

SPOTIFY TOP 50 ASEAN SONG CHARTS AUGUST 2022 ANALYSIS

BY : DIMAS BRAHMANTYO

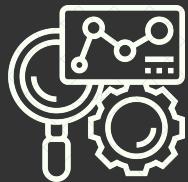
ANALYSIS BACKGROUND

Throughout history, music has been regarded as an expression of motion, tension, human character, identity, beauty, religious beliefs, and social conditions. However, the most common hypothesis is that listeners perceive music as an expression of emotion (Gabrielsson and Juslin, 2003). Music can evoke strong emotional responses such as chills and sensations in listeners. Positive emotions dominate the musical experience. Pleasant music can cause the release of reward-related neurotransmitters, such as dopamine. Listening to music is an easy way to change your mood or relieve stress. A song can give a different impression to the listener. It can be influenced by the genre, chart position, BPM (Beats per Minute), energy ratio, speechiness, loudness, acousticness, world popularity, and valence ratio.

Based on the existing problems, the author analyzes the Spotify Top 50 Music Charts in ASEAN countries in August 2022. using Python.



ANALYSIS PURPOSE



SEE THE CORRELATION
BETWEEN EACH DATA
VARIABLE



SEE DISTRIBUTION OF
ARTISTS, SONG GENRES,
AND SONG YEARS



COMPARE BETWEEN
GENRES OF SONGS ON
CHARTS WITH THE SONG
ADDITIONAL ATTRIBUTES

LIBRARY USED

- **Pandas** is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data.
- **Pandasql** allows you to query pandas DataFrames using SQL syntax. It works similarly to sqldf in R.
- **NumPy**, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed.
- **Matplotlib** is a comprehensive Python library for creating static, animated, and interactive visualizations.
- **Seaborn** has a similar function to Matplotlib. It provides a higher-level interface for drawing attractive and informative statistical graphics than Matplotlib.
- **Glob** is a Python library that finds all the pathnames matching a specified pattern according to the rules used by the Unix shell, although results are returned in arbitrary order.
- **Warnings**, Warning messages are typically issued in situations where it is useful to alert the user of some condition in a program, where that condition (normally) doesn't warrant raising an exception and terminating the program.

DATA PREPARATION

The dataset used is divided into 2; data1, which contains analytical data from the song, and data2 has data related to singers on the chart. The two datasets are combined into 1 using union in Pandasql.

```
In [5]: #Select the data
data = ps.read_csv('data.csv')
select data1.songid as songid, data2.singerid as singerid, data1.song_title as song_title, data1.song_singer as singer,
data2.singer_origin as singer_origin, data2.singer_based_from as singer_based_from, data1.song_album as album,
data1.song_year as year, data1.genre as genre, data1.song_position as position, data1.song_country as country,
data1.song.bpm as BPM, data1.song_duration as duration, data1.danceability as danceability, data1.energy as energy,
data1.key as key, data1.loudness as loudness, data1.mode as mode, data1.speechiness as speechiness,
data1.acousticness as acousticness, data1.instrumentalness as instrumentalness, data1.liveness as liveness,
data1.valence as valence, data1.popularity as popularity
from data1 left join data2 using(song_singer)
union all
select data1.songid as songid, data2.singerid as singerid, data1.song_title as song_title, data1.song_singer as singer,
data2.singer_origin as singer_origin, data2.singer_based_from as singer_based_from,
data1.song_album as album, data1.song_year as year, data1.genre as genre, data1.song_position as position,
data1.song_country as country, data1.song.bpm as BPM, data1.song_duration as duration, data1.danceability as danceability,
data1.energy as energy, data1.key as key, data1.loudness as loudness, data1.mode as mode, data1.speechiness as speechiness,
data1.acousticness as acousticness, data1.instrumentalness as instrumentalness, data1.liveness as liveness,
data1.valence as valence, data1.popularity as popularity
from data2 left join data1 using(song_singer)
where data1.song_singer is null
"""
#Show selected data
data
```

Out[5]:	songid	singerid	song_title	singer	singer_origin	singer_based_from	album	year	genre	p
0	THB3QUMUkZDq60XvQ9n8W	2ip3e49EEHvGg8eJ46L	Gulliver	Kang Seung-Yoon	Foreign Artist(s)	South Korea	Street Man Fighter (SMF) Original Vol.2 (OST)	2022	K-Pop	
1	TqZLU4gY1NWqX7mPB0	6eUKZXkX0qvHOKu9w2n3V	Shape of You	Ed Sheeran	Foreign Artist(s)	England	-	2017	Pop	
2	7BKLCZtjBu8VqR2FViTvW	69008x4162TqOewJ05jP	Closer (feat. Halsey)	The Chainsmokers	Foreign Artist(s)	USA	Collage	2016	Electropop	
3	0qVxDI68FyKsA1z0VMd4v	6eUKZXkX0qvHOKu9w2n3V	Perfect	Ed Sheeran	Foreign Artist(s)	England	-	2017	Pop	
4	2JxZz2UQj90f7ewp0hjs	137W8M8RPWhq8mrBG08FSop	See You Again (feat. Charlie Puth)	Wiz Khalifa	Foreign Artist(s)	USA	Furious 7: Original Motion Picture Soundtrack	2015	Pop Rap	

MISSING VALUES CHECKING

Missing Data can occur when no information is provided for one or more items or for a whole unit. Missing Data is a very big problem in a real-life scenario. Therefore, we need to check whether there is a missing value and delete the missing value if any. By In the dataset used, no missing value was found so that a query could be performed for analysis.

```
In [6]: #Check if there are missing data values  
data.isna().sum()  
  
out[6]: songid          0  
singerid         0  
song_title       0  
singer           0  
singer_origin    0  
singer_based_from 0  
album            0  
year             0  
genre            0  
position          0  
country          0  
BPM              0  
duration         0  
danceability     0  
energy            0  
key               0  
loudness          0  
mode              0  
speechiness       0  
acousticness      0  
instrumentalness 0  
liveness          0  
valence           0  
popularity        0  
dtype: Int64
```

INTRODUCTION INTO DATA

To see the number of songs and the number of singers in the Top 50 Music Chart in ASEAN countries, we use the distinct count in pandasql.

As seen in the picture above, there are a total of 349 songs from 245 different artists on the Top 50 Music Charts of ASEAN countries.

```
In [9]: ps.sql("select count (distinct songId) as song_unique_count from data1")
Out[9]:   song_unique_count
          0           349

In [10]: ps.sql("select count (distinct singerId) as singer_unique_count from data2")
Out[10]:   singer_unique_count
          0           245
```

CORRELATION ANALYSIS

```
In [11]: #Plotting Correlation Map Data  
data_corr = data.corr()  
plt.figure(figsize=(15,15))  
  
#Make the Correlation Map using Seaborn  
data_corr_map = sns.heatmap(data_corr, vmin = -0.8, vmax = 0.8, cmap = 'RdYlGn', linewidth = 0.4, annot = True)
```



Correlation Map Analysis:

- Valence has a positive correlation with danceability and tracks energy.
- The biggest positive correlation was found in the relationship between energy and loudness of the track (0.76).
- The most significant negative correlation was found in the relationship between energy and acousticness of the track (-0.69).



ARTISTS AND GENRES ANALYSIS

A. Top 10 artists on the ASEAN Charts

```
In [5]: #Query Data
data['singer_obj'] = data['singerid'].astype("str")
top10_artist = data.groupby(['singer','genre','singer_origin','singer_based_from'],as_index=False)[['singer_obj']].count()
data_top10_artist = top10_artist.nlargest(10,'singer_obj')

#Determine the size of the plot chart image
plt.subplots(figsize=(40,24))

#Then, we define axis chart variables
x = data_top10_artist['singer']
y = data_top10_artist['singer_obj']

#Plotting the chart
plt.barh(x, y, height = 0.9, color = ['#6017c1','#6017c1','#7b0d7f','#92195c','#87433d','#87433d','#826421','#8d7228',
    '#8e8733','#4b9e3f','#17c150'])

# Add annotation to bars
for index, value in enumerate(y):
    plt.text(value, index, str(value), fontsize = 20, fontweight="bold")

#Insert Title, X-axis and Y-axis label
plt.title('Singer with Most Songs on ASEAN Music Charts\n(in August 2022', fontname = 'Times New Roman',
          fontsize = 40, fontweight="bold")
plt.xlabel('\nTotal Songs', fontsize = 24, fontweight="bold")
plt.ylabel('\nArtist Name', fontsize = 24, fontweight="bold")

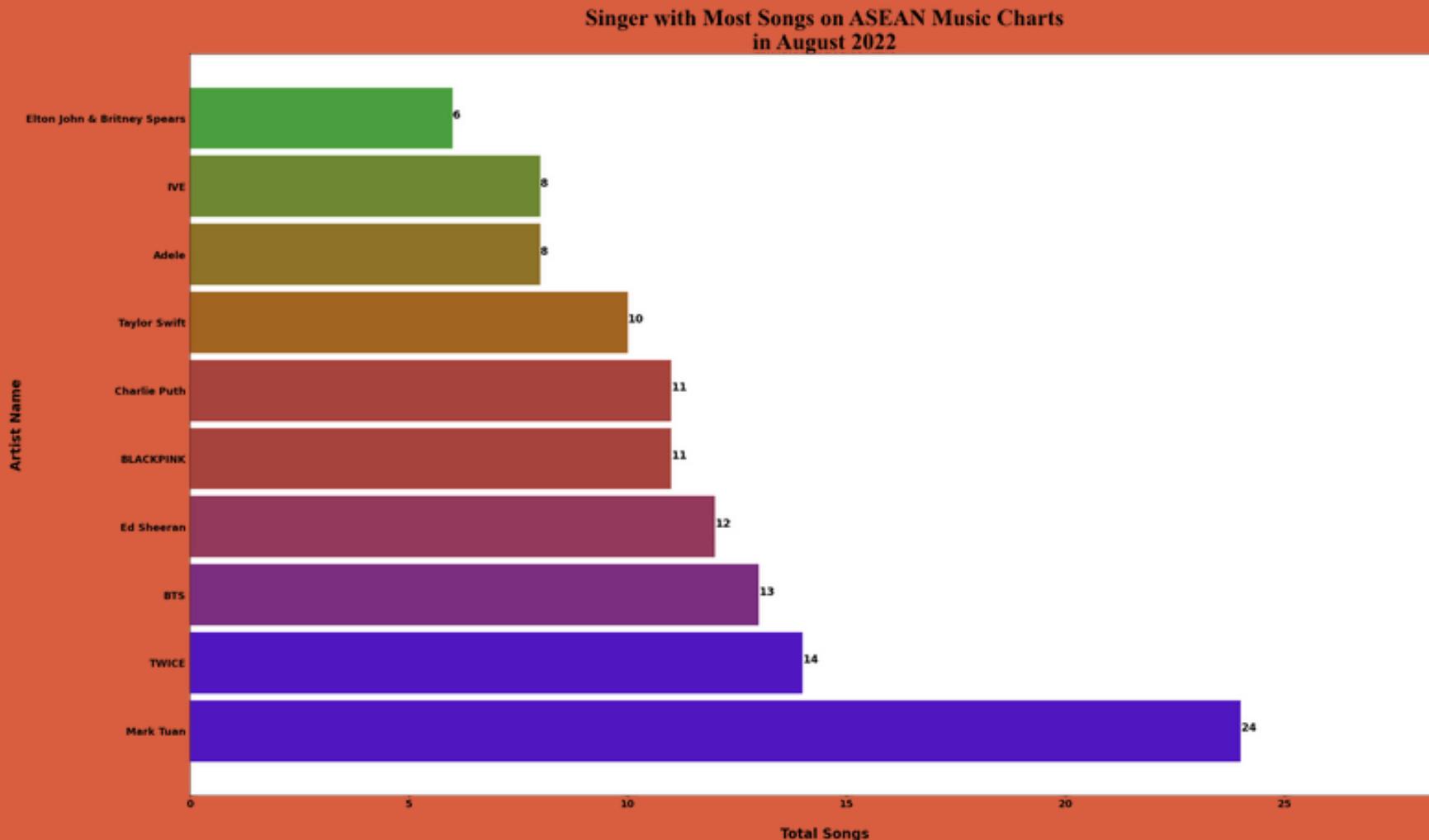
#Manage the X-ticks and Y-ticks
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=18, fontweight="bold")
plt.tick_params(left = False)

#Limit the X-axis
plt.xlim(xmin = 0, xmax = 20)

#Show the chart
plt.show()
```

