```
/kaggle/input/spotify-features/SpotifyFeatures.csv
/kaggle/input/spotify-tracks-data/tracks.csv
/kaggle/input/spotify-tracks-data/Spotify.png
```

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

add Codeadd Markdown

Reading the Datasets

```
df_tracks = pd.read_csv('/kaggle/input/spotify-tracks-data/tracks.csv')
```

df_tracks.head()

df_tracks.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 586672 entries, 0 to 586671
Data columns (total 20 columns):
 #   Column           Non-Null Count   Dtype  
---  ------           --------------   -----  
 0   id               586672 non-null  object 
 1   name             586601 non-null  object 
 2   popularity       586672 non-null  int64  
 3   duration_ms      586672 non-null  int64  
 4   explicit         586672 non-null  int64  
 5   artists          586672 non-null  object 
 6   id_artists       586672 non-null  object 
 7   release_date     586672 non-null  object 
 8   danceability     586672 non-null  float64
 9   energy           586672 non-null  float64
 10  key              586672 non-null  int64  
 11  loudness         586672 non-null  float64
 12  mode             586672 non-null  int64  
 13  speechiness      586672 non-null  float64
 14  acousticness     586672 non-null  float64
 15  instrumentalness 586672 non-null  float64
 16  liveness         586672 non-null  float64
 17  valence          586672 non-null  float64
 18  tempo            586672 non-null  float64
 19  time_signature   586672 non-null  int64  
dtypes: float64(9), int64(6), object(5)
memory usage: 89.5+ MB
```

add Codeadd Markdown

```
pd.isnull(df_tracks).sum()
```

[6]:

```
id                0
name             71
popularity        0
duration_ms       0
explicit          0
artists           0
id_artists        0
release_date      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
```

add Codeadd Markdown

```
sorted_df = df_tracks.sort_values('popularity', ascending = True).head()
sorted_df
```

df_tracks.describe().transpose()

	count	mean	std	min	25%	50%	75%	max
popularity	586672.0	27.570053	18.370642	0.0	13.0000	27.000000	41.00000	100.000
duration_ms	586672.0	230051.167286	126526.087418	3344.0	175093.0000	214893.000000	263867.00000	5621218.000
explicit	586672.0	0.044086	0.205286	0.0	0.0000	0.000000	0.00000	1.000
danceability	586672.0	0.563594	0.166103	0.0	0.4530	0.577000	0.68600	0.991
energy	586672.0	0.542036	0.251923	0.0	0.3430	0.549000	0.74800	1.000
key	586672.0	5.221603	3.519423	0.0	2.0000	5.000000	8.00000	11.000
loudness	586672.0	-10.206067	5.089328	-60.0	-12.8910	-9.243000	-6.48200	5.376
mode	586672.0	0.658797	0.474114	0.0	0.0000	1.000000	1.00000	1.000

	count	mean	std	min	25%	50%	75%	max
speechiness	586672.0	0.104864	0.179893	0.0	0.0340	0.044300	0.07630	0.971
acousticness	586672.0	0.449863	0.348837	0.0	0.0969	0.422000	0.78500	0.996
instrumentalness	586672.0	0.113451	0.266868	0.0	0.0000	0.000024	0.00955	1.000
liveness	586672.0	0.213935	0.184326	0.0	0.0983	0.139000	0.27800	1.000
valence	586672.0	0.552292	0.257671	0.0	0.3460	0.564000	0.76900	1.000
tempo	586672.0	118.464857	29.764108	0.0	95.6000	117.384000	136.32100	246.381
time_signature	586672.0	3.873382	0.473162	0.0	4.0000	4.000000	4.00000	5.000

```
most_popular = df_tracks.query('popularity>90', inplace = False).sort_values('popularity',ascending=False)
most_popular
```

```
df_tracks.set_index("release_date",inplace=True)
df_tracks.index = pd.to_datetime(df_tracks.index)
df_tracks.head()
```

[10]:

```
df_tracks[['artists']].iloc[18]
```

[11]:

```
artists    ['Victor Boucher']
Name: 1922-01-01 00:00:00, dtype: object
```

add Codeadd Markdown

```
df_tracks["duration"] = df_tracks["duration_ms"].apply(lambda x: round(x/1000))
df_tracks.drop("duration_ms", inplace = True, axis = 1)
```

```
df_tracks.duration.head()
```

[13]:

