SPOTIFY

DATA ANALYSIS PROJECT

BY - SUBROTO DUTTA

KEY OBJECTIVES

01 Conduct necessary data preparation steps and optimize the dataset for further analysis.

02 Analyze top tracks and popular artists within the dataset.

03 Conduct regression analysis on different audio features.

04 Investigate trends in music production and track length over the years.

05 Perform genre analysis to determine influence on song characteristics and popularity trends.

06 Summarize key findings highlighting the most significant patterns and conclusions from the data.

WORKFLOW MAP

1

DATA PREPARATION

Cleaning, formatting, and getting data ready for analysis

2

DATA EXPLORATION

Initiating an overview and gaining insights from data

3

DATA VISUALIZATION

Visualizing insights through visuals and graphs

4

ANALYTICAL SUMMARY

Summarizing key findings and insights from the analysis

01

DATA PREPARATION

REMOVING NULL VALUES

In [4]:
```
# Checking null values in each column.
tracks.isnull().sum()
```

Out[4]:
```
id                 0
name              71
popularity         0
duration_ms        0
explicit           0
artists            0
id_artists         0
year               0
danceability       0
energy             0
key                0
loudness           0
mode               0
speechiness        0
acousticness       0
instrumentalness   0
liveness           0
valence            0
tempo              0
time_signature     0
dtype: int64
```

In [5]:
```
#Droping null values
tracks.dropna(inplace = True)

# Checking null values after droping all null rows
tracks.isnull().sum()
```

Out[5]:
```
id                 0
name               0
popularity         0
duration_ms        0
explicit           0
artists            0
id_artists         0
year               0
danceability       0
energy             0
key                0
loudness           0
mode               0
speechiness        0
acousticness       0
instrumentalness   0
liveness           0
valence            0
tempo              0
time_signature     0
dtype: int64
```

DATASET OPTIMIZATION

```python
# Creating a duration column to show duration in seconds using apply method and lambda function
tracks['duration'] = tracks['duration_ms'].apply(lambda x : round(x/1000))

# Pulling the new duration column
tracks['duration'].head(10)
```

```
0    127
1     98
2    182
3    177
4    163
5    179
6    134
7    161
8    310
9    181
Name: duration, dtype: int64
```

Removed unnecessary columns from the dataset to streamline the dataset and shift focus to relevant features for analysis.

[180]:

```python
# Droping unnecessary columns
tracks.drop(['id','id_artists','key','mode', 'time_signature','duration_ms'], axis = 1, inplace =True)

#Checking the final dataset
tracks.head()
```

[180]:

	name	popularity	explicit	artists	year	danceability	energy	loudness	speechiness	acousticness	instrumentalness	liveness	valence	te
0	Carve	6	0	['Uli']	1922	0.645	0.4450	-13.338	0.4510	0.674	0.7440	0.151	0.127	104
1	Capítulo 2.16 - Banquero Anarquista	0	0	['Fernando Pessoa']	1922	0.695	0.2630	-22.136	0.9570	0.797	0.0000	0.148	0.655	102
2	Vivo para Quererte - Remasterizado	0	0	['Ignacio Corsini']	1922	0.434	0.1770	-21.180	0.0512	0.994	0.0218	0.212	0.457	130
3	El Prisionero - Remasterizado	0	0	['Ignacio Corsini']	1922	0.321	0.0946	-27.961	0.0504	0.995	0.9180	0.104	0.397	166
4	Lady of the Evening	0	0	['Dick Haymes']	1922	0.402	0.1580	-16.900	0.0390	0.989	0.1300	0.311	0.196	109

Created new **duration** column by converting the values from milliseconds to seconds using the lambda function, allowing the data for more straightforward comparisons and visualizations of track lengths in subsequent analytical tasks.

02

DATA EXPLORATION

POPULARITY ANALYSIS

TOP POPULAR TRACKS

```python
# Using sort values and head method using populrity as the base to extract top 10 popular songs
tracks[['name','artists', 'popularity']].sort_values('popularity', ascending = False).head(10).reset_index(drop = True)
```

	name	artists	popularity
0	Peaches (feat. Daniel Caesar & Giveon)	['Justin Bieber', 'Daniel Caesar', 'Giveon']	100
1	drivers license	['Olivia Rodrigo']	99
2	Astronaut In The Ocean	['Masked Wolf']	98
3	telepatía	['Kali Uchis']	97
4	Save Your Tears	['The Weeknd']	97
5	Leave The Door Open	['Bruno Mars', 'Anderson .Paak', 'Silk Sonic']	96
6	Blinding Lights	['The Weeknd']	96
7	The Business	['Tiésto']	95
8	Fiel	['Los Legendarios', 'Wisin', 'Jhay Cortez']	94
9	Bandido	['Myke Towers', 'Juhn']	94

