# spotify-song-analysis

April 14, 2024

[ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import csv
import pathlib as Path
```

[ ]:
```
df=pd.read_csv('tracks.csv', encoding='unicode_escape')
```

[ ]:
```
df.shape
```

[ ]:
```
(10144, 20)
```

[ ]:
```
df.head()
```

[ ]:
```
                       id                                  name  popularity  \
0  35iwgR4jXetI318WEWsa1Q                                 Carve         6.0
1  021ht4sdgPcrDgSk7JTbKY  CapÃtulo 2.16 - Banquero Anarquista         0.0
2  07A5yehtSnoedViJAZkNnc    Vivo para Quererte - Remasterizado         0.0
3  08FmqUhxtyLTn6pAh6bk45         El Prisionero - Remasterizado         0.0
4  08y9GfoqCWfOGsKdwojr5e                   Lady of the Evening         0.0

   duration_ms  explicit              artists                  id_artists  \
0     126903.0       0.0              ['Uli']  ['45tIt06XoI0Iio4LBEVpls']
1      98200.0       0.0  ['Fernando Pessoa']  ['14jtPCOoNZwquk5wd9DxrY']
2     181640.0       0.0  ['Ignacio Corsini']  ['5LiOoJbxVSAMkBS2fUm3X2']
3     176907.0       0.0  ['Ignacio Corsini']  ['5LiOoJbxVSAMkBS2fUm3X2']
4     163080.0       0.0      ['Dick Haymes']  ['3BiJGZsyX9sJchTqcSA7Su']

  release_date  danceability  energy  key  loudness  mode  speechiness  \
0   1922-02-22         0.645  0.4450  0.0   -13.338   1.0       0.4510
1   1922-06-01         0.695  0.2630  0.0   -22.136   1.0       0.9570
2   1922-03-21         0.434  0.1770  1.0   -21.180   1.0       0.0512
3   1922-03-21         0.321  0.0946  7.0   -27.961   1.0       0.0504
4         1922         0.402  0.1580  3.0   -16.900   0.0       0.0390
```

```
   acousticness  instrumentalness  liveness  valence    tempo  time_signature
0         0.674            0.7440     0.151    0.127  104.851             3.0
1         0.797            0.0000     0.148    0.655  102.009             1.0
2         0.994            0.0218     0.212    0.457  130.418             5.0
3         0.995            0.9180     0.104    0.397  169.980             3.0
4         0.989            0.1300     0.311    0.196  103.220             4.0
```

```
[ ]: pd.isnull(df)
```

```
[ ]:          id   name  popularity  duration_ms  explicit  artists  id_artists  \
0      False  False       False        False     False    False       False
1      False  False       False        False     False    False       False
2      False  False       False        False     False    False       False
3      False  False       False        False     False    False       False
4      False  False       False        False     False    False       False
…      …    …         …          …       …      …         …
10139  False  False       False        False     False    False       False
10140  False  False       False        False     False    False       False
10141  False  False       False        False     False    False       False
10142  False  False       False        False     False    False       False
10143  False   True        True         True      True     True        True

       release_date  danceability  energy    key  loudness   mode  \
0             False         False   False  False     False  False
1             False         False   False  False     False  False
2             False         False   False  False     False  False
3             False         False   False  False     False  False
4             False         False   False  False     False  False
…             …           …     …    …       …    …
10139         False         False   False  False     False  False
10140         False         False   False  False     False  False
10141         False         False   False  False     False  False
10142         False         False   False  False     False  False
10143          True          True    True   True      True   True

       speechiness  acousticness  instrumentalness  liveness  valence  tempo  \
0            False         False             False     False    False  False
1            False         False             False     False    False  False
2            False         False             False     False    False  False
3            False         False             False     False    False  False
4            False         False             False     False    False  False
…            …           …               …       …      …    …
10139        False         False             False     False    False  False
10140        False         False             False     False    False  False
10141        False         False             False     False    False  False
10142        False         False             False     False    False  False
10143         True          True              True      True     True   True
```

```
       time_signature
0               False
1               False
2               False
3               False
4               False
…                 …
10139           False
10140           False
10141           False
10142           False
10143            True

