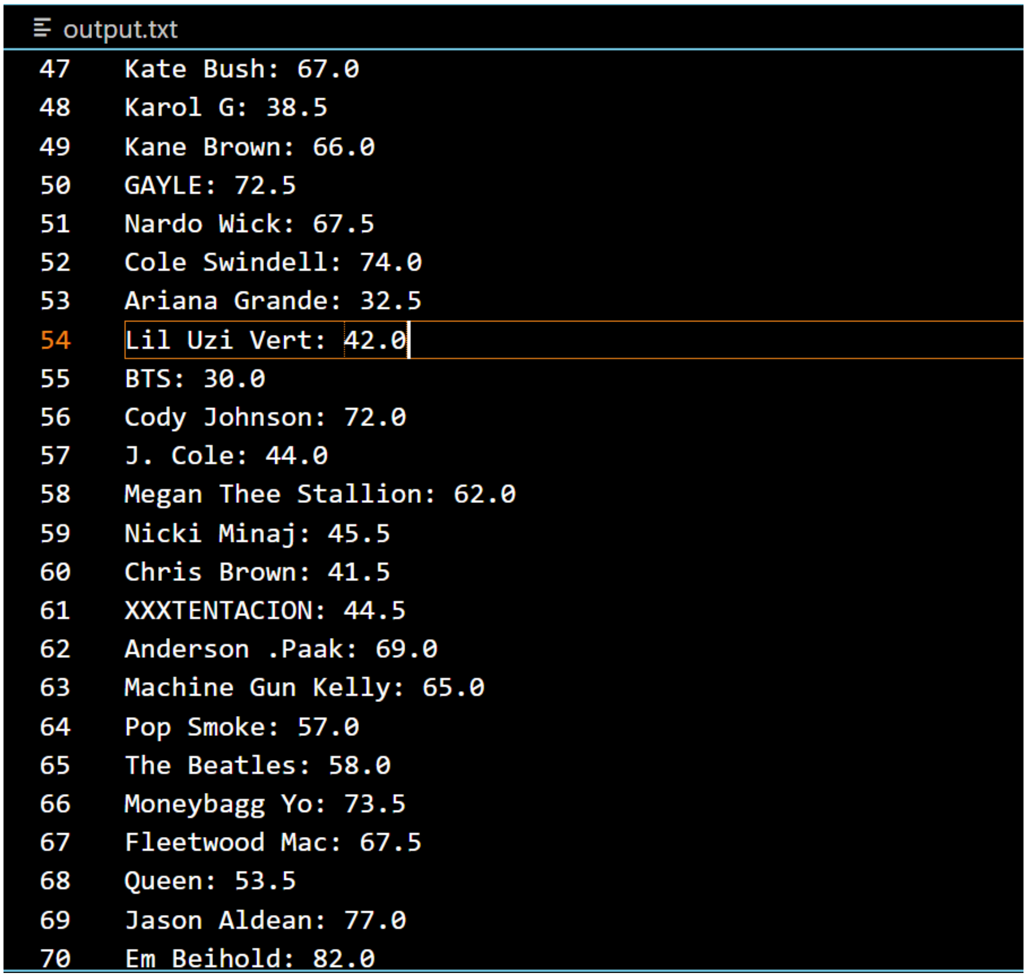
## 1. Output



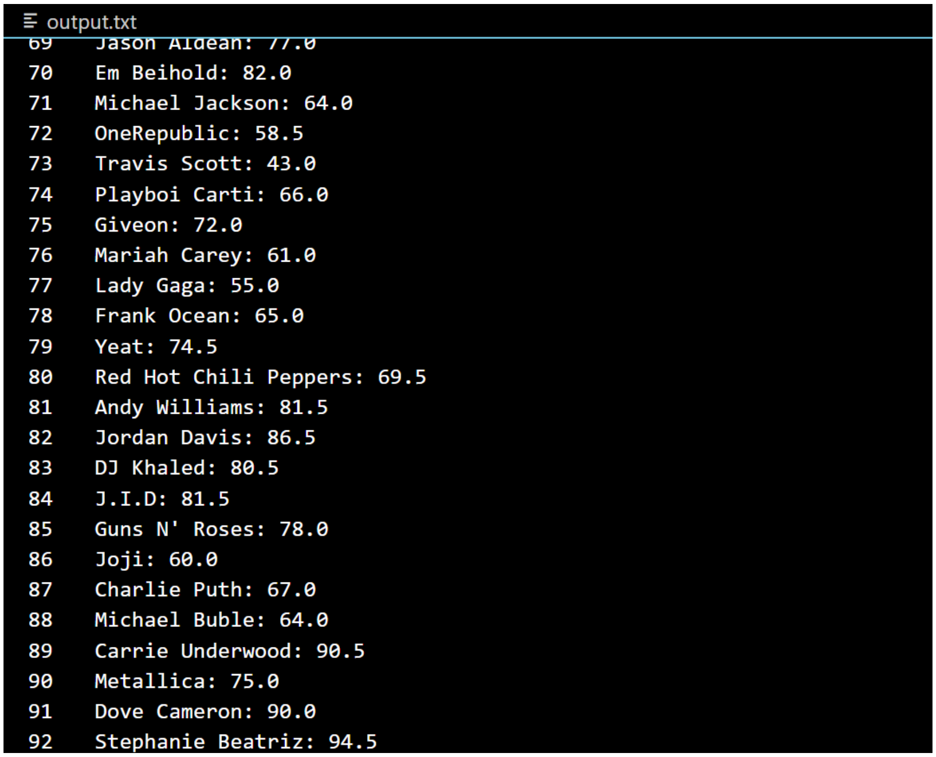
```
Average ranks of each artist based on billboard and spotify data:
Bad Bunny: 1.5
Taylor Swift: 1.5
Harry Styles: 6.0
Drake: 3.5
Morgan Wallen: 20.5
Doja Cat: 16.0
Ed Sheeran: 8.5
Adele: 22.5
The Weeknd: 6.5
Lil Baby: 10.5
Future: 13.5
Justin Bieber: 9.5
Post Malone: 16.0
Jack Harlow: 38.0
Kendrick Lamar: 16.0
Luke Combs: 39.5
Juice WRLD: 17.5
Glass Animals: 48.0
Lil Durk: 34.0
Lil Nas X: 37.5
Dua Lipa: 20.5
Elton John: 30.0
```



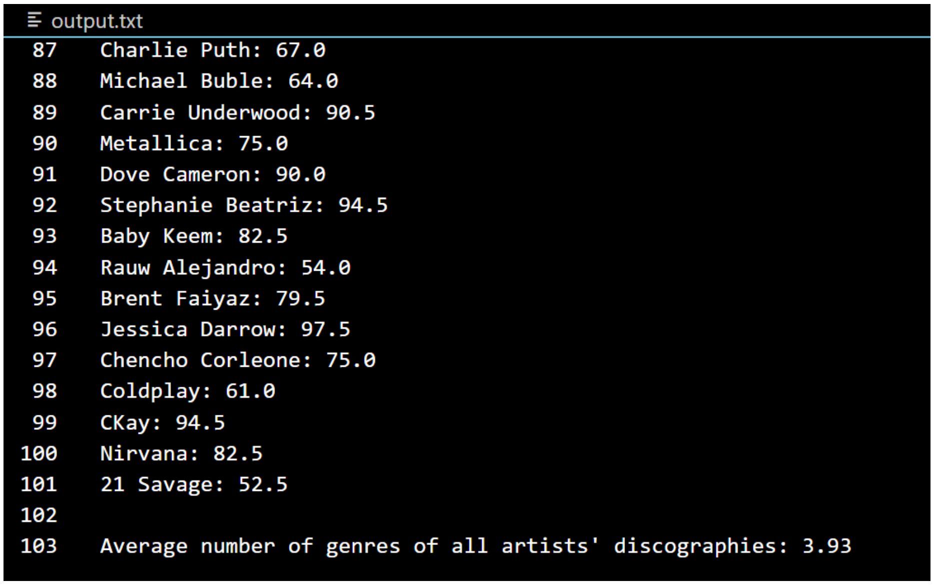
```
YoungBoy Never Broke Again: 32.5
Rod Wave: 49.0
Kanye West: 18.5
Olivia Rodrigo: 34.5
Beyonce: 27.0
The Kid LAROI: 46.0
Billie Eilish: 25.0
Kodak Black: 43.0
Lizzo: 53.0
Gunna: 35.5
Imagine Dragons: 27.5
Latto: 64.0
Chris Stapleton: 58.0
Summer Walker: 56.0
Walker Hayes: 68.0
Polo G: 44.0
SZA: 33.5
Bruno Mars: 31.5
Zach Bryan: 49.0
Steve Lacy: 43.0
Eminem: 25.5
Bailey Zimmerman: 65.5
Tyler, The Creator: 45.0
```



```
Kate Bush: 67.0
Karol G: 38.5
Kane Brown: 66.0
GAYLE: 72.5
Nardo Wick: 67.5
Cole Swindell: 74.0
Ariana Grande: 32.5
Lil Uzi Vert: 42.0
BTS: 30.0
Cody Johnson: 72.0
J. Cole: 44.0
Megan Thee Stallion: 62.0
Nicki Minaj: 45.5
Chris Brown: 41.5
XXXTENTACION: 44.5
Anderson .Paak: 69.0
Machine Gun Kelly: 65.0
Pop Smoke: 57.0
The Beatles: 58.0
Moneybagg Yo: 73.5
Fleetwood Mac: 67.5
Queen: 53.5
Jason Aldean: 77.0
Em Beihold: 82.0
```