TOP POPULAR ARTISTS

```python
# Using pivot table method to calculate mean popularity of each artist
artists_plty = tracks[['artists','popularity']].pivot_table(index = 'artists',values = 'popularity', aggfunc= 'mean')


# Sorting values and indexing to find out top 10 popular artists
artists_plty.sort_values('popularity', ascending = False).reset_index()[:10]
```

	artists	popularity
0	['Riton', 'Nightcrawlers', 'Mufasa & Hypeman',...	94.0
1	['Los Legendarios', 'Wisin', 'Jhay Cortez']	94.0
2	['Bad Bunny', 'ROSALÍA']	93.0
3	['Travis Scott', 'HVME']	92.0
4	['Rochy RD', 'Myke Towers', 'Nicki Nicole']	92.0
5	['MEDUZA', 'Dermot Kennedy']	92.0
6	['Nathan Evans', '220 KID', 'Billen Ted']	92.0
7	['Bad Bunny', 'Jhay Cortez']	91.0
8	['Saweetie', 'Doja Cat']	90.0
9	['Maroon 5', 'Megan Thee Stallion']	90.0

STATISTICAL INSIGHTS

DESCRIPTIVE STATISTICS

```
# Using Describe and Transpose function to display descriptive statistics for the dataset
tracks.describe().transpose()
```

	count	mean	std	min	25%	50%	75%	max
popularity	586601.0	27.573212	18.369417	0.0	13.0000	27.000000	41.00000	100.000
explicit	586601.0	0.044091	0.205298	0.0	0.0000	0.000000	0.00000	1.000
year	586601.0	1988.571729	22.826243	1900.0	1974.0000	1992.000000	2007.00000	2021.000
danceability	586601.0	0.563612	0.166101	0.0	0.4530	0.577000	0.68600	0.991
energy	586601.0	0.542071	0.251910	0.0	0.3430	0.549000	0.74800	1.000
loudness	586601.0	-10.205789	5.089422	-60.0	-12.8910	-9.242000	-6.48100	5.376
speechiness	586601.0	0.104870	0.179902	0.0	0.0340	0.044300	0.07630	0.971
acousticness	586601.0	0.449803	0.348812	0.0	0.0969	0.422000	0.78400	0.996
instrumentalness	586601.0	0.113425	0.266843	0.0	0.0000	0.000024	0.00955	1.000
liveness	586601.0	0.213933	0.184328	0.0	0.0983	0.139000	0.27800	1.000
valence	586601.0	0.552306	0.257673	0.0	0.3460	0.564000	0.76900	1.000
tempo	586601.0	118.467930	29.762942	0.0	95.6060	117.387000	136.32400	246.381
duration	586601.0	230.054333	126.532822	3.0	175.0000	215.000000	264.00000	5621.000

CORRELATIONAL DATASET

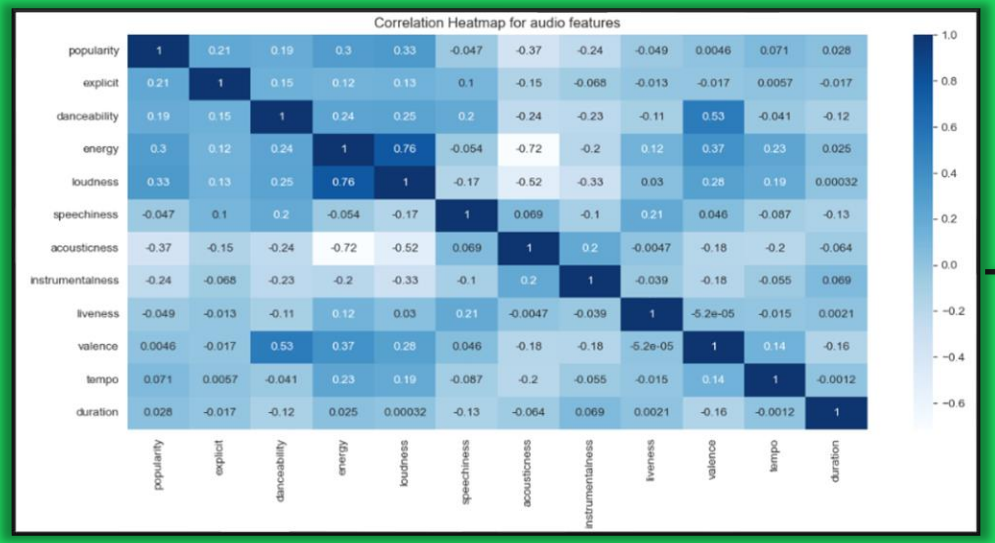
```
# Creating a correlational dataset using corr function through pearson correlation method
corr_tracks = tracks.drop(['year','name','artists'], axis = 1).corr('pearson')
corr_tracks
```