[10144 rows x 20 columns]
```

```
[ ]: pd.isnull(df).sum()
```

```
[ ]: id                  0
     name                1
     popularity          1
     duration_ms         1
     explicit            1
     artists             1
     id_artists          1
     release_date        1
     danceability        1
     energy              1
     key                 1
     loudness            1
     mode                1
     speechiness         1
     acousticness        1
     instrumentalness    1
     liveness            1
     valence             1
     tempo               1
     time_signature      1
     dtype: int64
```

```
[ ]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10144 entries, 0 to 10143
Data columns (total 20 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
```

```
 0   id                10144 non-null  object
 1   name              10143 non-null  object
 2   popularity        10143 non-null  float64
 3   duration_ms       10143 non-null  float64
 4   explicit          10143 non-null  float64
 5   artists           10143 non-null  object
 6   id_artists        10143 non-null  object
 7   release_date      10143 non-null  object
 8   danceability      10143 non-null  float64
 9   energy            10143 non-null  float64
 10  key               10143 non-null  float64
 11  loudness          10143 non-null  float64
 12  mode              10143 non-null  float64
 13  speechiness       10143 non-null  float64
 14  acousticness      10143 non-null  float64
 15  instrumentalness  10143 non-null  float64
 16  liveness          10143 non-null  float64
 17  valence           10143 non-null  float64
 18  tempo             10143 non-null  float64
 19  time_signature    10143 non-null  float64
dtypes: float64(15), object(5)
memory usage: 1.5+ MB
```

[ ]:

```
sort_df=df.sort_values('popularity',ascending=False).head(10)
sort_df
```

[ ]:

```
                          id  \
8095  1LGqJ3nvxpVXDWpEzq4DJD
4857  0RNxWy0PC3AyH4ThH3aGK6
8096  0Otf1ZfYNIjhqFIuJk0fsy
8097  62Xj6x2LSWv47PtRAXrg5Z
8098  3d3aPk50dBMA2YbA7lC92r
8099  41CHb7F7SXcmkj0h8wekeF
3857  61znp1Iy11bdJ2YAbwaqw7
9490  5qWXXF3On2xUfVu1ND1NLN
1857  3BFRqZFLSrqtQr6cjHbAxU
6505  71bkiosdVssjMrVDbdeG7n

                                                   name  popularity  \
8095                                          All of Me        65.0
4857                                     Mack the Knife        55.0
8096                                        Tea for Two        53.0
8097                                         Summertime        53.0
8098    Easy Living (with Teddy Wilson & His Orchestra)        51.0
8099  Gloomy Sunday (with Teddy Wilson & His Orchest…        51.0
3857                                   Sing, Sing, Sing        51.0
9490                                     Cheek to Cheek        51.0
```

```
1857                                  Ain't Misbehavin'        51.0
6505                                             Volver        50.0

      duration_ms  explicit                             artists  \
8095     181440.0       0.0                  ['Billie Holiday']
4857     201467.0       0.0                 ['Louis Armstrong']
8096     193867.0       0.0                       ['Art Tatum']
8097     173493.0       0.0                  ['Billie Holiday']
8098     182400.0       0.0  ['Billie Holiday', 'Teddy Wilson']
8099     190800.0       0.0  ['Billie Holiday', 'Teddy Wilson']
3857     520133.0       0.0                   ['Benny Goodman']
9490     198107.0       0.0                    ['Fred Astaire']
1857     237773.0       0.0                     ['Fats Waller']
6505     172613.0       0.0                   ['Carlos Gardel']

                                             id_artists release_date  \
8095                         ['1YzCsTRb22dQkh9lghPIrp']         1933
4857                         ['19eLuQmk9aCobbVDHc6eek']         1929
8096                         ['3DtSOCXYU6o4EV0K1NgIKq']         1933
8097                         ['1YzCsTRb22dQkh9lghPIrp']         1933
8098  ['1YzCsTRb22dQkh9lghPIrp', '0tg5uVI4VjzZOFzBry…         1933
8099  ['1YzCsTRb22dQkh9lghPIrp', '0tg5uVI4VjzZOFzBry…         1933
3857                         ['1pBuKaLHJlIlqYxQQaflve']         1928
9490                         ['4BtDAwCZhR6nPrJtbVgQNX']         1935
1857                         ['0DYWCXTkNqGFZIf67SrWEa']         1926
6505                         ['05Q9xndTxhXhD5trpmTtfU']   1931-01-01