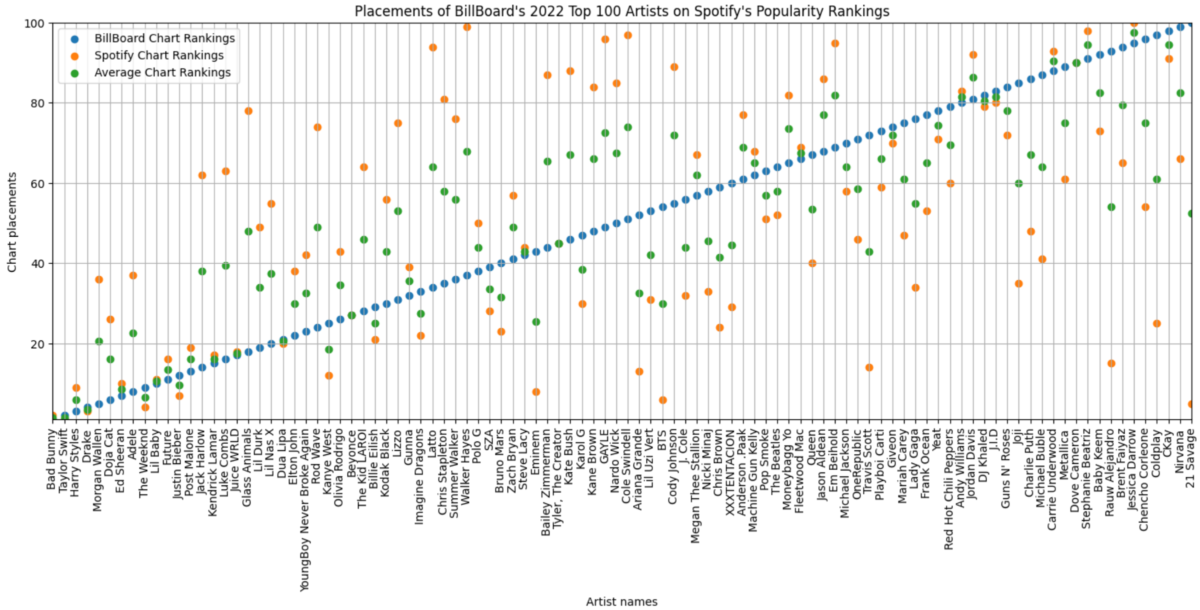
```
Jason Aldean: 77.0
Em Beihold: 82.0
Michael Jackson: 64.0
OneRepublic: 58.5
Travis Scott: 43.0
Playboi Carti: 66.0
Giveon: 72.0
Mariah Carey: 61.0
Lady Gaga: 55.0
Frank Ocean: 65.0
Yeat: 74.5
Red Hot Chili Peppers: 69.5
Andy Williams: 81.5
Jordan Davis: 86.5
DJ Khaled: 80.5
J.I.D: 81.5
Guns N' Roses: 78.0
Joji: 60.0
Charlie Puth: 67.0
Michael Buble: 64.0
Carrie Underwood: 90.5
Metallica: 75.0
Dove Cameron: 90.0
Stephanie Beatriz: 94.5
```