```
release_date
```

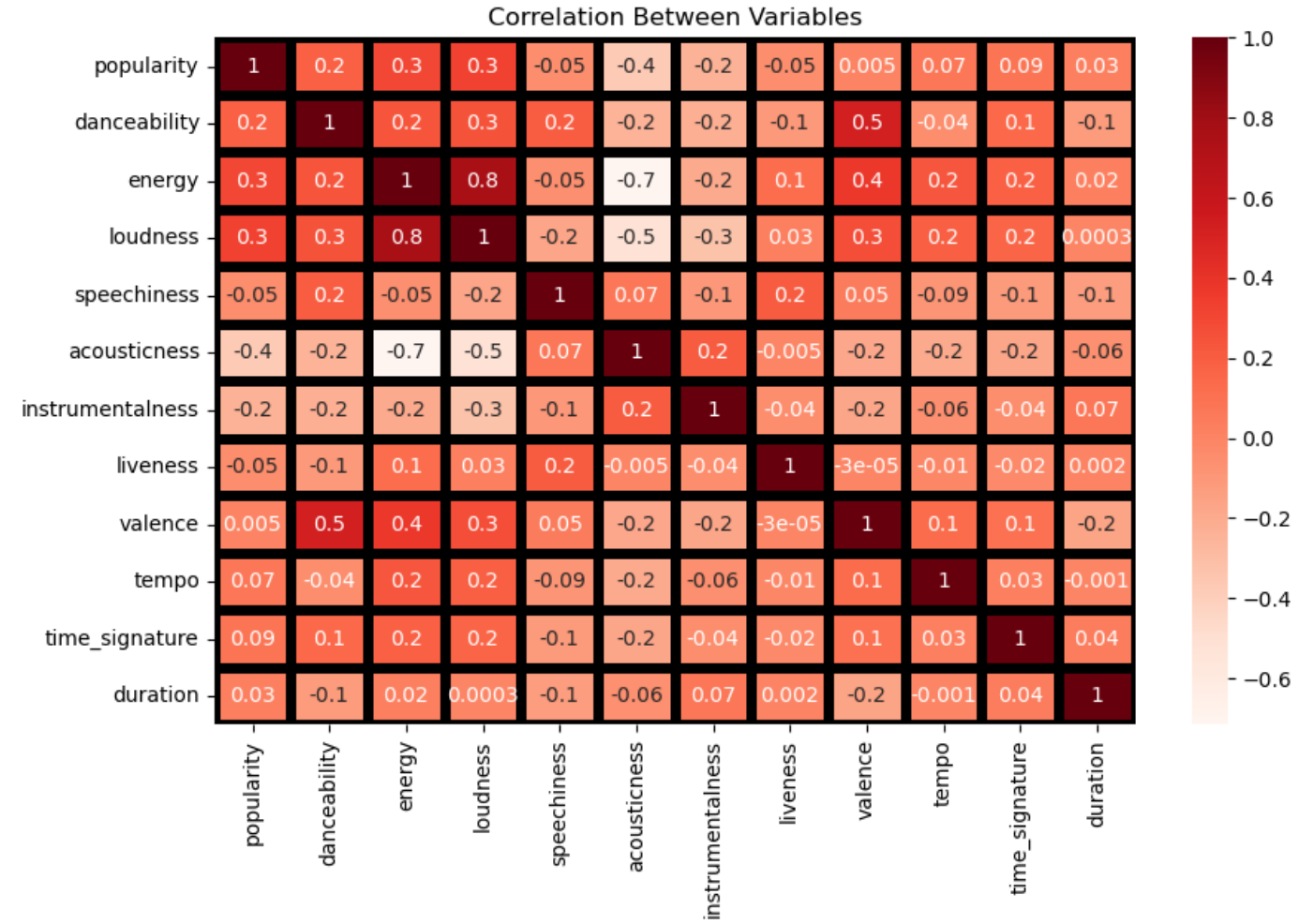
```
1922-02-22    127
1922-06-01     98
1922-03-21    182
1922-03-21    177
1922-01-01    163
Name: duration, dtype: int64
```

```
corr_df = df_tracks.drop(["key", "mode", "explicit"], axis = 1).corr(method = "pearson", numeric_only = True)
```

add Codeadd Markdown