      danceability  energy   key  loudness  mode  speechiness  acousticness  \
8095         0.504  0.0644   2.0   -14.754   0.0       0.0408         0.972
4857         0.673  0.3770   0.0   -14.141   1.0       0.0697         0.586
8096         0.358  0.4520   1.0   -13.020   1.0       0.0347         0.991
8097         0.628  0.2260  10.0   -14.075   0.0       0.0371         0.903
8098         0.640  0.2300   8.0   -10.651   1.0       0.0624         0.965
8099         0.484  0.0823   7.0   -11.273   0.0       0.0364         0.980
3857         0.626  0.7440   2.0    -9.189   0.0       0.0662         0.847
9490         0.506  0.1800   0.0   -12.184   1.0       0.0443         0.908
1857         0.515  0.2220   0.0   -16.918   0.0       0.0575         0.821
6505         0.482  0.2870   1.0    -9.108   0.0       0.0482         0.981

      instrumentalness  liveness  valence    tempo  time_signature
8095          0.000004    0.1740    0.403  106.994             5.0
4857          0.000000    0.3320    0.713   88.973             4.0
8096          0.864000    0.1400    0.525  166.973             4.0
8097          0.000006    0.1710    0.608   99.392             4.0
8098          0.000054    0.1010    0.663   88.868             4.0
8099          0.000002    0.1590    0.191  127.089             3.0
3857          0.892000    0.1450    0.259  113.117             4.0
```

```
9490           0.000002    0.0652     0.665  161.510             4.0
1857           0.001930    0.1900     0.350   98.358             4.0
6505           0.000017    0.3000     0.536   69.804             4.0
```

```
[ ]: df.describe().transpose()
```

```
[ ]:                     count           mean           std     min          25%  \
     popularity        10143.0       1.834270      5.128216     0.0       0.0000
     duration_ms       10143.0  188537.323671  97789.129824  3344.0  151399.5000
     explicit          10143.0       0.003155      0.056082     0.0       0.0000
     danceability      10143.0       0.586564      0.163355     0.0       0.4770
     energy            10143.0       0.264970      0.151218     0.0       0.1650
     key               10143.0       4.925762      3.469692     0.0       2.0000
     loudness          10143.0     -15.375837      5.401065   -60.0     -19.3475
     mode              10143.0       0.719807      0.449116     0.0       0.0000
     speechiness       10143.0       0.284304      0.365090     0.0       0.0462
     acousticness      10143.0       0.872828      0.196546     0.0       0.8395
     instrumentalness  10143.0       0.277411      0.372569     0.0       0.0000
     liveness          10143.0       0.210923      0.149883     0.0       0.1120
     valence           10143.0       0.585615      0.235069     0.0       0.4150
     tempo             10143.0     111.348108     31.079018     0.0      86.6525
     time_signature    10143.0       3.752736      0.698328     0.0       4.0000

                                50%          75%          max
     popularity            0.00000       1.0000       65.000
     duration_ms       177733.00000  194873.5000  3093226.000
     explicit              0.00000       0.0000        1.000
     danceability          0.62800       0.7100        0.956
     energy                0.24100       0.3270        1.000
     key                   5.00000       8.0000       11.000
     loudness            -14.98400     -11.4750        0.674
     mode                  1.00000       1.0000        1.000
     speechiness           0.07560       0.3530        0.969
     acousticness          0.96900       0.9920        0.996
     instrumentalness      0.00759       0.6885        0.993
     liveness              0.15800       0.2650        0.973
     valence               0.61700       0.7720        0.986
     tempo               111.15900     126.6480      221.741
     time_signature        4.00000       4.0000        5.000
```

```
[ ]: most_popular=df.query('popularity>50',inplace=False).
       ↪sort_values('popularity',ascending=False)
     most_popular[:10]
```

```
[ ]:                           id  \
     8095  1LGqJ3nvxpVXDWpEzq4DJD
     4857  0RNxWy0PC3AyH4ThH3aGK6
```

```
8096  0Otf1ZfYNIjhqFIuJk0fsy
8097  62Xj6x2LSWv47PtRAXrg5Z
1857  3BFRqZFLSrqtQr6cjHbAxU
3857  61znp1Iy11bdJ2YAbwaqw7
8098  3d3aPk50dBMA2YbA7lC92r
8099  41CHb7F7SXcmkj0h8wekeF
9490  5qWXXF3On2xUfVu1ND1NLN