ARTISTS AND GENRES ANALYSIS

A. Top 10 artists on the ASEAN Charts





ARTISTS AND GENRES ANALYSIS

A. Top 10 artists on the ASEAN Charts

Out[14]:

	singer	genre	singer_origin	singer_based_from	singer_obj
151	Mark Tuan	Pop	Foreign Artist(s)	USA	24
211	TWICE	K-Pop	Foreign Artist(s)	South Korea	14
24	BTS	K-Pop	Foreign Artist(s)	South Korea	13
60	Ed Sheeran	Pop	Foreign Artist(s)	England	12
22	BLACKPINK	K-Pop	Foreign Artist(s)	South Korea	11
39	Charlie Puth	Pop	Foreign Artist(s)	USA	11
214	Taylor Swift	Pop	Foreign Artist(s)	USA	10
5	Adele	RnB/Soul	Foreign Artist(s)	England	8
90	ME	K-Pop	Foreign Artist(s)	South Korea	8
63	Elton John & Britney Spears	Pop Rock	Foreign Artist(s)	England	6

Analysis:

- Mark Tuan became the singer with the most songs on the Charts with a total of 24 songs.
- All the singers who are included in the top 10 artists are from outside ASEAN, and 80% are from Pop and K-Pop genres.



ARTISTS AND GENRES ANALYSIS

B. Comparison of artists from ASEAN countries and from foreign countries

```
In [6]: #Query Data
data['singer_origin_obj'] = data['singer_origin'].astype('str')
singer_origin_data = data.groupby(['singer_origin'], as_index = False)[['singer_origin_obj']].count()

#Set the figure plot size
plt.figure(figsize=(20,10))

# create plot
myexplode=[0.1,0]
plt.pie(singer_origin_data['singer_origin_obj'],labels = singer_origin_data['singer_origin'],
        autopct='%.1f%%', explode=myexplode ,startangle=90, textprops={'fontsize': 20,'color':'w'}, labeldistance=None)

# set label & title
plt.title('Comparison of Artists from ASEAN Countries and Foreign Countries\\non ASEAN Song Charts August 2022\\n',
          fontname = 'Times New Roman', fontsize = 20, fontweight="bold")

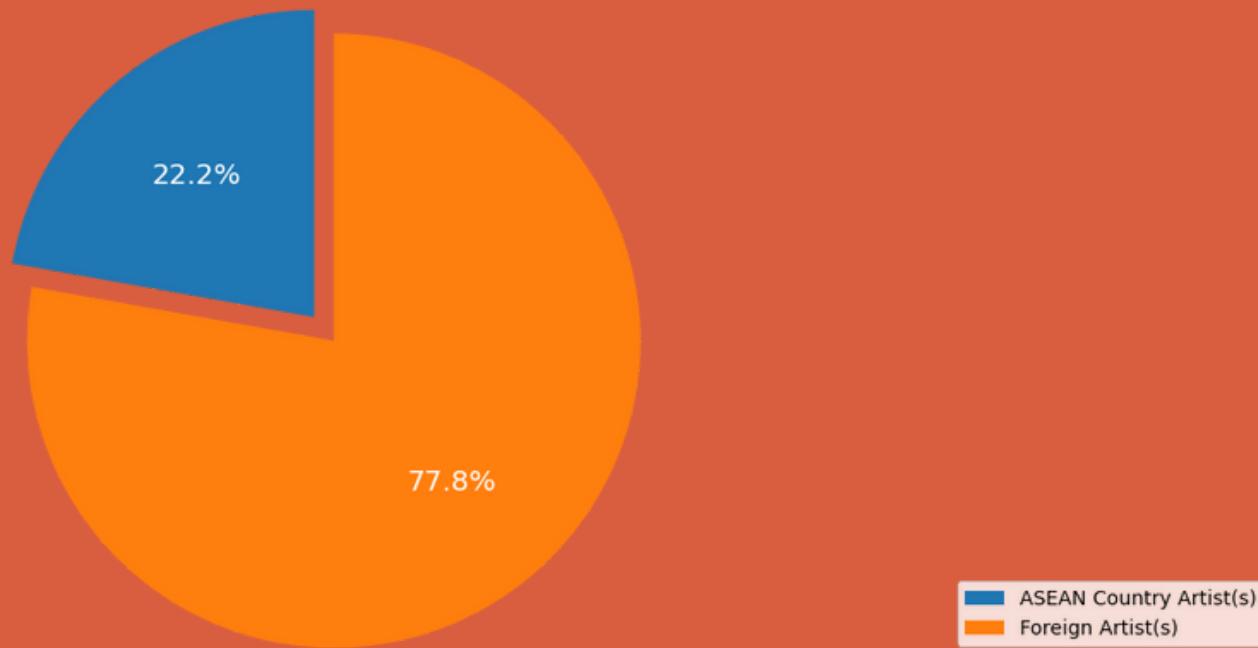
# add legend
plt.legend(bbox_to_anchor=(1.3,0.2), prop={'size': 14})

#Show the plot chart data
plt.show()
```



ARTISTS AND GENRES ANALYSIS

B. Comparison of artists from ASEAN countries and from foreign countries



Analysis:

Foreign Artist(s) dominate the number of artists origin by 77.8%, while ASEAN Countries Artist(s) make up 22.2% of the total artists in the data.



ARTISTS AND GENRES ANALYSIS

C. Top 10 most famous songs on the ASEAN charts based on song distribution

```
In [20]: top10_songs = data.groupby(['song_title', 'singer', 'genre'])['songid'].count()
top10_songs.sort_values(ascending=False).head(10)
```

```
Out[20]:   song_title           singer          genre  count
0  Pink Venom        BLACKPINK      K-Pop       9
1  After LIKE            IVE      K-Pop       7
2  Hold Me Closer  Elton John & Britney Spears  Pop Rock       6
3  Talk that Talk           TWICE      K-Pop       6
4  Until I Found You    Stephen Sanchez  Alternative/Indie       6
5  Perfect             Ed Sheeran        Pop       5
6  FOREVER 1        Girls' Generation      K-Pop       5
7  Closer (feat. Halsey)  The Chainsmokers  Electropop       5
8  I Ain't Worried      OneRepublic  Pop Rock       5
9  When I Was Your Man        Bruno Mars        Pop       4
Name: songid, dtype: int64
```

Analysis:

Blackpink's "Pink Venom" became the most famous song on ASEAN Music Charts in August 2022, distributed in 9 ASEAN countries.