```
plt.figure(figsize = (10,6))
ax = sns.heatmap(corr_df, annot = True, fmt = ".1g", linecolor = 'k', linewidths = '5', cmap = 'Reds')
plt.title("Correlation Between Variables")
```

Text(0.5,1.0, 'Correlation Between Variables')



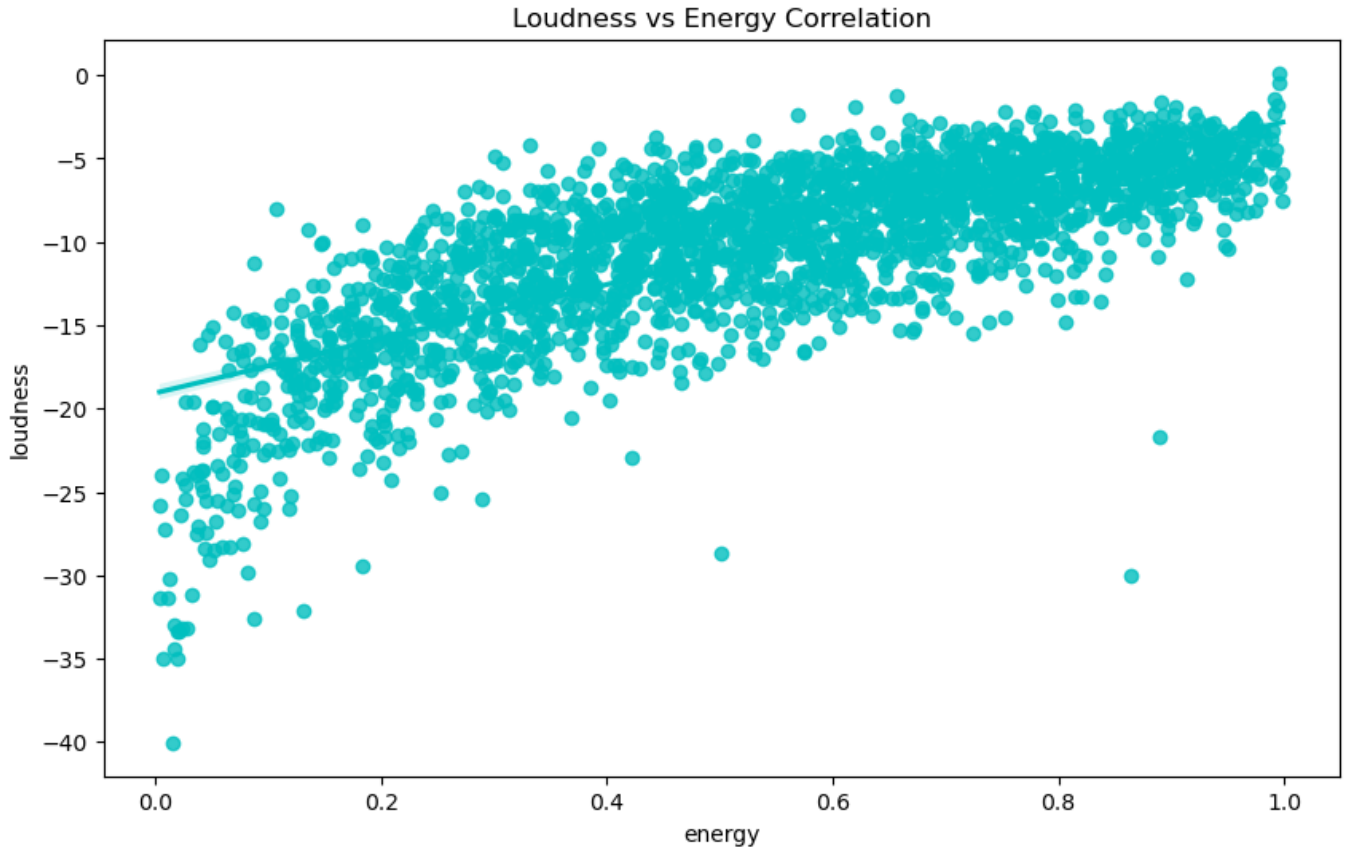
```
sample_df = df_tracks.sample(int(0.004*len(df_tracks)))
print(len(sample_df))

plt.figure(figsize = (10,6))
sns.regplot(data = sample_df, y = "loudness", x = "energy", color = "c")
plt.title("Loudness vs Energy Correlation")
```