	popularity	explicit	danceability	energy	loudness	speechiness	acousticness	instrumentalness	liveness	valence	tempo	durati
popularity	1.000000	0.211749	0.186878	0.302178	0.327001	-0.047415	-0.370723	-0.236403	-0.048735	0.004558	0.071223	0.0276
explicit	0.211749	1.000000	0.150216	0.123060	0.134598	0.102251	-0.149001	-0.067510	-0.013113	-0.016551	0.005723	-0.0167
danceability	0.186878	0.150216	1.000000	0.241464	0.251394	0.199291	-0.242838	-0.225831	-0.106175	0.528136	-0.040896	-0.1204
energy	0.302178	0.123060	0.241464	1.000000	0.764744	-0.053560	-0.715366	-0.195727	0.124636	0.372224	0.230006	0.0247
loudness	0.327001	0.134598	0.251394	0.764744	1.000000	-0.167140	-0.519423	-0.329255	0.029509	0.275416	0.189252	0.0003
speechiness	-0.047415	0.102251	0.199291	-0.053560	-0.167140	1.000000	0.069121	-0.102425	0.207062	0.046481	-0.086950	-0.1257
acousticness	-0.370723	-0.149001	-0.242838	-0.715366	-0.519423	0.069121	1.000000	0.204312	-0.004742	-0.180878	-0.195117	-0.0643
instrumentalness	-0.236403	-0.067510	-0.225831	-0.195727	-0.329255	-0.102425	0.204312	1.000000	-0.038836	-0.175195	-0.055380	0.0693
liveness	-0.048735	-0.013113	-0.106175	0.124636	0.029509	0.207062	-0.004742	-0.038836	1.000000	-0.000052	-0.014923	0.0021
valence	0.004558	-0.016551	0.528136	0.372224	0.275416	0.046481	-0.180878	-0.175195	-0.000052	1.000000	0.135198	-0.1632
tempo	0.071223	0.005723	-0.040896	0.230006	0.189252	-0.086950	-0.195117	-0.055380	-0.014923	0.135198	1.000000	-0.0012
duration	0.027640	-0.016748	-0.120408	0.024785	0.000322	-0.125782	-0.064395	0.069321	0.002140	-0.163231	-0.001248	1.0000