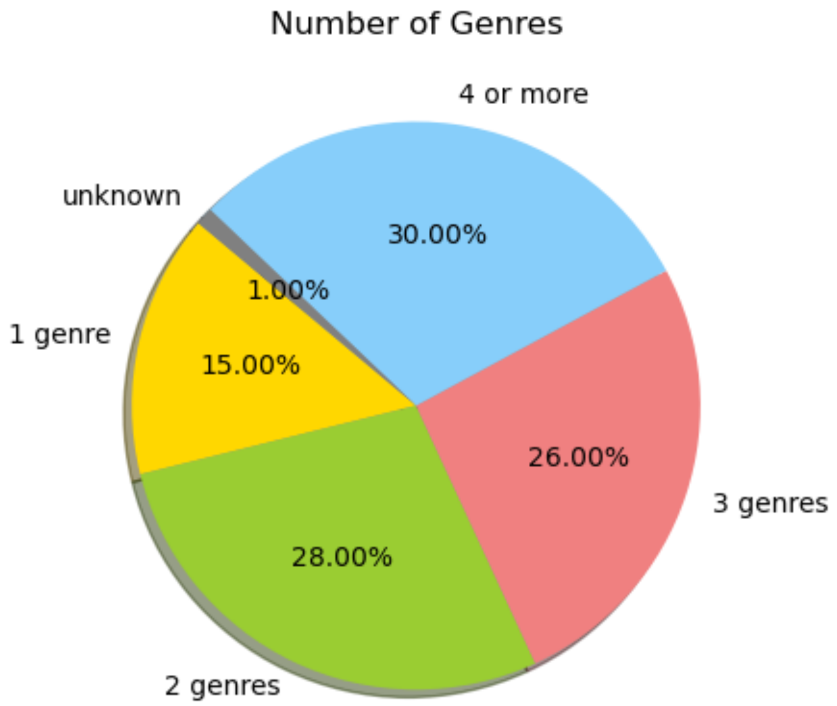


```
Charlie Puth: 67.0
Michael Buble: 64.0
Carrie Underwood: 90.5
Metallica: 75.0
Dove Cameron: 90.0
Stephanie Beatriz: 94.5
Baby Keem: 82.5
Rauw Alejandro: 54.0
Brent Faiyaz: 79.5
Jessica Darrow: 97.5
Chencho Corleone: 75.0
Coldplay: 61.0
CKay: 94.5
Nirvana: 82.5
21 Savage: 52.5

Average number of genres of all artists' discographies: 3.93
```

## 2. Visualization





3. Instructions of running the code

1. Unzip the file to whichever file folder on your computer you would have the easiest time accessing.
2. In the event that you must create new tables in a new database other than the one attached, replace ‘project.db’ in API.py with your database name and run the entire python file in your terminal. This will also write calculations to a text file and generate our graphs
3. The data visualizations only open one at a time, so when you are done viewing or saving the first graph, close the graph’s window to allow the second one to appear

**4. Functions**

- authentication()
- get_artist_info(access_token, ID)
- open_database(db_name)
- spotify_table1(artist_list, cur, conn)
- spotify_table2(artist_list, cur, conn)
- billboard_table(link, cur, conn)
- get_genre_average(cur, conn)
- get_rank_averages(cur, conn)

- write_averages(avg_genre_num, diff_dict, file)
- graph_ranks(cur, conn)
- genre_percentage(cur, conn)

| Function | Description | Input | Output |
|---|---|---|---|
| authentication() : | To get the access token to Spotify API | None | str: access_token |
| get_artist_info(access_token, ID) | To get certain artist's name, genres and popularity info on Spotify by entering their ID. | access_token: string ; ID: str | tuple: (ID, name, genres, popularity) |
| open_database(db_name) | To open the database named db_name and return cur, conn. | db_name: string, database name. | cur, conn |
| spotify_table1(artist_list, cur, conn) | To create a table of genres of each artist. The table contains 3 columns: id(the order of artists), genres and the number of genres. | artist_list: a list of artists' information tuples including genres and the number of genres;conn: connect; cur: cursor | None |
| spotify_table2(artist_list, cur, conn) | To create a table of artists' popularity. The table includes id (the order of artists), popularity, popularity ranking | artist_list: a list of artists' information tuples including genres, the number of genres & popularity; conn: connect; cur: cursor | None |
| billboard_table(link, cur, conn) | To create a table of artists' billboard ranking. The table includes id, name and rank. | link: the link of billboard website; conn: connect; cur: cursor | None |

| Function | Description | Input | Output |
|---|---|---|---|
| get_genre_average(cur, conn) | To calculate the average number of genres 100 popular artists have | cur, conn | the average number of genres |
| get_rank_averages(cur, conn) | To calculate the average of billboard rank and spotify rank for each artist | cur, conn | diff_dict: dict, keys: each artist's name; values: each artists' average rank |
| write_averages(avg_genre_num, diff_dict, file) | To write the genre and the rank average data into a file | avg_genre_num: the average number of genres; diff_dict: the dict of average ranks; file: string, a filename. | None |
| graph_ranks(cur, conn) | To plot average ranking data as a scatterpoint graph to show the potential relationship between two rankings. | cur, conn | None |
| genre_percentage(cur, conn) | To plot the number of genres as a pie chart to show the distribution of genre numbers. | cur, conn | None |