                                                   name  popularity  \
8095                                          All of Me        65.0   
4857                                     Mack the Knife        55.0   
8096                                        Tea for Two        53.0   
8097                                         Summertime        53.0   
1857                                  Ain't Misbehavin'        51.0   
3857                                   Sing, Sing, Sing        51.0   
8098    Easy Living (with Teddy Wilson & His Orchestra)        51.0   
8099  Gloomy Sunday (with Teddy Wilson & His Orchest…        51.0   
9490                                     Cheek to Cheek        51.0   

      duration_ms  explicit                             artists  \
8095     181440.0       0.0                  ['Billie Holiday']   
4857     201467.0       0.0                 ['Louis Armstrong']   
8096     193867.0       0.0                       ['Art Tatum']   
8097     173493.0       0.0                  ['Billie Holiday']   
1857     237773.0       0.0                     ['Fats Waller']   
3857     520133.0       0.0                   ['Benny Goodman']   
8098     182400.0       0.0  ['Billie Holiday', 'Teddy Wilson']   
8099     190800.0       0.0  ['Billie Holiday', 'Teddy Wilson']   
9490     198107.0       0.0                    ['Fred Astaire']   

                                             id_artists release_date  \
8095                         ['1YzCsTRb22dQkh9lghPIrp']         1933   
4857                         ['19eLuQmk9aCobbVDHc6eek']         1929   
8096                         ['3DtSOCXYU6o4EV0K1NgIKq']         1933   
8097                         ['1YzCsTRb22dQkh9lghPIrp']         1933   
1857                         ['0DYWCXTkNqGFZIf67SrWEa']         1926   
3857                         ['1pBuKaLHJlIlqYxQQaflve']         1928   
8098  ['1YzCsTRb22dQkh9lghPIrp', '0tg5uVI4VjzZOFzBry…        1933   
8099  ['1YzCsTRb22dQkh9lghPIrp', '0tg5uVI4VjzZOFzBry…        1933   
9490                         ['4BtDAwCZhR6nPrJtbVgQNX']         1935   

      danceability  energy   key  loudness  mode  speechiness  acousticness  \
8095         0.504  0.0644   2.0   -14.754   0.0       0.0408         0.972   
4857         0.673  0.3770   0.0   -14.141   1.0       0.0697         0.586   
8096         0.358  0.4520   1.0   -13.020   1.0       0.0347         0.991   
8097         0.628  0.2260  10.0   -14.075   0.0       0.0371         0.903   
1857         0.515  0.2220   0.0   -16.918   0.0       0.0575         0.821   
```

```
3857         0.626  0.7440   2.0    -9.189   0.0       0.0662         0.847
8098         0.640  0.2300   8.0   -10.651   1.0       0.0624         0.965
8099         0.484  0.0823   7.0   -11.273   0.0       0.0364         0.980
9490         0.506  0.1800   0.0   -12.184   1.0       0.0443         0.908

      instrumentalness  liveness  valence    tempo  time_signature
8095          0.000004    0.1740    0.403  106.994             5.0
4857          0.000000    0.3320    0.713   88.973             4.0
8096          0.864000    0.1400    0.525  166.973             4.0
8097          0.000006    0.1710    0.608   99.392             4.0
1857          0.001930    0.1900    0.350   98.358             4.0
3857          0.892000    0.1450    0.259  113.117             4.0
8098          0.000054    0.1010    0.663   88.868             4.0
8099          0.000002    0.1590    0.191  127.089             3.0
9490          0.000002    0.0652    0.665  161.510             4.0
```

```
[ ]: df['popularity'].dtypes
```

```
[ ]: dtype('float64')
```

```
[ ]: df['popularity'].describe()
```

```
[ ]: count    10143.000000
     mean         1.834270
     std          5.128216
     min          0.000000
     25%          0.000000
     50%          0.000000
     75%          1.000000
     max         65.000000
     Name: popularity, dtype: float64
```

```
[ ]: df.set_index("release_date",inplace=True)
     df.index=pd.to_datetime(df.index)
     df.head()
```