ARTISTS AND GENRES ANALYSIS

D. Distribution of the tracks by song release year

```
In [22]: #Query Data
data['year_obj']=data['year'].astype('int')
df_year = data.groupby(['year'],as_index=False)[['year_obj']].count()

#size figure chart
plt.figure(figsize=(16,16))

#Input axis variable
x = df_year['year']
y = df_year['year_obj']

#Plotting the chart
plot2 = plt.plot(x,y)

#Axis Chart Setting
plt.yticks([1, 10, 40, 90, 120, 150, 180, 210, 240],['1','10','40','90','120','150','180','210','240'],
           fontweight="bold", fontsize = 14)
plt.xticks([1969, 1981, 1992, 2001, 2002, 2003, 2008, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022],
           ['1969', '1981', '1992', '2001', '2002', '2003', '2008', '2011', '2012', '2013', '2014', '2015', '2016', '2017',
            '2018', '2019', '2020', '2021', '2022'], rotation=90, fontweight="bold", fontsize = 14)
plt.ylim(ymin=0, ymax=250)
plt.tick_params(bottom = False)

#Set the color, shape, and size of lines and markers
plt.setp(plot2, color='steelblue', linestyle='--', linewidth=3, marker='o', markersize=8, markerfacecolor='firebrick')

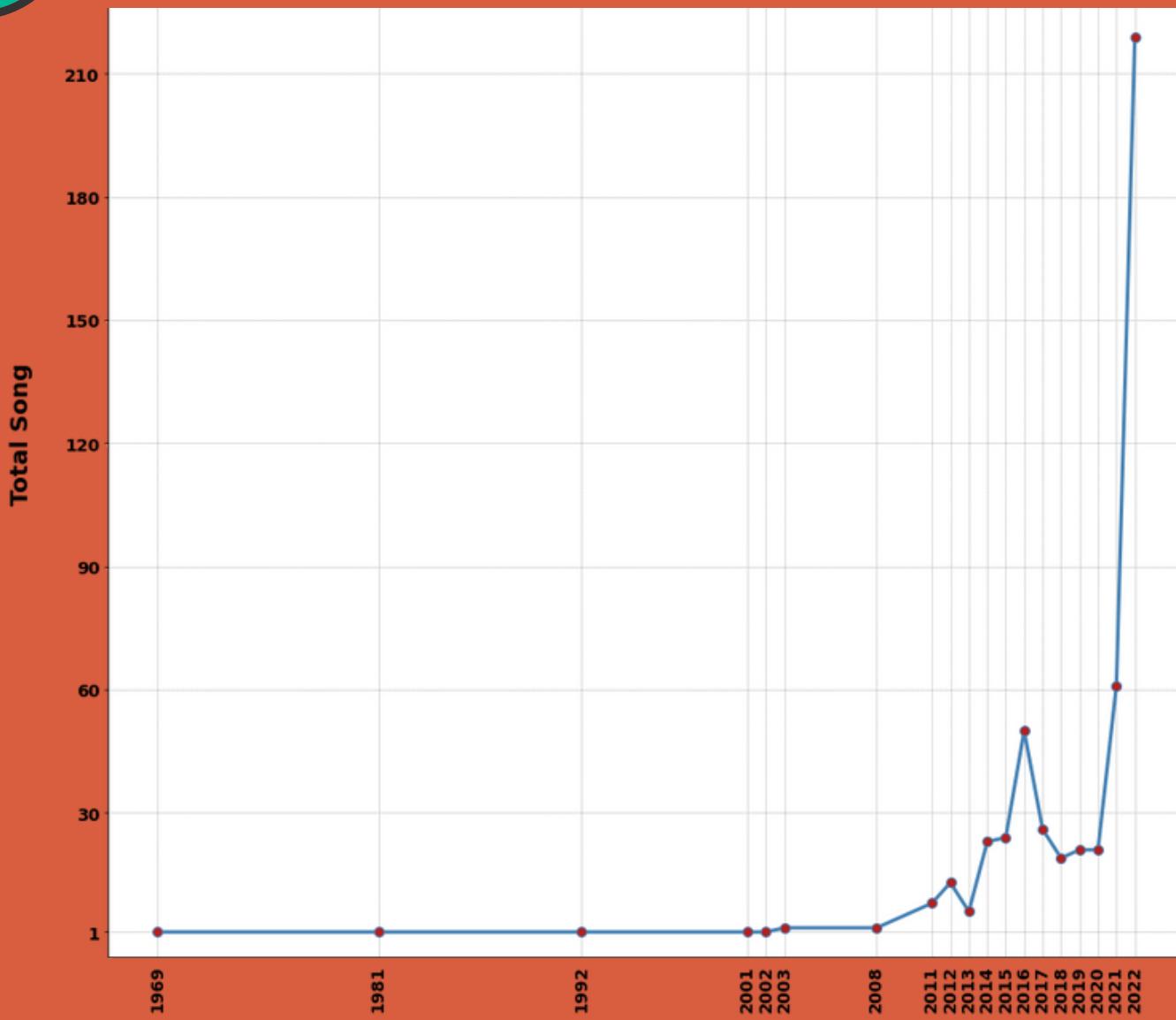
#Add Grid
plt.grid(color='grey', linestyle=':', linewidth=0.75)

#Add Title, X-axis and Y-axis Label
plt.title('Spread of Number of Songs on ASIAN Charts for August 2022\n(By Year)\n', fontsize = 20, fontname = 'Times New Roman',
           fontweight = 'bold')
plt.xlabel('Year', fontsize = 20, fontweight = 'bold')
plt.ylabel('Total Song\n', fontsize = 20, fontweight = 'bold')

#Show chart
plt.show()
```

ARTISTS AND GENRES ANALYSIS

D. Distribution of the tracks by song release year



Analysis:

The tracks on the ASEAN Song Charts August 2022 are from 1969 to 2022, with the highest number of songs dating from 2022, as many as 219 tracks.



ARTISTS AND GENRES ANALYSIS

E. Composition of the number of songs by genre

```
In [19]: #The distribution of the types of genres on the ASEAN song charts in August 2022
data['genre_obj'] = data['genre']
data_top = data.groupby(['genre'],as_index=False)[['genre_obj']].count()

#resize figure chart
plt.figure(figsize=(10,25))

#input axis variable
x = data_top['genre']
y = data_top['genre_obj']

#Plotting the chart
plot3 = plt.bar(x,y, width = 0.65, color = ['#199999','#198943','#196f5d','#195f6e','#194d89','#1941bd','#193995','#193691',
                                              '#19338f','#462e68','#632783','#702180','#752076','#951978','#95126f','#951262',
                                              '#953c53','#954f40','#957419'])
plt.bar_label(plot3, label_type = "edge", fontsize = 25, fontweight = "bold")

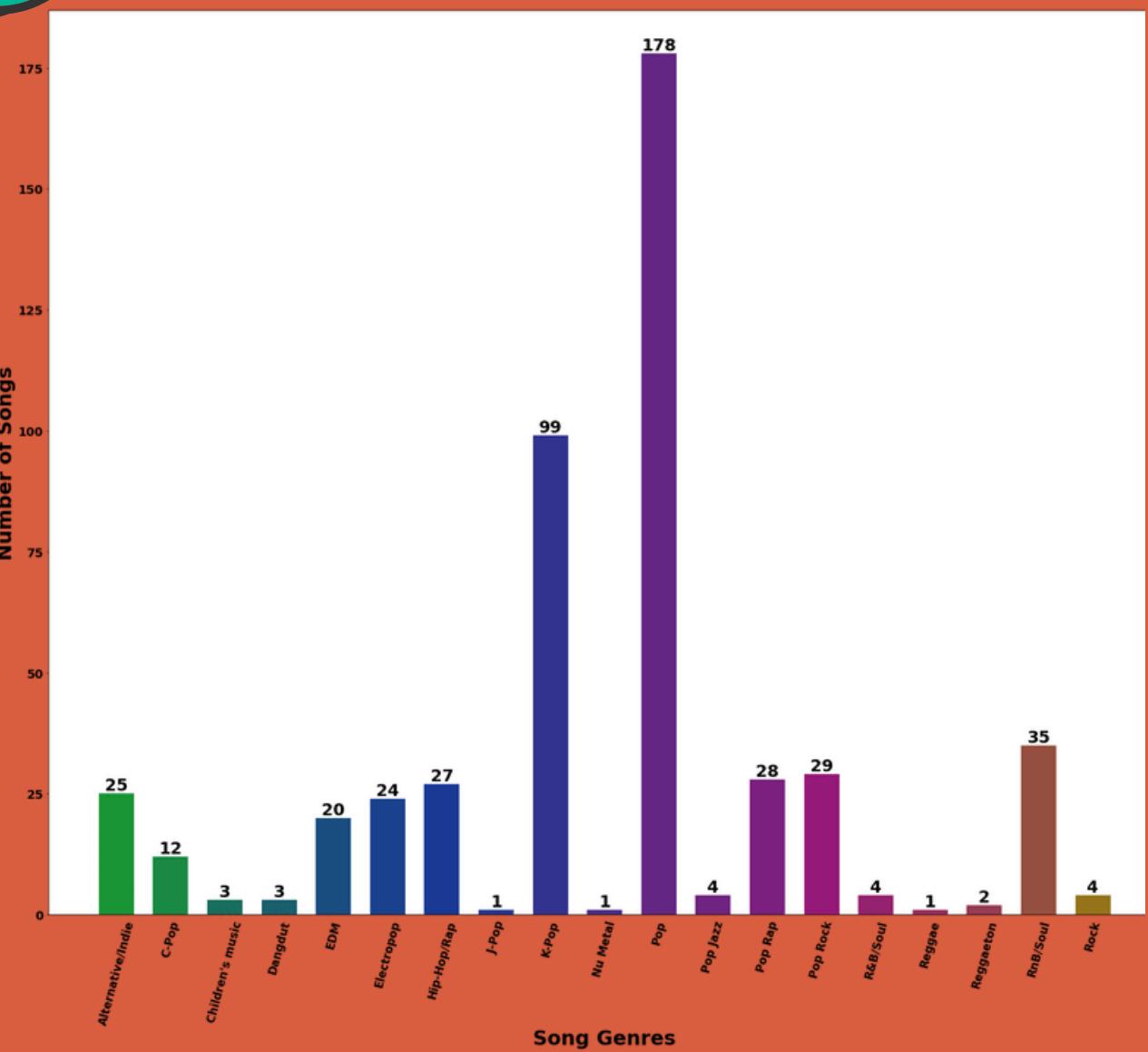
#Insert Title, X-axis and Y-axis label
plt.title('Comparison Graph of the Number of Songs on the ASEAN Charts in August 2022\nby Music Genre\n',
          fontname = 'Times New Roman', fontsize = 40, fontweight="bold")
plt.xlabel('Song Genres', fontsize = 30, fontweight="bold")
plt.ylabel("Number of Songs", fontsize = 30, fontweight="bold")

#Manage the X-ticks and Y-ticks
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

#Show the Chart
plt.show()
```

ARTISTS AND GENRES ANALYSIS

E. Composition of the number of songs by genre



Analysis:

- There are 19 genre tracks on the ASEAN song charts in August 2022.
- Pop and K-Pop tracks have the most significant number among other genres, where these two genres make up 55% of the total number of songs.
- J-Pop, Nu Metal, and Reggae become the genre with the least number of songs, with 1 song each.