```
2346
```

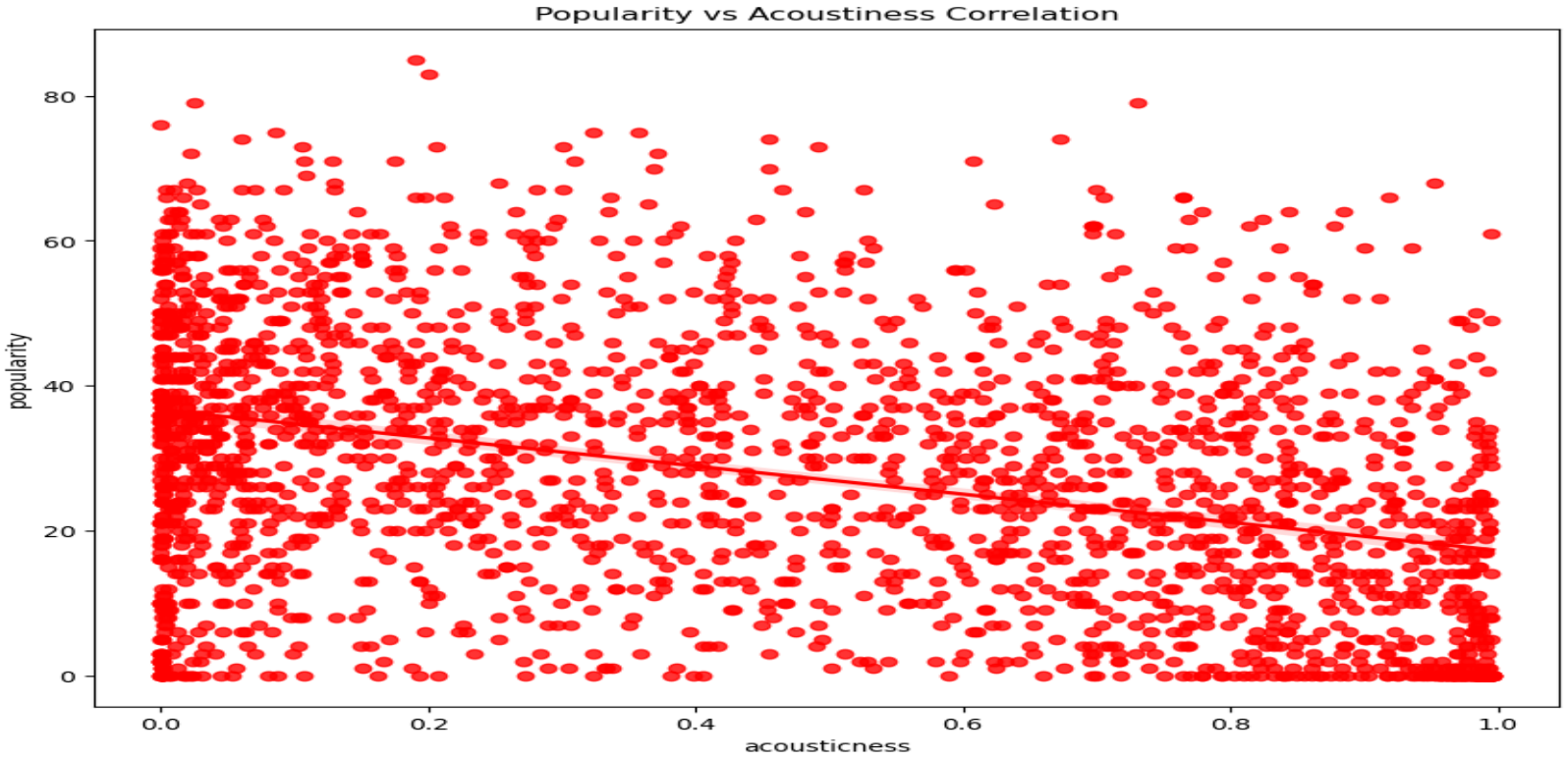
[16]:

```
Text(0.5, 1.0, 'Loudness vs Energy Correlation')
```



```
plt.figure(figsize = (10,8))
sns.regplot(data = sample_df, y = "popularity", x = "acousticness", color = "red")
plt.title("Popularity vs Acoustiness Correlation")
```