03

DATA VISUALIZATION

CORRELATION HEATMAP

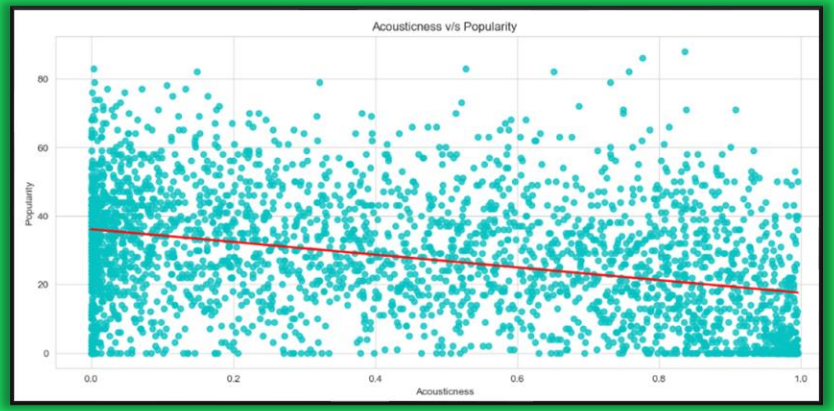


The heatmap reveals several key insights -

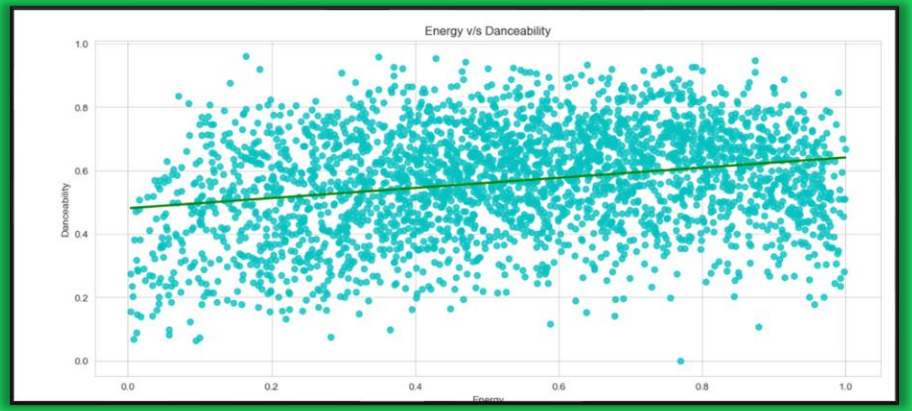
- Energy and loudness have a strong positive correlation, indicating louder songs tend to be more energetic.
- Acousticness has a strong negative correlation with energy and loudness, suggesting that more acoustic songs are generally less energetic and quieter.
- Popularity shows moderate positive correlations with energy and loudness.