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

a. Track Genre with Average Track Position on Chart

```
In [7]: #Query Data
data_genre_position = data.groupby(['genre'], as_index = False)[['position']].mean()
data_genre_position = data_genre_position.drop([2,3,7,8,11,14,15,16,18])

#Resize figure chart
plt.figure(figsize=(30,25))

#Input axis variable
x = data_genre_position['genre']
y = data_genre_position['position']

#visualize the data with Bar Chart
data_genre_position_plot = plt.bar(x,y, color = ['#db2a22','#db4c22','#b37dib','#e9719','#9ec28','#329964','#8b9c61',
                                                '#174672','#3d8ef9','#6000a70'])
plt.bar_label(data_genre_position_plot, label_type = 'edge', fontsize = 26, fmt='%.2f', fontweight = 'bold')

#Add Title, X-axis and Y-axis Label
plt.title('\nComparison Graph of Song Genres on ASEAN Charts in August 2022\nBased on Average Song Positions on Charts\n', 
          fontsize = 40, fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel("\nSong Genre", fontsize = 26, fontweight = 'bold')
plt.ylabel("Average Song Position on charts\n", fontsize = 26, fontweight = 'bold')

#Axis Chart Settings
plt.ylim(ymax = 40)
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

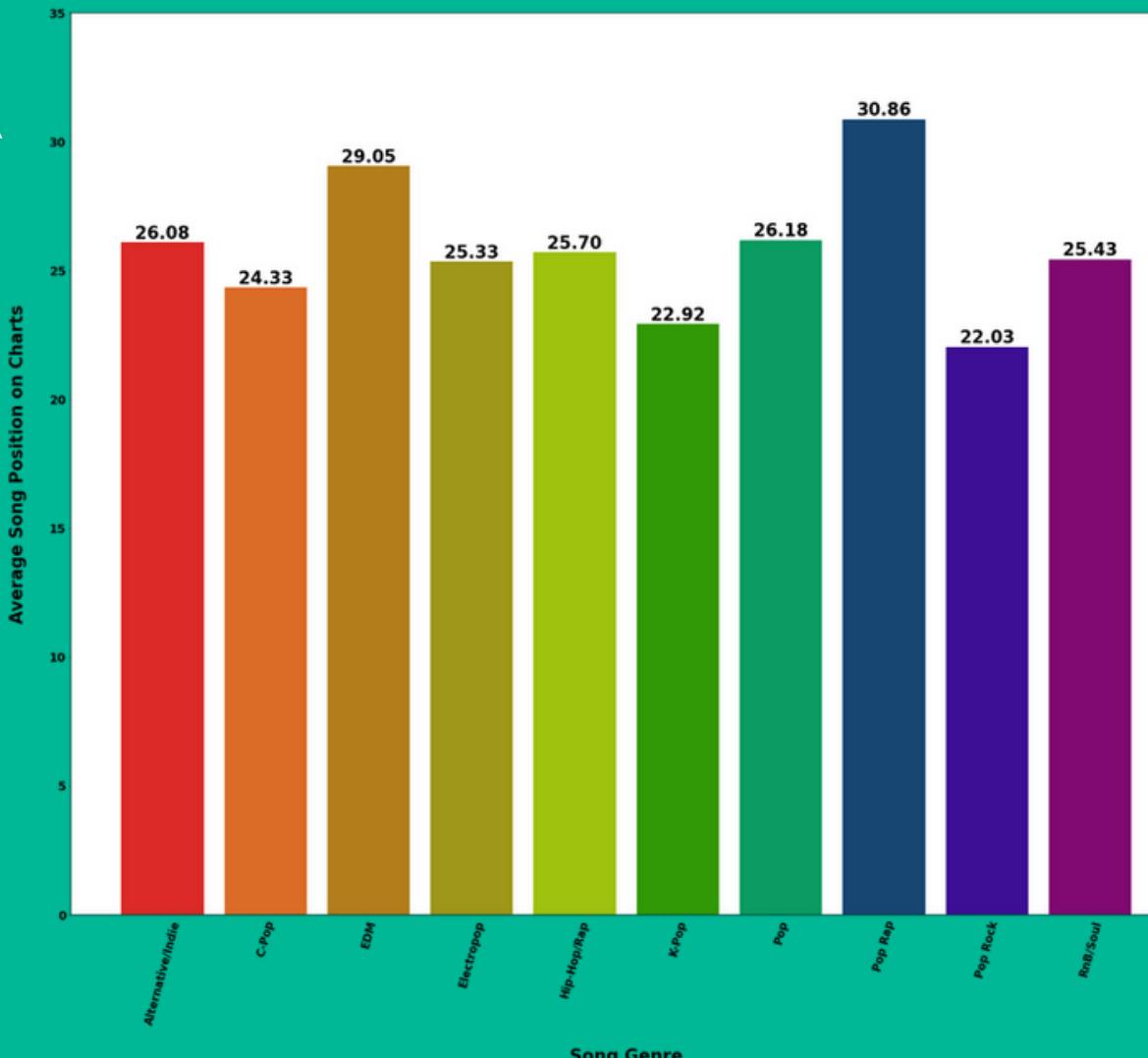
#Show the chart
plt.show()
```

Activate

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

a. Track Genre with Average Track Position on Chart

Comparison Graph of Song Genres on ASEAN Charts in August 2022
Based on Average Song Positions on Charts



Analysis:

Tracks with the Pop Rock genre have the highest average chart ranking, with an average of being at position 22.03 out of 50. In contrast, tracks with the Pop Rap genre have the lowest average chart ranking, 30.86 out of 50.

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

b. Track Genre with Average Track Duration

```
In [10]: #Data Query
data_genre_duration = data.groupby(['genre'], as_index = False)[['duration']].mean()
data_genre_duration = data_genre_duration.drop([3,5,7,8,11,14,15,16,18])

#Resize figure chart
plt.figure(figsize=(10,25))

#Input axis variable
x = data_genre_duration['genre']
y = data_genre_duration['duration']

#visualize the data with Bar chart
data_genre_duration_plot = plt.bar(x,y, width = 0.8, color = ['#db2a27','#dc6c27','#b37d1b','#e9719','#ec20e','#329996',
                           '#eb9c62','#174673','#3def94','#006a70'])
plt.bar_label(data_genre_duration_plot, label_type = 'edge', fontsize = 22, fmt='%.0f', fontweight = "bold")

#Add Title, X-axis and Y-axis Label
plt.title('Comparison Graph of Song Genres on ASEAN Charts in August 2022\nBased on Song Duration\n', fontsize = 36,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel('\nSong Genre', fontsize = 26, fontweight = 'bold')
plt.ylabel('Duration (ms)\n', fontsize = 26, fontweight = 'bold')

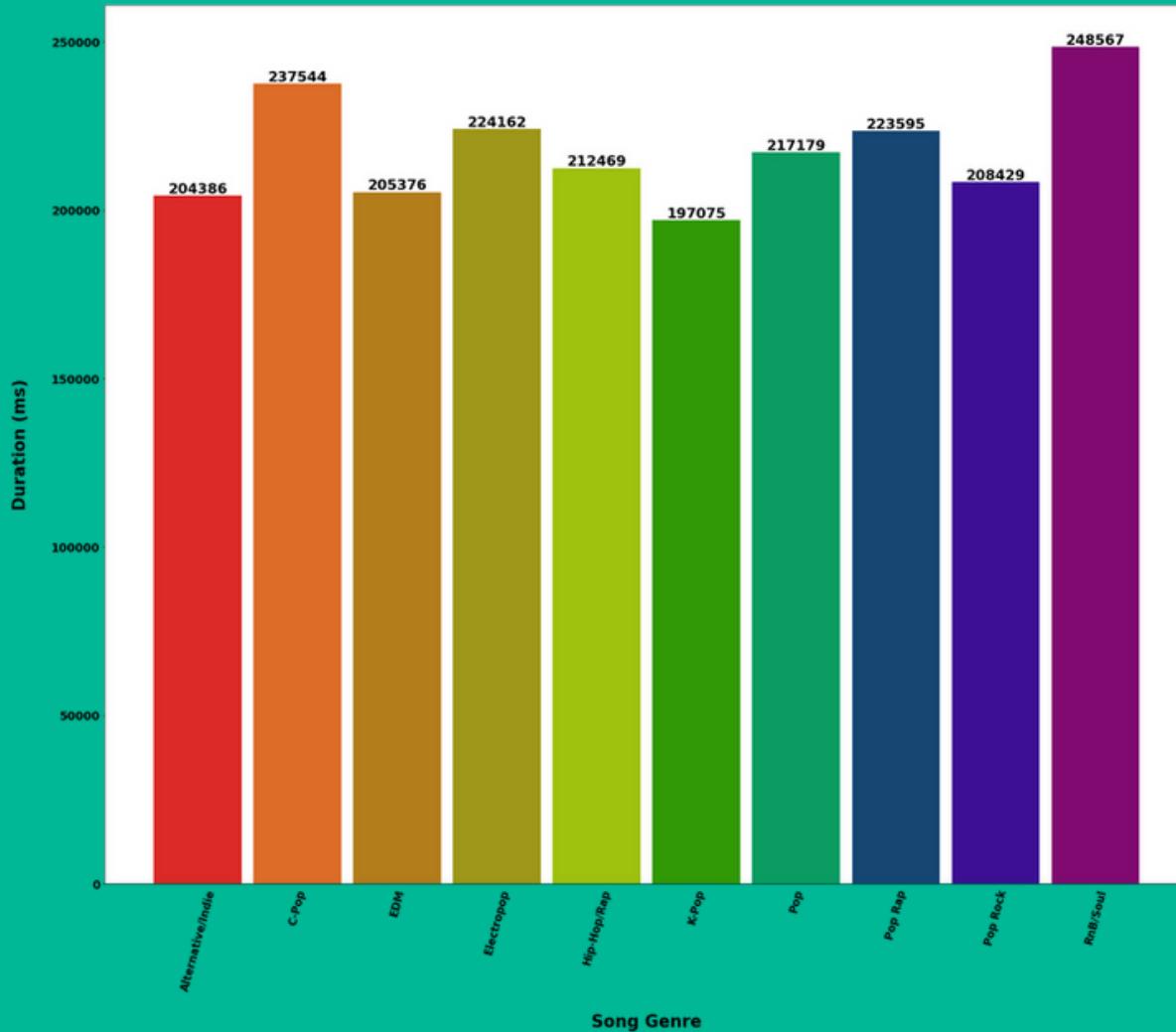
#Axis Chart Settings
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

#show the chart
plt.show()
```

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

b. Track Genre with Average Track Duration

Comparison Graph of Song Genres on ASEAN Charts in August 2022
Based on Song Duration



Analysis:

The RnB/Soul genre has the longest average track duration of 248567 ms, while K-Pop has the fastest average track duration of 197075 ms.

