ETL and EDA Implementation

Import Libraries

```
# Import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline
```

Loading the dataset

```
file_path = '/content/drive/MyDrive/Copy_of_spotify.csv'
df1 = pd.read_csv(file_path)
df1.head(10)
```

	Artist	Track	Album_type	Danceability	Energy	Loudness	Speechiness	Acousticness	Instrumentalness	...	Title	Channel	Views	Likes	Comments	Licensed	official_video
0	Gorillaz	Feel Good Inc(feat. Daryl Hall)	album	0.818	0.705	-6.679	0.1770	0.0293e-02	0.0015e-02	...	Gorillaz - Feel Good Inc(feat. Daryl Hall)(Official Video)	Gorillaz	693550213.0	5110568.0	168072.0	True	True
1	Gorillaz	Rhinestone Eyes	single	0.676	0.793	-5.815	0.0332	0.0042e-03	0.0000e-07	...	Gorillaz - Rhinestone Eyes(Official Video)	Gorillaz	72074642.0	1076318.0	31053.0	True	True
2	Gorillaz	New Gold(feat. Andrew Wyatt, Pusha T & Stevie Nicks)	single	0.695	0.923	-3.930	0.0522	0.0432e02	0.0000e02	...	Gorillaz - New Gold(feat. Andrew Wyatt, Pusha T & Stevie Nicks)(Official Video)	Gorillaz	94355053.0	2621413.0	73953.0	True	True
3	Gorillaz	El Mañana	Plastic Beach	album	0.689	0.739	-5.643	0.0280	0.0005e-05	0.0360e02	Gorillaz - El Mañana(Official Video)	Gorillaz	157594922.0	1796773.0	50229.0	True	True
4	Gorillaz	Clint Eastwood	album	0.663	0.694	-8.627	0.1710	0.0153e03	0.0293e02	...	Gorillaz - Clint Eastwood(Official Video)	Gorillaz	874809904.0	8787816.0	550050.0	True	True

5 rows × 20 columns

Initial Data Exploration

```
print("Dataset shape: {df.shape}")
print("\nColumns: {df.columns.tolist()}")
print("\nData Types:\n{df.dtypes}")
print("\nMissing Values:\n{df.isnull().sum()}")
df.head(5)
df.describe(include=[np.number])
```

```
Dataset shape: (20718, 27)

Columns: ['Artist', 'Track', 'Album_type', 'Danceability', 'Energy', 'Loudness', 'Speechiness', 'Acousticness', 'Instrumentalness', 'Liveness', 'Valence', 'Tempo', 'Duration_min', 'Views', 'Likes', 'Comments', 'Licensed', 'official_video', 'Stream', 'Energy+Loudness', 'mood_score', 'popularity_views']

Data Types:
Artist                 object
Track                  object
Album_type             object
Danceability          float64
Energy                float64
Loudness              float64
Speechiness           float64
Acousticness          float64
Instrumentalness      float64
Liveness              float64
Valence               float64
Tempo                 float64
Duration_min          float64
Views                 float64
Likes                 float64
Comments              float64
Licensed               object
official_video         object
Stream                 object
Energy+Loudness       float64
mood_score            float64
popularity_views      float64
dtype: object

Missing Values:
Artist                 2
Track                  2
Album_type             2
Danceability           2
Energy                 2
Loudness               2
Speechiness            2
Acousticness           2
Instrumentalness       2
Liveness               2
Valence                2
Tempo                  2
Duration_min           2
Views                 95
Likes                 96
Comments              97
Licensed               2
official_video         2
Stream                95
Energy+Loudness        2
mood_score             2
popularity_views      95
dtype: object
```

Data Cleaning

```
# Dropping the title column as it is a duplicate of track. It has the same information.
df.drop(['Title'], inplace=True)
df.shape, df.columns.tolist()

# Dropping rows that has missing values of Energy+Loudness
df[df.duplicated(subset=['Energy+Loudness'])]

# Standardizing the text
text_columns= ['Artist', 'Track', 'Album_type', 'Channel', 'Licensed', 'official_video', 'mood_playedon']
for col in text_columns:
    df[col] = df[col].str.lower().str.strip()
```

Feature Engineering

```
# Creating 2 new features
# Measures user interaction relative to views
df['engagement_rate'] = df['Likes'].fillna(0)/df['Comments'].fillna(0)*100

# Combines tempo and energy to represent mood
df['mood_score'] = df['Tempo']*df['Energy']

# Integrates danceability and energy to assess dance potential
df['danceability_score'] = df['Danceability']*df['Tempo']
```

Top 10 Most Viewed tracks

```
top_viewed = df.groupby('Track')['Views'].agg(['count','mean','sum','median','max','min'])
print(top_viewed)
```

```
                                                  count          mean  \
Track                                                                   
!!!                                                   1  1.000000e+00
$ ( dollars )                                         1  4.270000e+03
$ ( dollars ) - remix                                 1  1.000000e+00
$ ( dollars ) (feat. charlie puth)                    1  5.775700e+05
$ ( dollars ) (feat. lil uzi vert)                    1  1.000000e+00
$ ( dollars ) - spider-man (feat. tekno miles ...     1  1.000000e+00
$ ( dollars ) (feat. lil uzi vert) - remix            1  1.000000e+00
$ ( dollars ) - remix (feat. ty dolla $ign & ...      1  1.000000e+00
$ ( dollars ) - remix (feat. ty dolla $ign & ...      1  1.000000e+00
$ ( dollars ) (feat. tekno miles & lil uzi ver...     
```