REGGRESSION ANALYSIS

ACOUSTICNESS V/S POPULARITY

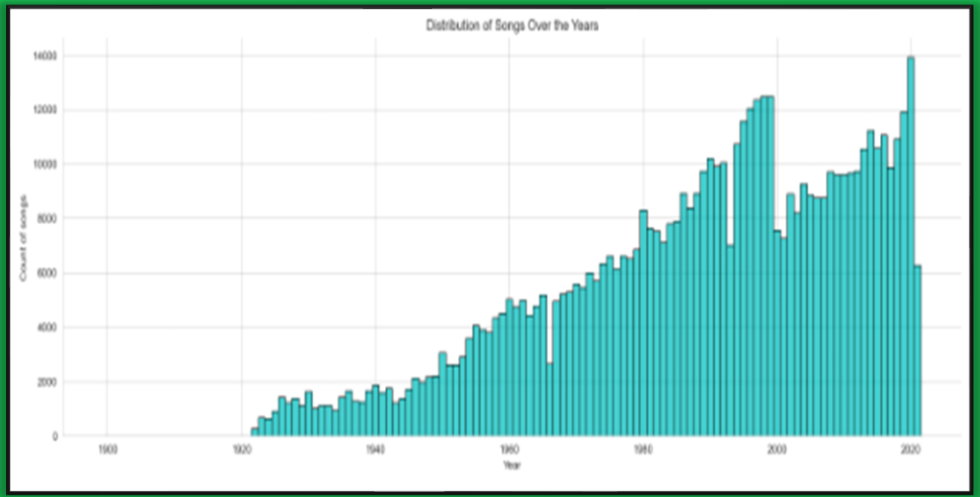


ENERGY V/S DANCEABILITY

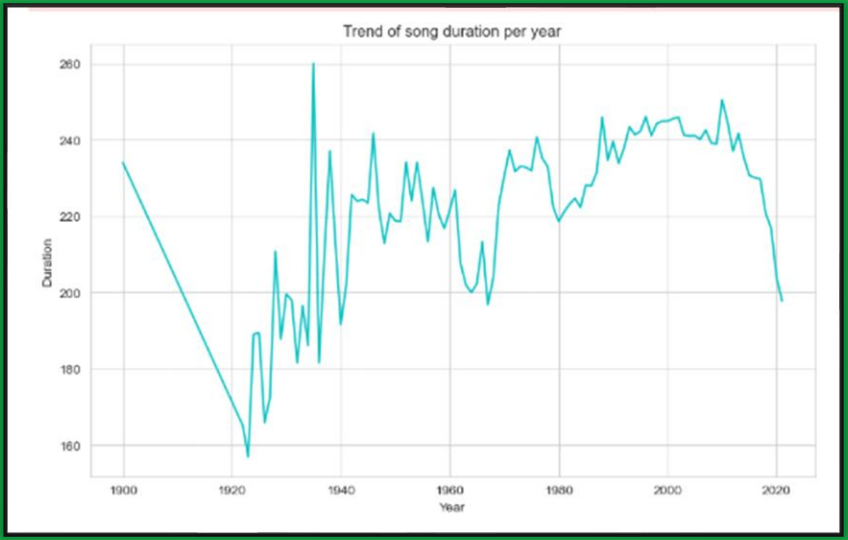


TREND ANALYSIS

TRACK RELEASE BY **YEAR**

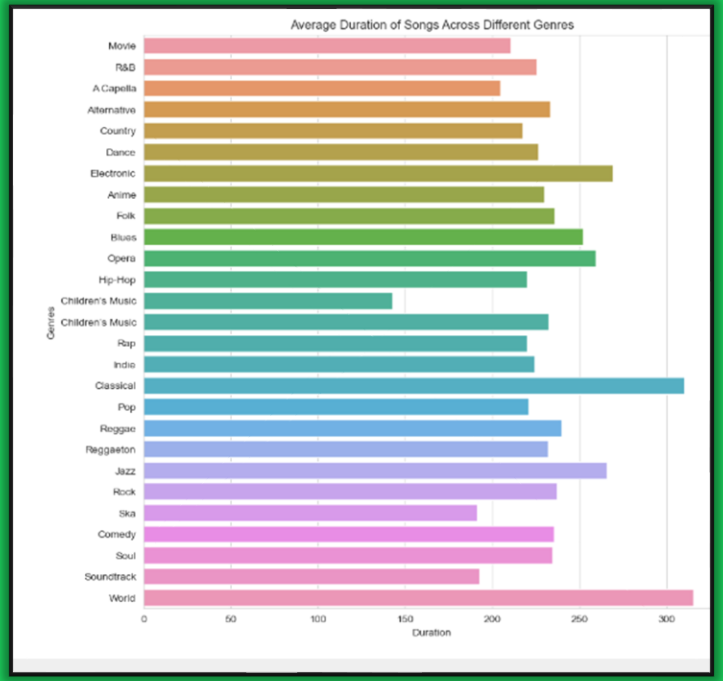


AVERAGE TRACK LENGTH OVER **YEARS**

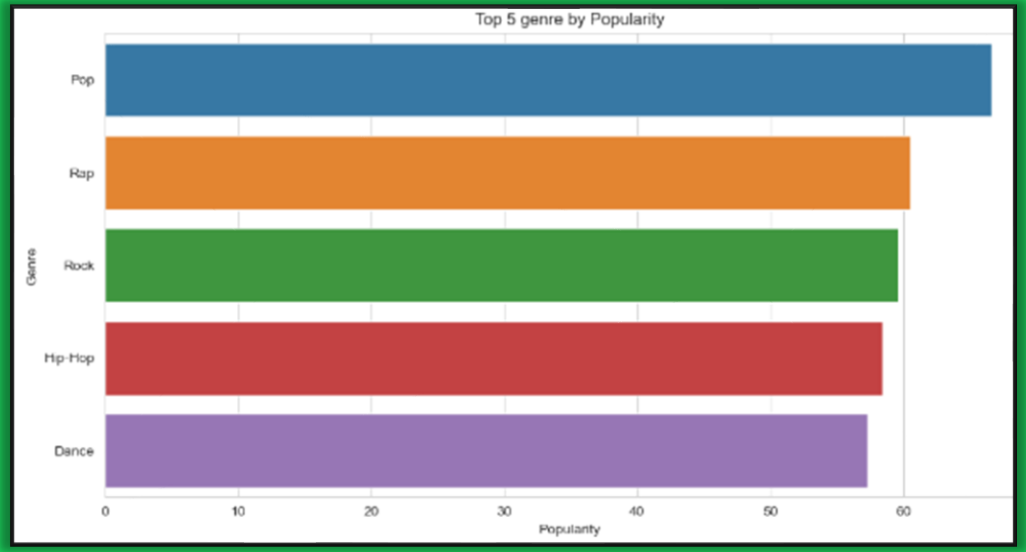


TREND ANALYSIS

AVERAGE SONG DURATION BY GENRE



TOP GENRE BY POPULARITY



04

ANALYTICAL SUMMARY

KEY INSIGHTS

01

TREND ANALYSIS

- The **trend of music production** over the years reveals consistent growth, with a notable surge around 1960 and a recent peak, reflecting the music industry's response to technological advancements and cultural shifts impacting popularity of music across different genres.
- The shift towards **shorter song durations** reflects changes in consumption patterns, driven by shorter attention spans, which emphasize concise and engaging content to capture attention

02

CORRELATIONAL INSIGHTS

- A strong positive correlation exists between **energy and loudness**, suggesting that energetic songs are also louder. Additionally, a strong positive regression is observed between **energy and danceability**, highlighting their combined impact on song popularity.
- However, a negative correlation exists between **energy and acousticness**, therefore it's obvious that negative regression is derived between **popularity and acousticness**, thereby negatively affecting the popularity of the song.

03

GENRE ANALYSIS

- Analyzing different genres, **Pop** emerges as the most popular genre among the top five, followed closely by **Rap, Rock, Hip-Hop, and Dance** based on their respective average popularity ratings.
- **Classical genre** typically has the longest average durations due to its complex compositions. In contrast, **Children's music** is designed to be short and catchy for young audiences, resulting in significantly shorter durations.

THANK YOU