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

c. Track Genre with Average BPM

```
In [34]: #Query Data
data_genre_bpm = data.groupby(['genre'], as_index = False)[['BPM']].mean()
data_genre_bpm = data_genre_bpm.drop([2,3,7,9,11,14,15,16,18])

#Resize figure chart
plt.figure(figsize=(30,25))

#Input axis variable
x = data_genre_bpm['genre']
y = data_genre_bpm['BPM']

#Visualize the data with Bar Chart
data_genre_bpm_plot = plt.bar(x,y, width = 0.8, color = ['#db2a27','#e66c37','#b37d1b','#9e9719','#9ec20e','#329996',
          '#e0b9c62','#174673','#3ddef54','#800ea76'])
plt.bar_label(data_genre_bpm_plot, label_type = 'edge', fontsize = 22, fmt='%.4f', fontweight = 'bold')

#Add Title, X-axis and Y-axis Label
plt.title('Comparison Graph of Song Genres on ASIAN charts in August 2022(based on song review)', fontsize = 30,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel('Song genre', fontsize = 20, fontweight = 'bold')
plt.ylabel('Beats per Minute(BPM)', fontsize = 20, fontweight = 'bold')

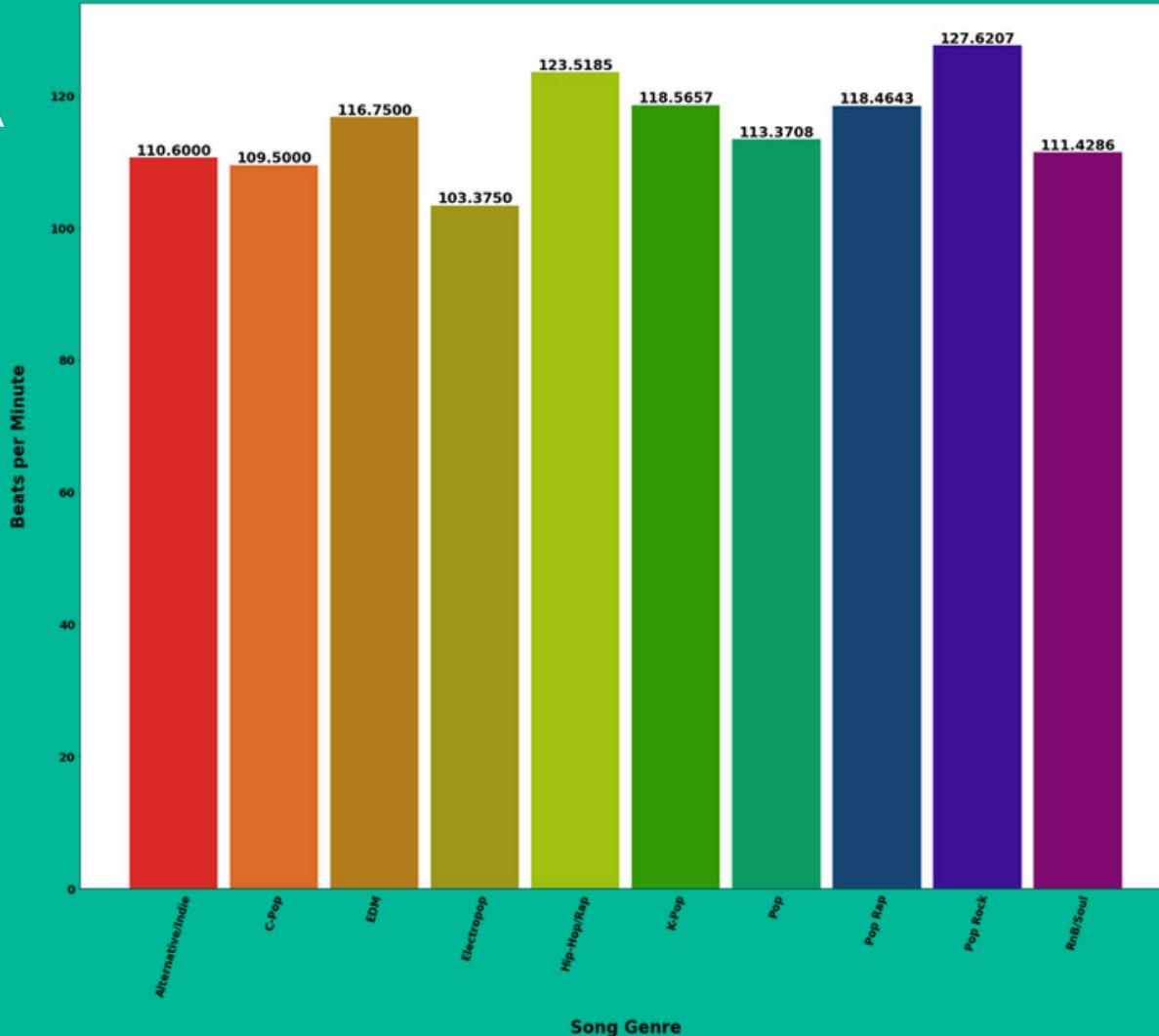
#Axis Chart Settings
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

#Show the Query Data Chart
plt.show()
```

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

c. Track Genre with Average BPM

Comparison Graph of Song Genres on ASEAN Charts in August 2022
Based on Song BPM



Analysis:

Pop Rock has a higher BPM than any other genre (127.6207), followed by Hip-hop/Rap (123.5185). Meanwhile, Electropop is the genre with the lowest BPM (103.375).

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

d. Track Genre with Average Track Energy Ratio

```
In [9]: #Query Data
data_genre_energy = data.groupby(['genre'], as_index = False)[['energy']].mean()
data_genre_energy = data_genre_energy.drop([1,3,7,9,11,14,15,16,18])

#resize figure chart
plt.figure(figsize=(30,25))

#Input axis variable
x = data_genre_energy['genre']
y = data_genre_energy['energy']

#visualize the data with bar chart
data_genre_energy_plot = plt.bar(x,y, color = ['#db2a27','#dc6c27','#b37dib','#9e9719','#9ec2be','#329906','#0b9c62',
                                              '#174473','#3d0f94','#000070'])
plt.bar_label(data_genre_energy.plot, label_type = 'edge', fontsize = 24, font='X.3f', fontweight = 'bold')

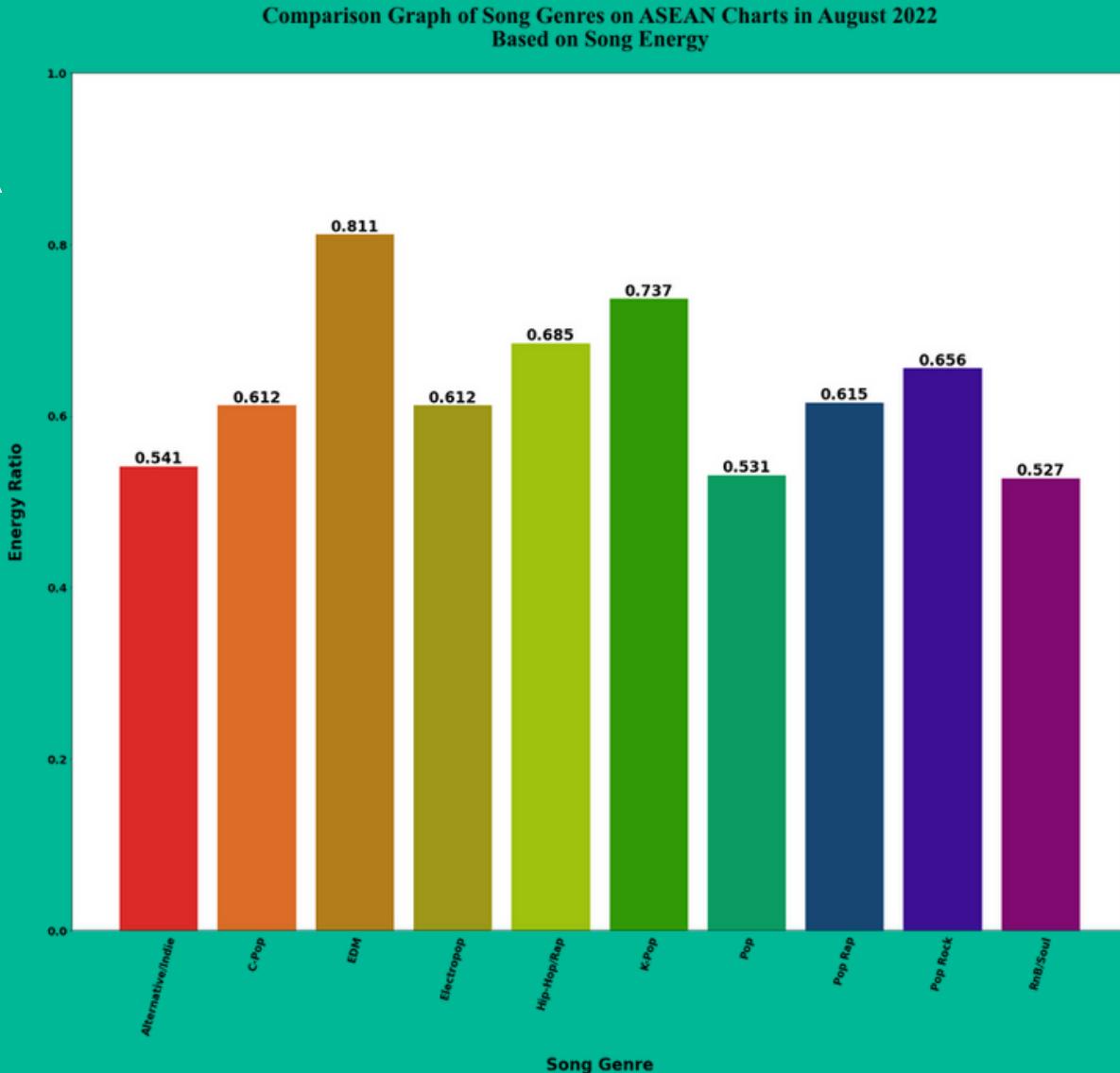
#Add Title, X-axis and Y-axis Label
plt.title('\nComparison Graph of Song Genres on ASEAN Charts in August 2022\nBased on Song Energy\n', fontsize = 36,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel('\nSong genre', fontsize = 26, fontweight = 'bold')
plt.ylabel('Energy Ratio\n', fontsize = 26, fontweight = 'bold')

#Axis Chart Settings
plt.ylim(ymax = 1)
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

#Show the Chart
plt.show()
```

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

d. Track Genre with Average Track Energy Ratio



Analysis:

EDM is the track genre with the highest average energy ratio of 0.811, followed by K-pop (0.737).

Meanwhile, RnB/Soul is the track genre with the smallest energy ratio (0.531).