```
[ ]:                                   id                                  name  \
     release_date
     1922-02-22    35iwgR4jXetI318WEWsa1Q                                 Carve
     1922-06-01    021ht4sdgPcrDgSk7JTbKY  CapÃtulo 2.16 - Banquero Anarquista
     1922-03-21    07A5yehtSnoedViJAZkNnc    Vivo para Quererte - Remasterizado
     1922-03-21    08FmqUhxtyLTn6pAh6bk45         El Prisionero - Remasterizado
     1922-01-01    08y9GfoqCWfOGsKdwojr5e                   Lady of the Evening

                   popularity  duration_ms  explicit              artists  \
     release_date
     1922-02-22           6.0     126903.0       0.0              ['Uli']
```

```
1922-06-01           0.0      98200.0       0.0  ['Fernando Pessoa']
1922-03-21           0.0     181640.0       0.0  ['Ignacio Corsini']
1922-03-21           0.0     176907.0       0.0  ['Ignacio Corsini']
1922-01-01           0.0     163080.0       0.0      ['Dick Haymes']

                              id_artists  danceability  energy  key  loudness  \
release_date
1922-02-22    ['45tIt06XoI0Iio4LBEVpls']         0.645  0.4450  0.0   -13.338
1922-06-01    ['14jtPCOoNZwquk5wd9DxrY']         0.695  0.2630  0.0   -22.136
1922-03-21    ['5LiOoJbxVSAMkBS2fUm3X2']         0.434  0.1770  1.0   -21.180
1922-03-21    ['5LiOoJbxVSAMkBS2fUm3X2']         0.321  0.0946  7.0   -27.961
1922-01-01    ['3BiJGZsyX9sJchTqcSA7Su']         0.402  0.1580  3.0   -16.900

              mode  speechiness  acousticness  instrumentalness  liveness  \
release_date
1922-02-22     1.0       0.4510         0.674            0.7440     0.151
1922-06-01     1.0       0.9570         0.797            0.0000     0.148
1922-03-21     1.0       0.0512         0.994            0.0218     0.212
1922-03-21     1.0       0.0504         0.995            0.9180     0.104
1922-01-01     0.0       0.0390         0.989            0.1300     0.311

              valence    tempo  time_signature
release_date
1922-02-22      0.127  104.851             3.0
1922-06-01      0.655  102.009             1.0
1922-03-21      0.457  130.418             5.0
1922-03-21      0.397  169.980             3.0
1922-01-01      0.196  103.220             4.0
```

```
[ ]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 10144 entries, 1922-02-22 to NaT
Data columns (total 19 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   id                10144 non-null  object
 1   name              10143 non-null  object
 2   popularity        10143 non-null  float64
 3   duration_ms       10143 non-null  float64
 4   explicit          10143 non-null  float64
 5   artists           10143 non-null  object
 6   id_artists        10143 non-null  object
 7   danceability      10143 non-null  float64
 8   energy            10143 non-null  float64
 9   key               10143 non-null  float64
 10  loudness          10143 non-null  float64
 11  mode              10143 non-null  float64
 12  speechiness       10143 non-null  float64
 13  acousticness      10143 non-null  float64
 14  instrumentalness  10143 non-null  float64
 15  liveness          10143 non-null  float64
 16  valence           10143 non-null  float64
 17  tempo             10143 non-null  float64
 18  time_signature    10143 non-null  float64
dtypes: float64(15), object(4)
memory usage: 1.5+ MB
```