```
Text(0.5, 1.0, 'Popularity vs Acoustiness Correlation')
```

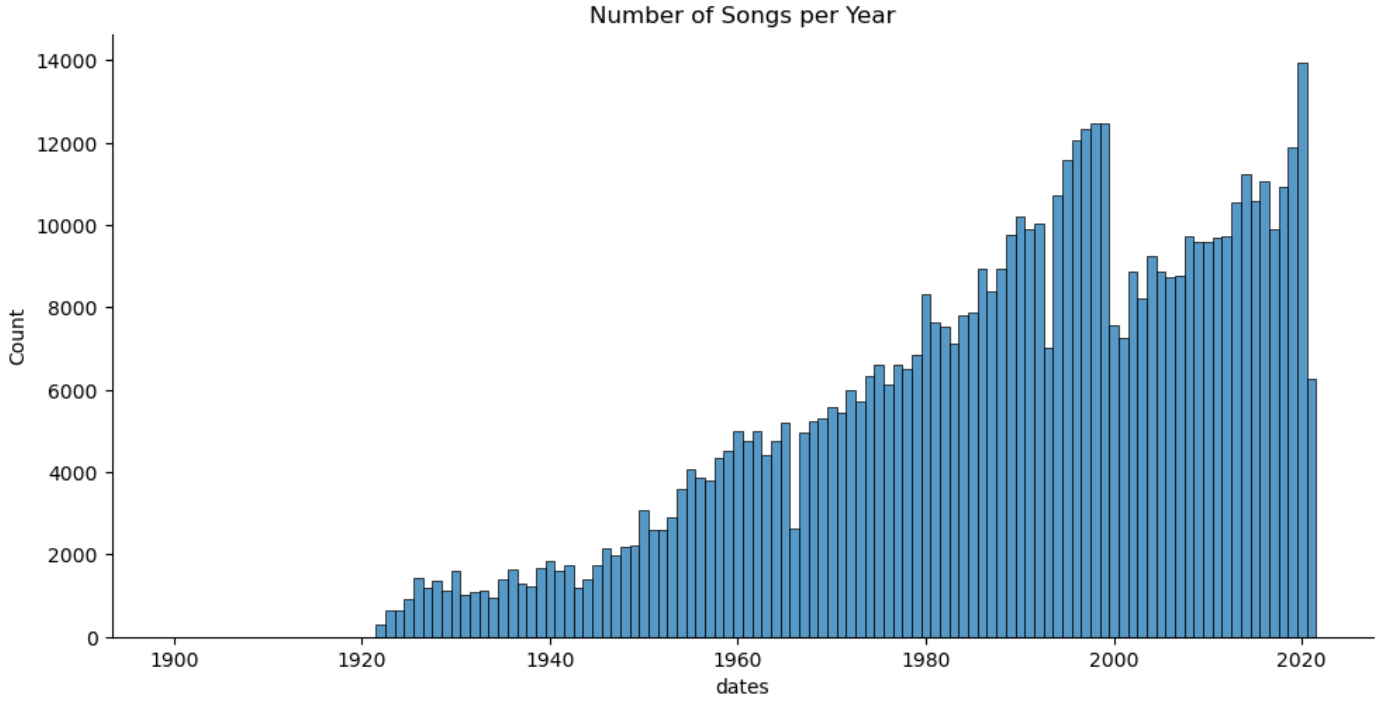


```
df_tracks['dates'] = df_tracks.index.get_level_values('release_date')
df_tracks.dates = pd.to_datetime(df_tracks.dates)
years = df_tracks.dates.dt.year
```

```
sns.displot(years, discrete = True, aspect = 2, height = 5, kind = "hist")
plt.title("Number of Songs per Year")
```

[19]:

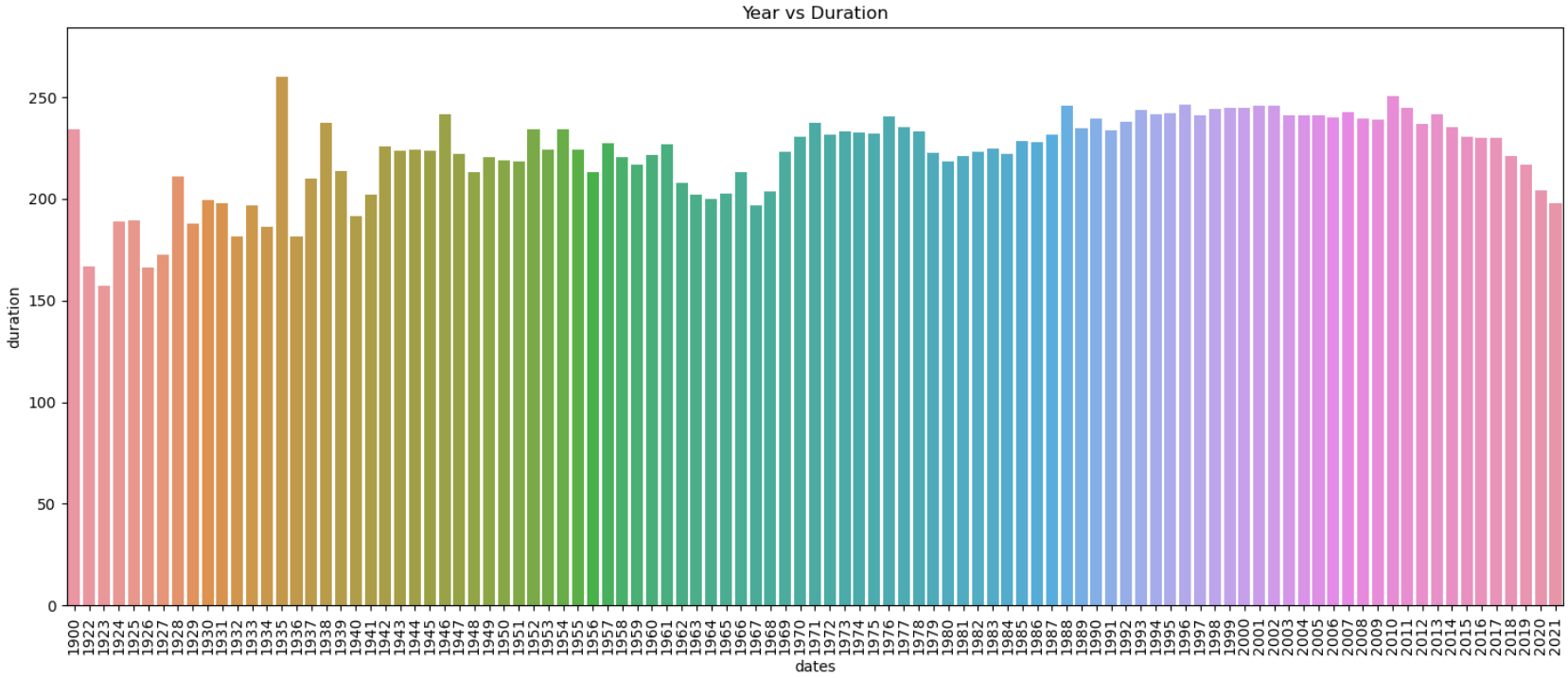
Text(0.5, 1.0, 'Number of Songs per Year')