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

e. Track Genre with Average Track Speechiness Ratio

```
In [10]: #Query Data
data_genre_speechiness = data.groupby(["genre"], as_index = False)[["speechiness"]].mean()
data_genre_speechiness = data_genre_speechiness.drop([2,3,7,9,11,14,15,16,18])

#Resize figure chart
plt.figure(figsize=(10,15))

#Input axis variable
x = data_genre_speechiness["genre"]
y = data_genre_speechiness["speechiness"]

#visualize the data with bar chart
data_genre_speechiness_plot = plt.bar(x,y, width = 0.8, color = ['#db2a27','#dc6c27','#b37d1b','#e9719','#ec20c','#329900',
                                                               '#8b9c62','#174673','#3daf94','#808a70'])

plt.bar_label(data_genre_speechiness_plot, label_type = 'edge', fontsize = 12, fmt='%.4f', fontweight = 'bold')

#Add Title, X-axis and Y-axis Label
plt.title('Comparison Graph of Song Genres on ASEAN Charts in August 2022\nBased on Song Speechiness\n', fontsize = 36,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel('Song Genre', fontsize = 26, fontweight = 'bold')
plt.ylabel('Speechiness Ratio\n', fontsize = 26, fontweight = 'bold')

#Baris Chart Settings
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

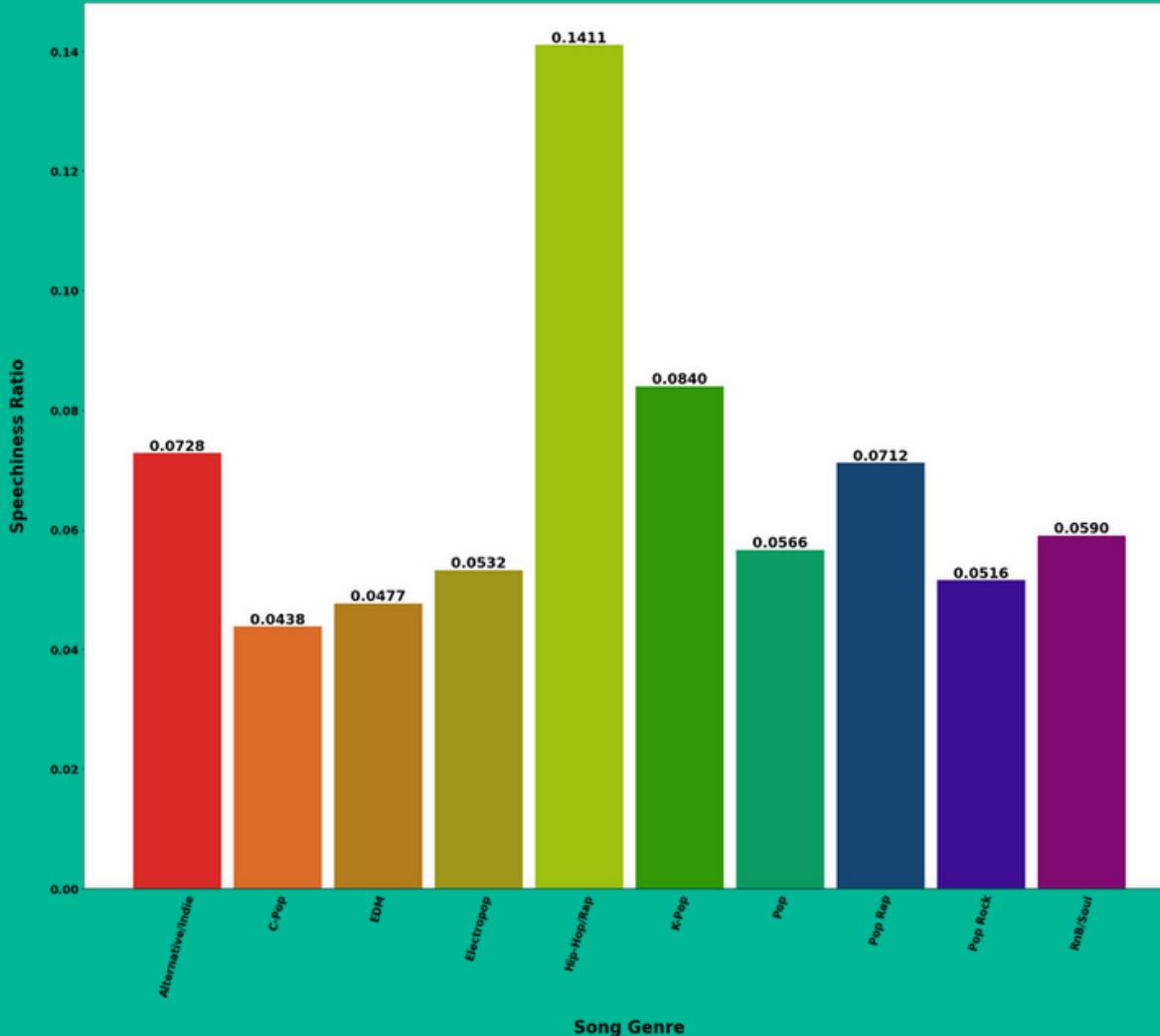
#show the query Data chart
plt.show()
```

Activ

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

e. Track Genre with Average Track Speechiness Ratio

Comparison Graph of Song Genres on ASEAN Charts in August 2022
Based on Song Speechiness



Analysis:

Hip-Hop has the highest speechiness ratio (0.1411) compared to other genres. Whereas, C-Pop has the lowest speechiness ratio (0.0438).

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

f. Track Genre with Average Loudness (dB)

```
In [36]: #Data Query
data_genre_loudness = data.groupby(['genre'], as_index = False)[['loudness']].mean()
data_genre_loudness = data_genre_loudness.drop([2,3,7,9,11,14,15,16,18])

#Resize figure chart
plt.figure(figsize=(20,25))

#Input axis variable
x = data_genre_loudness["genre"]
y = data_genre_loudness["loudness"]

#Visualize the data with Bar Chart
data_genre_loudness_plot = plt.bar(x,y, width = 0.8, color = ['#db2a27','#dc6c27','#b37d1b','#9e9719','#ec20e','#329900',
                '#8b9c62','#174673','#3d8f94','#800a70'])
plt.bar_label(data_genre_loudness_plot, label_type = 'edge', fontsize = 20, fmt='%.3f', fontweight = 'bold')

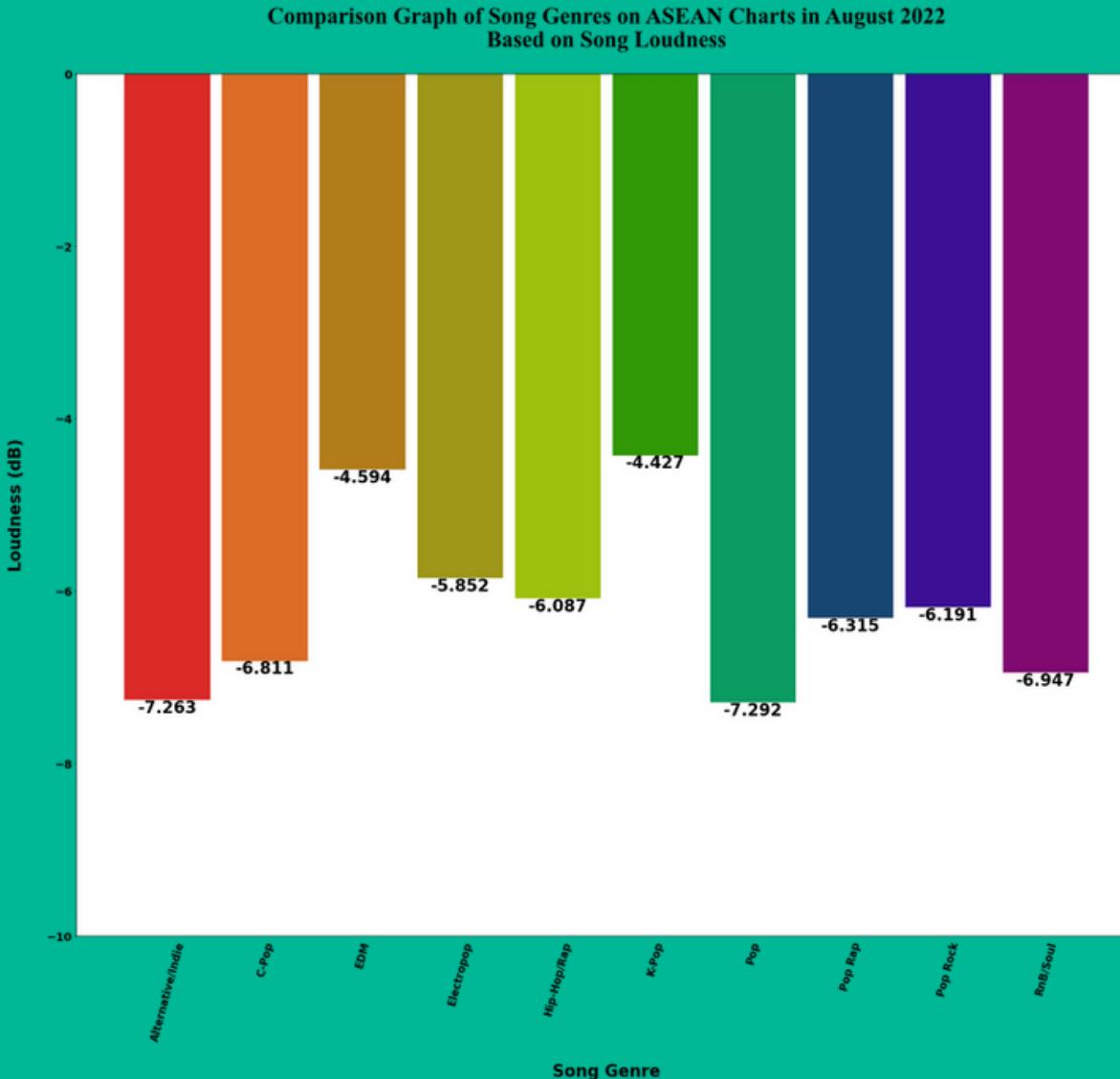
#Add Title, X-axis and Y-axis Label
plt.title("\nComparison Graph of Song Genres on ASEAN Charts in August 2022\nBased on Song Loudness\n", fontsize = 36,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel("Song Genre", fontsize = 26, fontweight = "bold")
plt.ylabel("Loudness (dB)\n", fontsize = 26, fontweight = 'bold')

#Axis Chart Settings
plt.ylim(ymax = 0, ymin = -10)
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

#show the Query Data Chart
plt.show()
```

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

f. Track Genre with Average Loudness (dB)



Analysis:

Pop genre tracks have the smallest loudness among other genres, followed by Alternative/Indie tracks, whereas K-pop tracks have the greatest loudness.