```
[ ]: df[['artists']].iloc[18]
```

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

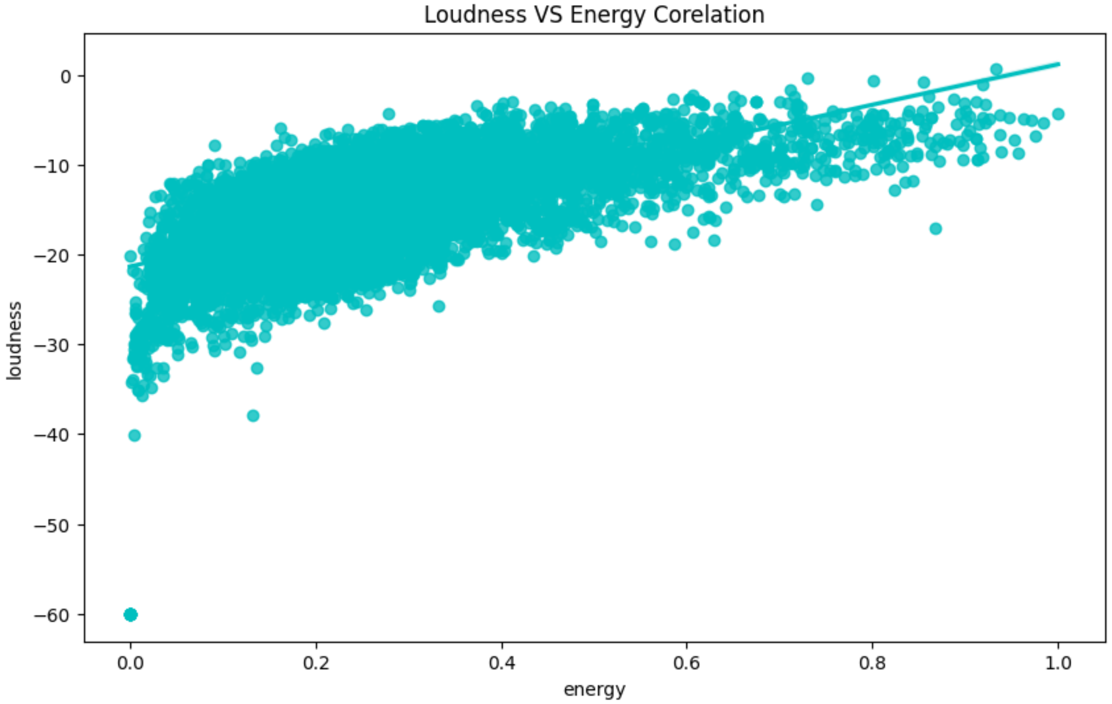
```
[ ]: df['duration_seconds'] = df['duration_ms'].apply(lambda x: round(x / 1000))
     df.drop('duration_ms', inplace=True, axis=1)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-43-146d1bc49d5a> in <cell line: 1>()
----> 1 df['duration_seconds'] = df['duration_ms'].apply(lambda x: round(x /␣
 ↪1000))
      2 df.drop('duration_ms', inplace=True, axis=1)

/usr/local/lib/python3.10/dist-packages/pandas/core/series.py in apply(self,␣
 ↪func, convert_dtype, args, **kwargs)
   4769         dtype: float64
   4770         """
-> 4771         return SeriesApply(self, func, convert_dtype, args, kwargs).
 ↪apply()
   4772
   4773     def _reduce(

/usr/local/lib/python3.10/dist-packages/pandas/core/apply.py in apply(self)
   1121
   1122         # self.f is Callable
-> 1123         return self.apply_standard()
   1124
   1125     def agg(self):

/usr/local/lib/python3.10/dist-packages/pandas/core/apply.py in␣
 ↪apply_standard(self)
   1172             else:
   1173                 values = obj.astype(object)._values
```

```
-> 1174                 mapped = lib.map_infer(

   1175                     values,
   1176                     f,

/usr/local/lib/python3.10/dist-packages/pandas/_libs/lib.pyx in pandas._libs.lib.
 ↪map_infer()

<ipython-input-43-146d1bc49d5a> in <lambda>(x)
----> 1 df['duration_seconds'] = df['duration_ms'].apply(lambda x: round(x /␣
 ↪1000))
      2 df.drop('duration_ms', inplace=True, axis=1)