```
total_dr = df_tracks.duration
fig_dims = (18,7)
fig, ax = plt.subplots(figsize = fig_dims)
fig = sns.barplot(x = years, y = total_dr, ax = ax, errwidth = False)
plt.title("Year vs Duration")
plt.xticks(rotation = 90)
```

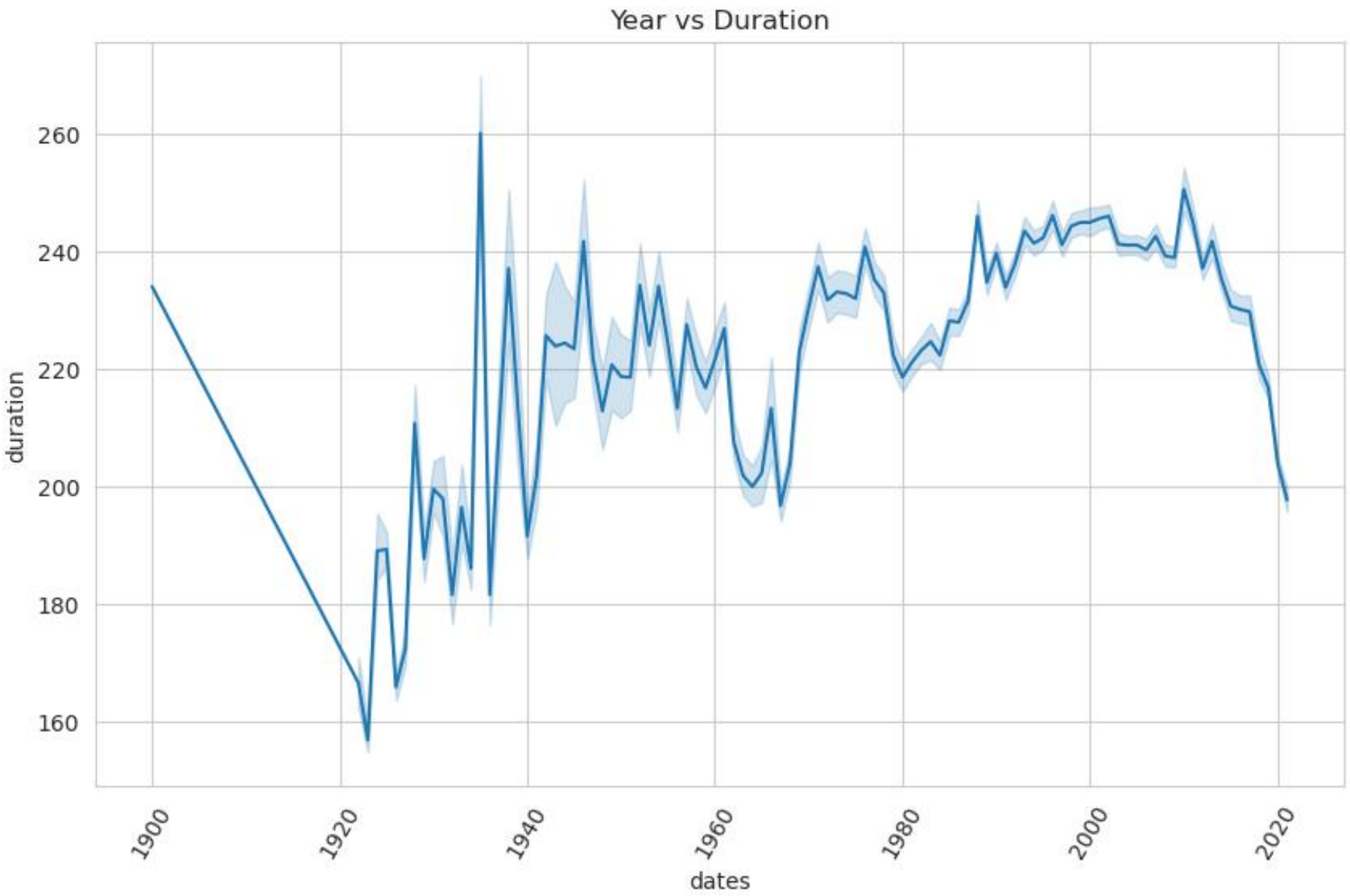
[20]:

```
(array([  0,   1,   2,   3,   4,   5,   6,   7,   8,   9,  10,  11,  12,
         13,  14,  15,  16,  17,  18,  19,  20,  21,  22,  23,  24,  25,
         26,  27,  28,  29,  30,  31,  32,  33,  34,  35,  36,  37,  38,
         39,  40,  41,  42,  43,  44,  45,  46,  47,  48,  49,  50,  51,
         52,  53,  54,  55,  56,  57,  58,  59,  60,  61,  62,  63,  64,
         65,  66,  67,  68,  69,  70,  71,  72,  73,  74,  75,  76,  77,
         78,  79,  80,  81,  82,  83,  84,  85,  86,  87,  88,  89,  90,
         91,  92,  93,  94,  95,  96,  97,  98,  99, 100]),
```



```
total_dr = df_tracks.duration
sns.set_style(style = "whitegrid")
fig_dims = (10,6)
fig, ax = plt.subplots(figsize = fig_dims)
fig = sns.lineplot(x = years, y = total_dr, ax = ax)
plt.title("Year vs Duration")
plt.xticks(rotation = 60)
```

```
(array([1880., 1900., 1920., 1940., 1960., 1980., 2000., 2020., 2040.]),
[Text(1880.0, 0, '1880'),
 Text(1900.0, 0, '1900'),
 Text(1920.0, 0, '1920'),
 Text(1940.0, 0, '1940'),
 Text(1960.0, 0, '1960'),
 Text(1980.0, 0, '1980'),
 Text(2000.0, 0, '2000'),
 Text(2020.0, 0, '2020'),
 Text(2040.0, 0, '2040')])
```


Year vs Duration
duration
260
240
220
200
180
160
1900
1920
1940
1960
1980
2000
2020
dates

```
df_features = pd.read_csv('/kaggle/input/spotify-features/SpotifyFeatures.csv')
```

add Codeadd Markdown