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

g. Track Genre with Average Acousticness Ratio

```
In [12]: #Query Data
data_genre_acousticness = data.groupby(['genre'], as_index = False)[['acousticness']].mean()
data_genre_acousticness = data_genre_acousticness.drop([2,3,7,9,11,14,15,16,18])

#Resize figure chart
plt.figure(figsize=(30,25))

#Input axis variable
x = data_genre_acousticness['genre']
y = data_genre_acousticness['acousticness']

#visualize the data with Bar Chart
data_genre_acousticness_plot = plt.bar(x,y, width = 0.8, color = ['#db2a27','#dc6c27','#b37d1b','#9e9719','#ec18e','#329998',
 '#0b9cc2','#174673','#3d0ef94','#000070'])

plt.bar_label(data_genre_acousticness_plot, label_type = 'edge', fontsize = 26, fmt='%.4f', fontweight = 'bold')

#Add Title, X-axis and Y-axis Label
plt.title('\nComparison Graph of Song Genres on ASEAN Charts in August 2022\nBased on Song Acousticness\n', fontsize = 36,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel('Song Genre', fontsize = 26, fontweight = 'bold')
plt.ylabel('Acousticness\n', fontsize = 26, fontweight = 'bold')

#Axis Chart Settings
plt.ylim(ymax = 1)
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

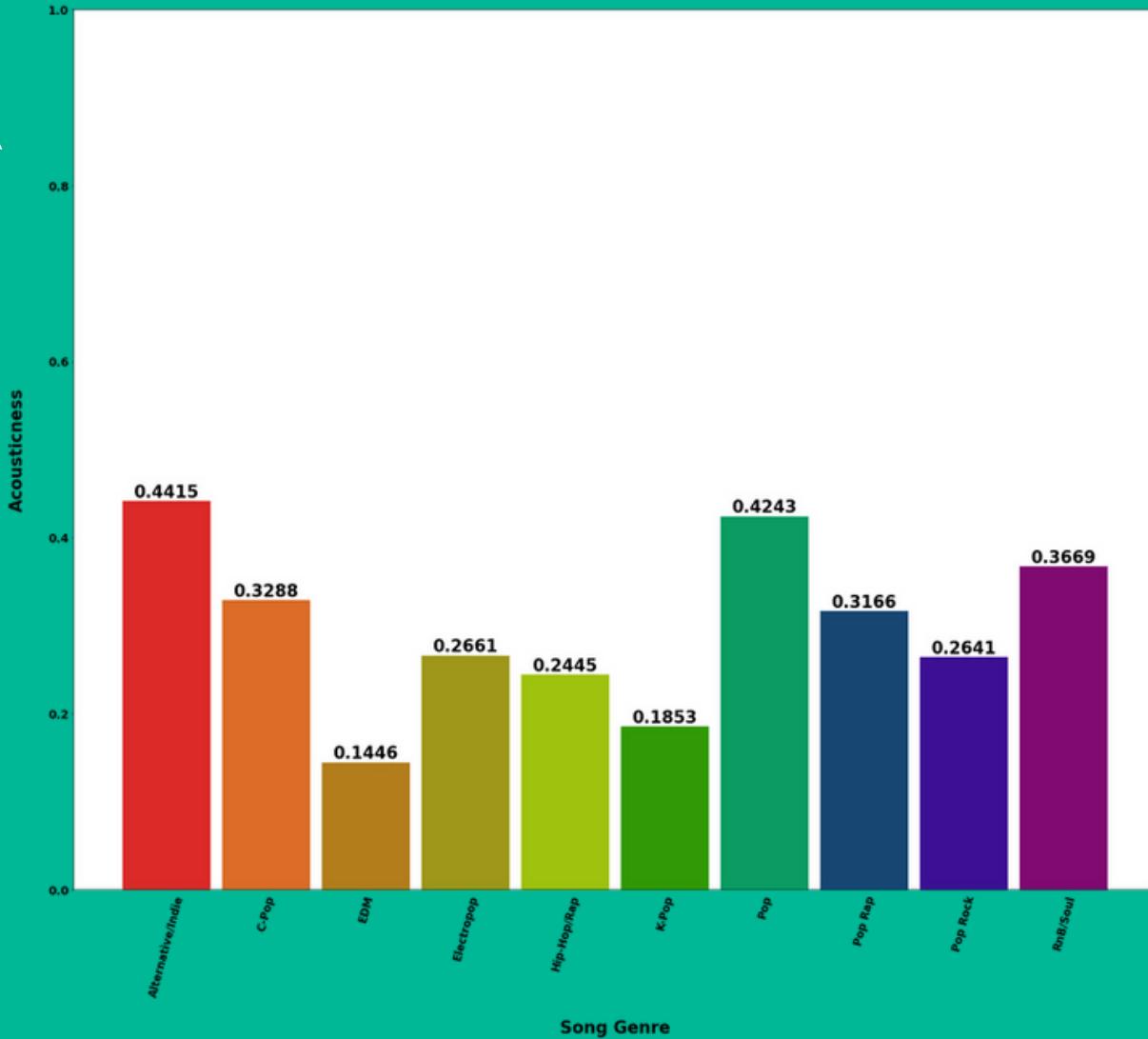
#Show the Query Data Chart
plt.show()
```

Acti

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

g. Track Genre with Average Acousticness Ratio

Comparison Graph of Song Genres on ASEAN Charts in August 2022
Based on Song Acousticness



Analysis:

Alternative/Indie tracks have the most significant average acousticness ratio among other genres of 0.4415, followed by Pop tracks (0.4243). Whereas EDM tracks that have the smallest average ratio of 0.1446.

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

h. Track Genre with Average Worldwide Popularity Score

```
In [40]: #Query Data
data_genre_popularity = data.groupby(['genre'], as_index = False)[['popularity']].mean()
data_genre_popularity = data_genre_popularity.drop([1,3,7,9,11,14,15,16,18])

#size figure chart
plt.figure(figsize=(30,25))

#Input axis variable
x = data_genre_popularity['genre']
y = data_genre_popularity["popularity"]

#visualize the data with bar chart
data_genre_popularity_plot = plt.bar(x,y, width = 0.8, color = ['#db2a27','#dc6c27','#b37d1b','#9e9719','#ec18e','#029986',
                           '#69c62','#174673','#3d9f94','#99aa70'])
plt.bar_label(data_genre_popularity_plot, label_type = 'edge', fontsize = 20, fmt='%.4f', fontweight = 'bold')

#Add Title, X-axis and Y-axis Label
plt.title('\ncomparison graph of song genres on ADAM charts in August 2022\nbased on Song Popularity\n', fontsize = 30,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel('\nSong genre', fontsize = 20, fontweight = 'bold')
plt.ylabel('Popularity\n', fontsize = 20, fontweight = 'bold')

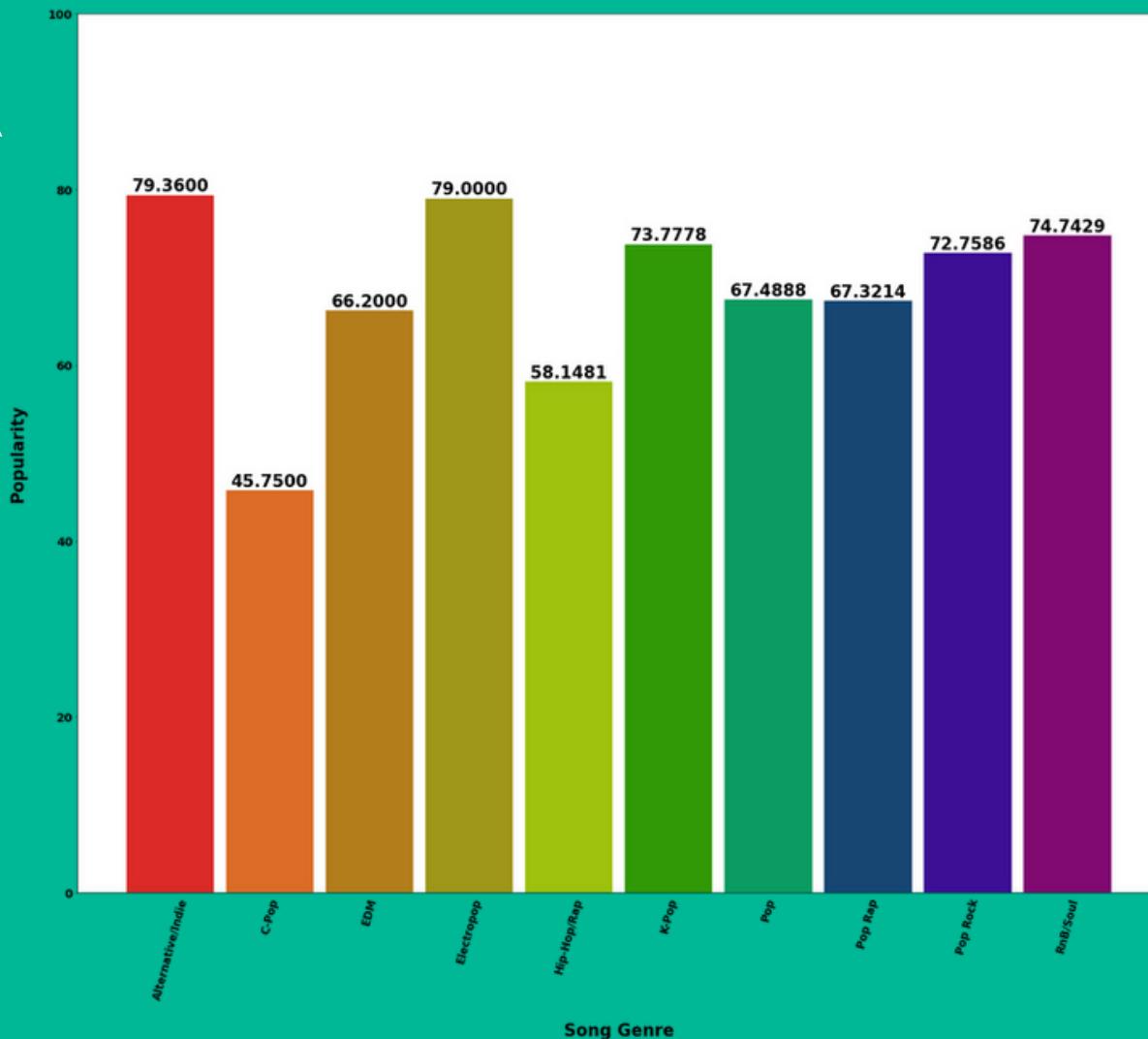
#Axis chart Settings
plt.ylim(ymax = 100)
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

#show the Query Data Chart
plt.show()
```

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

h. Track Genre with Average Worldwide Popularity Score

Comparison Graph of Song Genres on ASEAN Charts in August 2022
Based on Song Popularity



Analysis:

Alternative/Indie songs have the highest average popularity in the world, with an average rating of 79.36, followed by Electropop songs (79). In contrast, C-Pop songs have the lowest worldwide popularity, averaging at 45.75.

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

i. Track Genre with Average Track Valence Ratio

```
In [28]: #Data Query
data_genre_valence = data.groupby(['genre'], as_index = False)[["valence"]].mean()
data_genre_valence = data_genre_valence.drop([2,3,7,9,11,14,15,16,18])

#Resize figure chart
plt.figure(figsize=(30,25))

#Input axis variable
x = data_genre_valence['genre']
y = data_genre_valence['valence']

#visualize the data with bar chart
data_genre_valence_plot = plt.bar(x,y, width = 0.9, color = ['#db3a37','#dc6c37','#b37d1b','#e9719','#9ec20e','#129995',
                '#bbdc62','#174673','#3d0f94','#800a70'])
plt.bar_label(data_genre_valence_plot, label_type = 'edge', fontsize = 24, font='X.4f', fontweight = 'bold')

#Add Title, X-axis and Y-axis Label
plt.title('\nComparison Graph of song Genres on ASIAN Charts in August 2022\nbased on song valence\n', fontsize = 36,
          fontname = 'Times New Roman', fontweight = 'bold')
plt.xlabel("\nSong genre", fontsize = 26, fontweight = 'bold')
plt.ylabel("Valence Ratio\n", fontsize = 26, fontweight = 'bold')

#Axis Chart Settings
plt.ylim(ymax = 1)
plt.yticks(fontsize=18, fontweight="bold")
plt.xticks(fontsize=17, fontweight="bold", rotation = 75)
plt.tick_params(bottom = False)

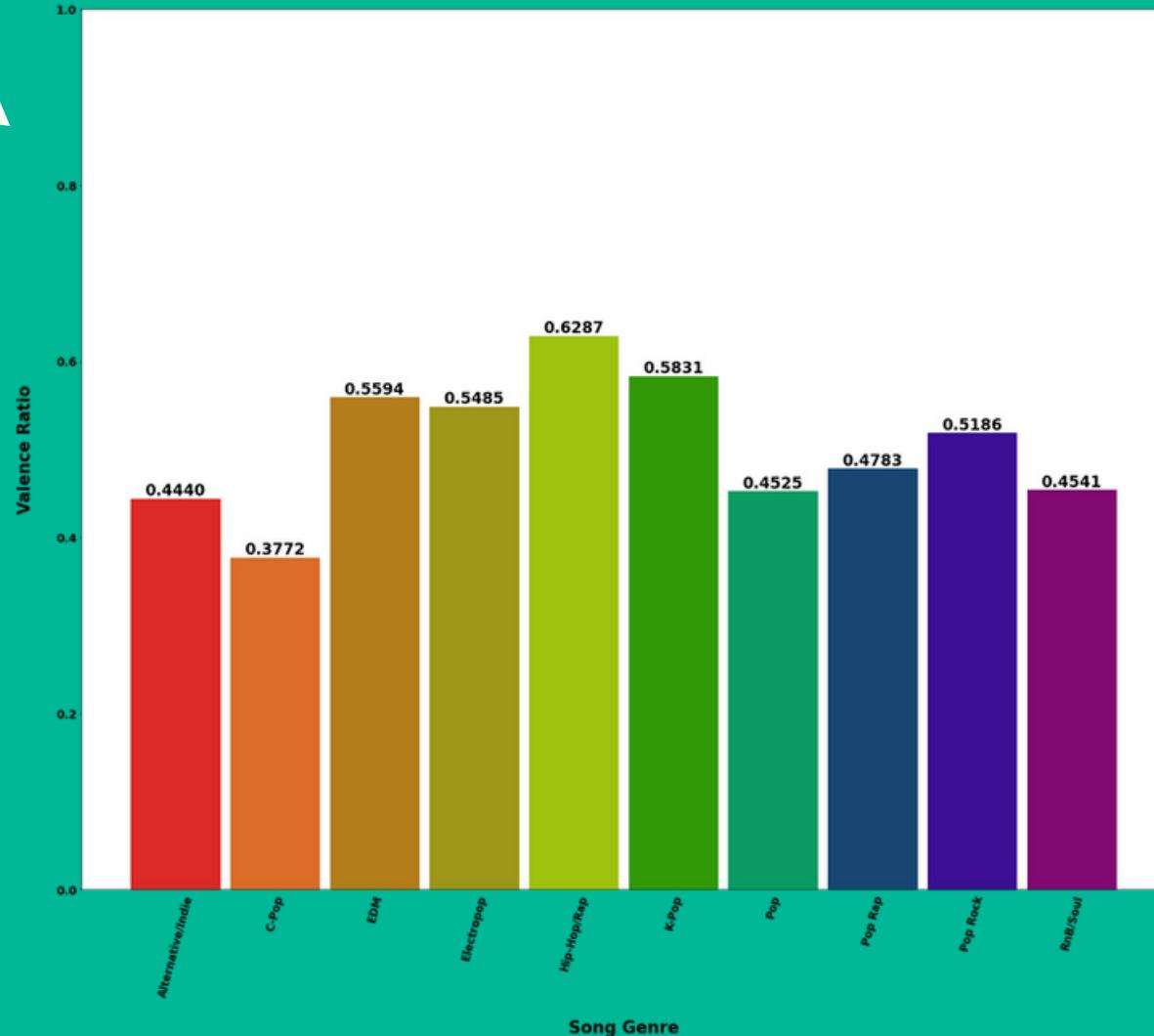
#Show the Chart
plt.show()
```

Activ

ANALYSIS OF AVERAGE ATTRIBUTES IN TRACK GENRES

i. Track Genre with Average Track Valence Ratio

Comparison Graph of Song Genres on ASEAN Charts in August 2022
Based on Song Valence



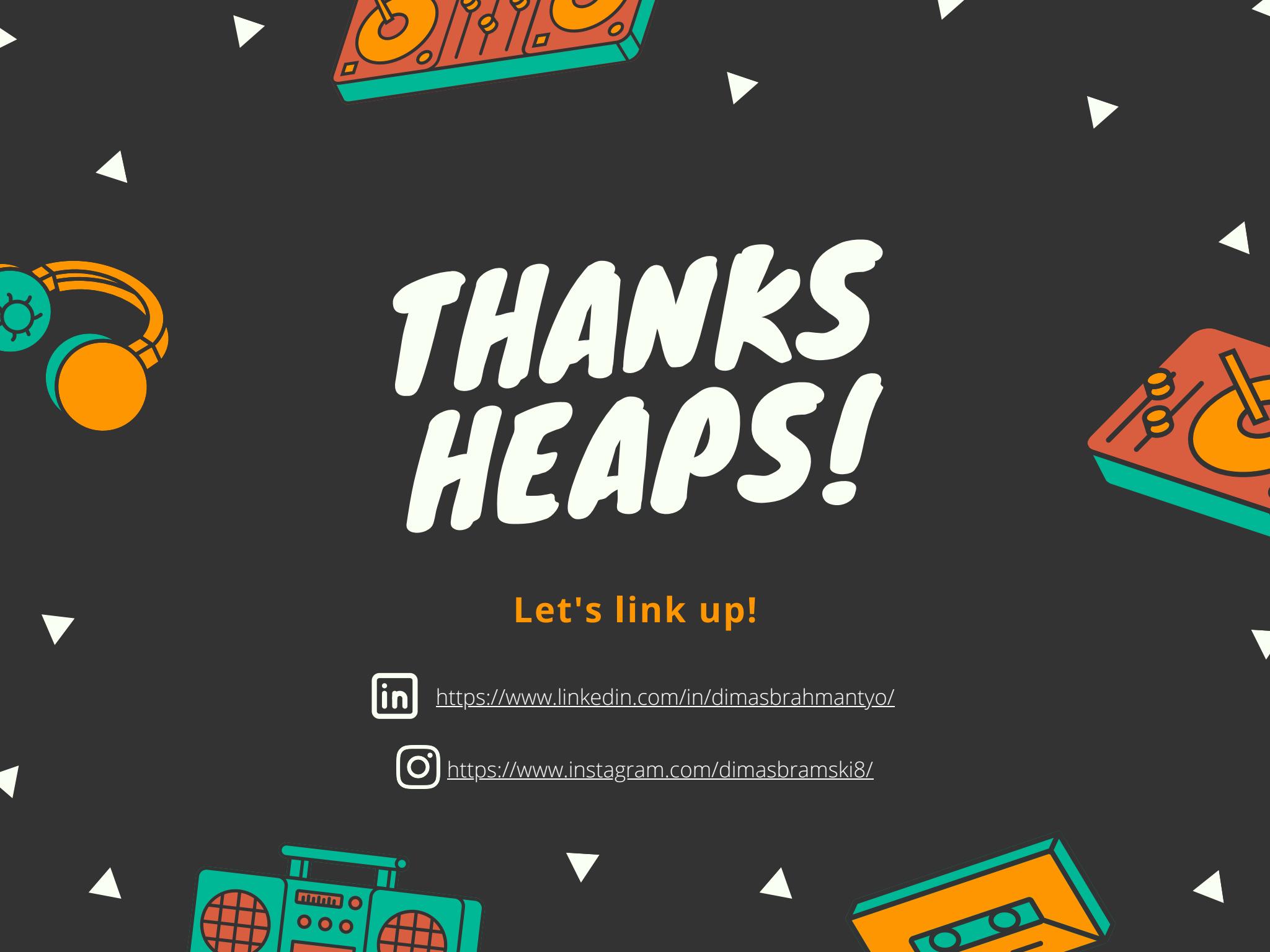
Analysis:

Hip-hop/Rap became the genre with the highest average valence ratio (0.6287), which shows the greatest positivity on the mood of the genre track. C-Pop has the lowest valence ratio among other genres, with a valence ratio of 0.3772.

CONCLUSIONS

Based on the data of the Top 50 ASEAN Song Charts that we have analyzed previously, the following information is obtained:

- There are **349** songs from **245** artists on the music charts of 10 ASEAN countries.
- **Mark Tuan** became the singer with the most tracks, as many as **24 tracks**. "**Pink Venom**" from **Black Pink** became the song with the widest chart spread in **9 out of 10 ASEAN countries**.
- The tracks listed on the charts are from **1969 to 2022**, with the music tracks published in 2022 having the highest number of tracks at 219.
- The ASEAN song-charts tracks are divided into **19 genres**, with **Pop** followed by **K-pop** having the highest number of genres.
- Based on correlation map analysis, **the valence ratio** of music tracks has the largest **positive correlation to the ratio of danceability and energy**. The most significant positive correlation is found in the relationship between the energy ratio and the track's loudness. In contrast, the most significant negative correlation occurs in the relationship between the energy ratio and acousticness ratio of the music track.
- The genre of a music track has a different effect on each attribute that affects a track.



THANKS HEAPS!

Let's link up!



<https://www.linkedin.com/in/dimasbrahmantyo/>



<https://www.instagram.com/dimasbramski8/>