```
df_features.head()
```

[23]:

df_features.info()

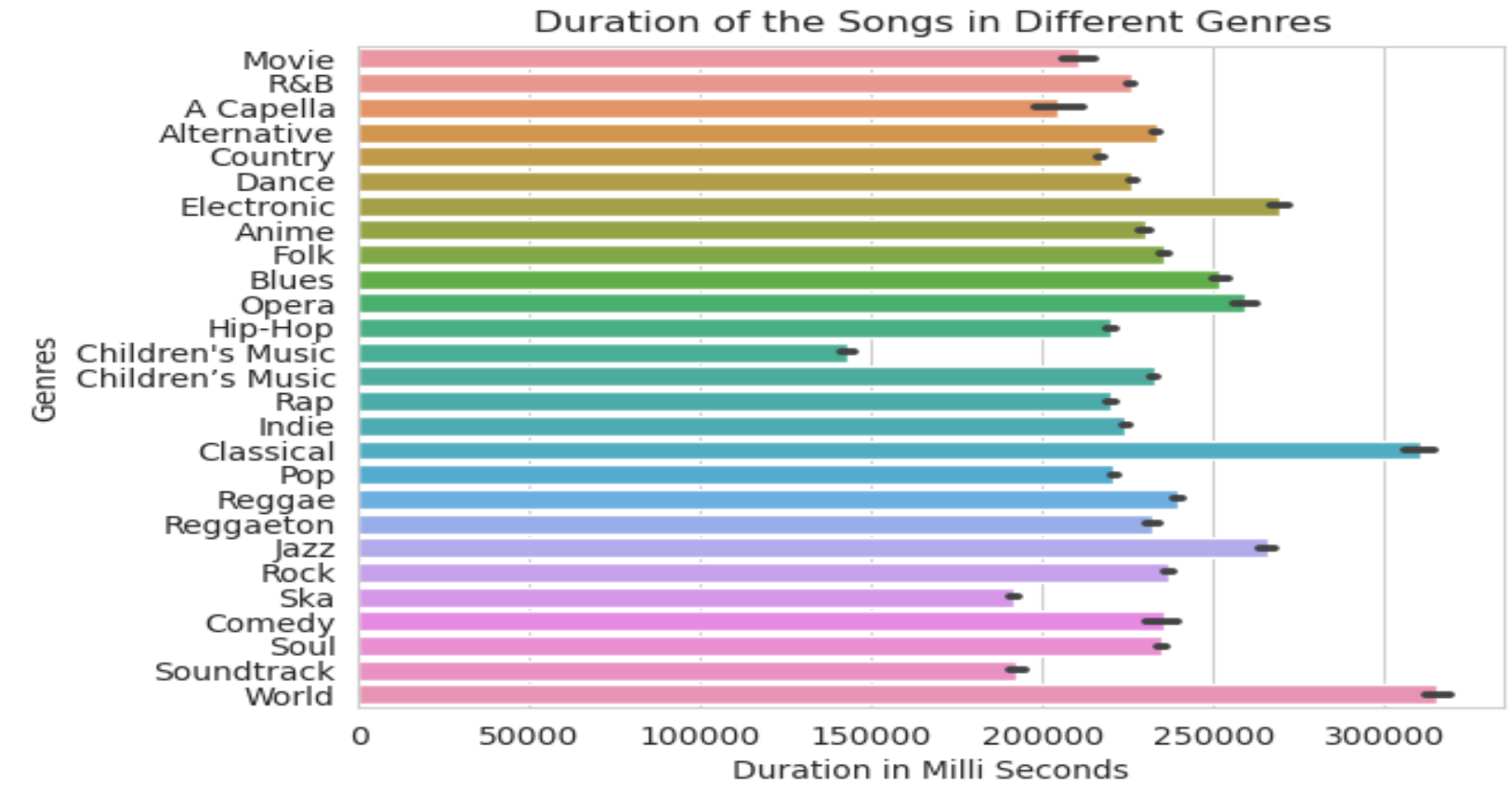
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 232725 entries, 0 to 232724
Data columns (total 18 columns):
 #   Column           Non-Null Count   Dtype  
---  ------           --------------   -----  
 0   genre            232725 non-null  object 
 1   artist_name      232725 non-null  object 
 2   track_name       232725 non-null  object 
 3   track_id         232725 non-null  object 
 4   popularity       232725 non-null  int64  
 5   acousticness     232725 non-null  float64
 6   danceability     232725 non-null  float64
 7   duration_ms      232725 non-null  int64  
 8   energy           232725 non-null  float64
 9   instrumentalness 232725 non-null  float64
 10  key              232725 non-null  object 
 11  liveness         232725 non-null  float64
 12  loudness         232725 non-null  float64
 13  mode             232725 non-null  object 
 14  speechiness      232725 non-null  float64
 15  tempo            232725 non-null  float64
 16  time_signature   232725 non-null  object 
 17  valence          232725 non-null  float64
dtypes: float64(9), int64(2), object(7)
memory usage: 32.0+ MB
```

```
pd.isnull(df_features).sum()
```

```
genre             0
artist_name       0
track_name        0
track_id          0
popularity        0
acousticness      0
danceability      0
duration_ms       0
energy            0
instrumentalness  0
key               0
liveness          0
loudness          0
mode              0
speechiness       0
tempo             0
time_signature    0
valence           0
dtype: int64
```

```
plt.title("Duration of the Songs in Different Genres")
sns.color_palette("rocket", as_cmap = True)
sns.barplot(y = 'genre', x = 'duration_ms', data = df_features)
plt.xlabel("Duration in Milli Seconds")
plt.ylabel("Genres")
```

Text(0, 0.5, 'Genres')



```
sns.set_style(style = "darkgrid")
plt.figure(figsize = (10,6))
famous = df_features.sort_values('popularity', ascending = False).head(10)
sns.barplot(y = 'genre', x = 'popularity', data = famous)
plt.title("Top 5 Genres by Popularity")
```

[27]:

```
Text(0.5, 1.0, 'Top 5 Genres by Popularity')
```