ValueError: cannot convert float NaN to integer
```

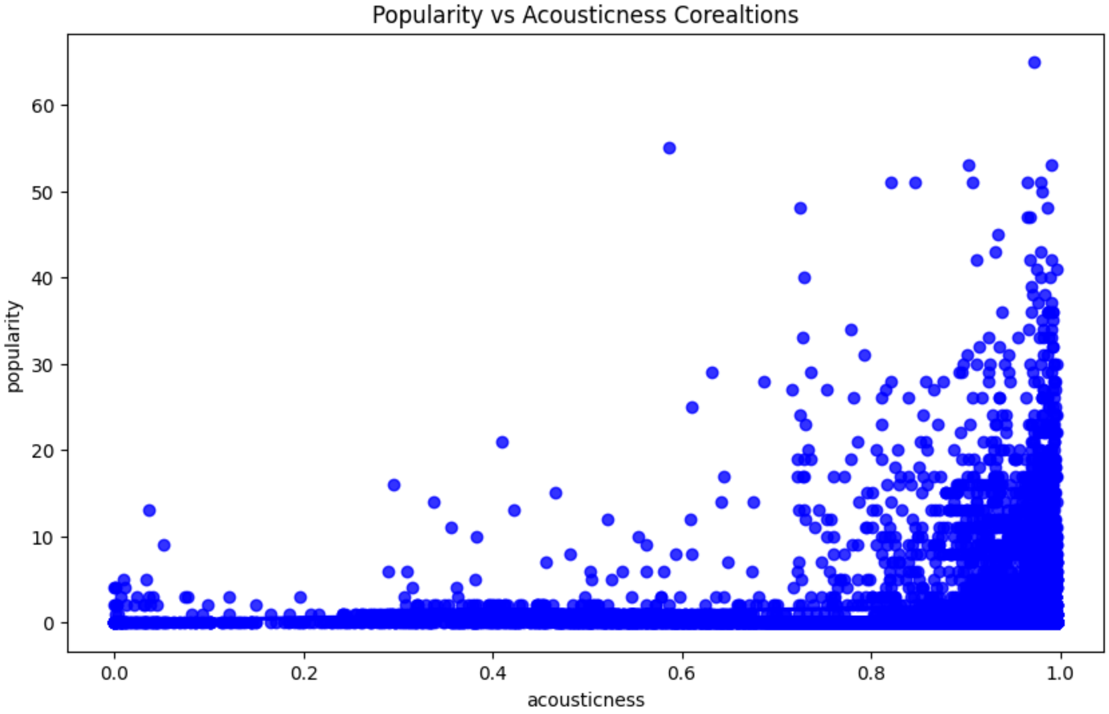
[ ]:
```
plt.figure(figsize=(10,6))
sns.regplot(data=df,y='loudness',x='energy',color='c').set(title="Loudness VS␣
 ↪Energy Corelation")
```

[ ]:
```
[Text(0.5, 1.0, 'Loudness VS Energy Corelation')]
```



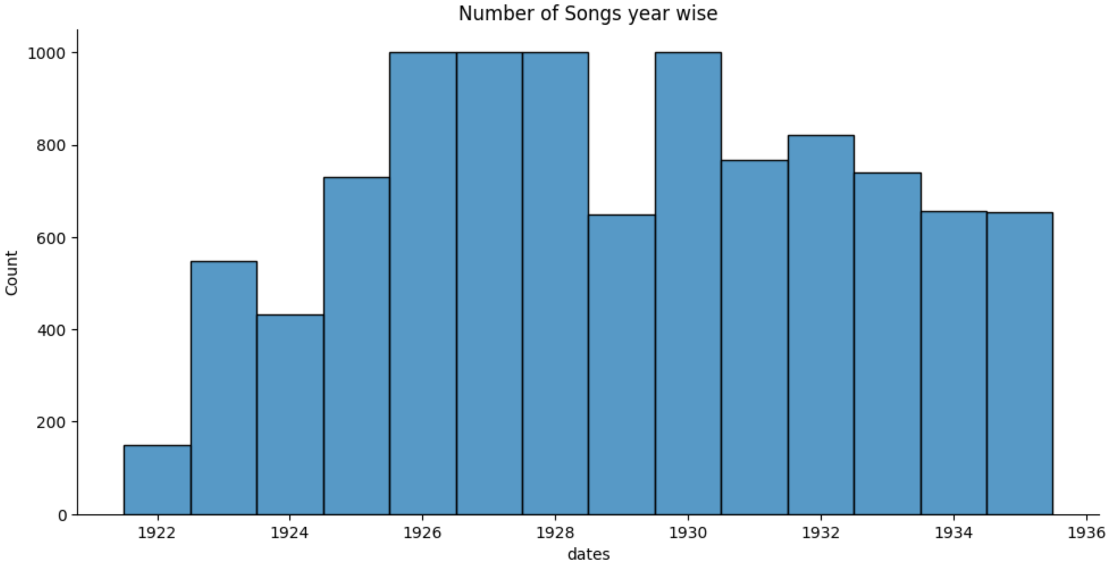
[ ]:
```
plt.figure(figsize=(10,6))
sns.regplot(data=df,y='popularity',x='acousticness',color='b').
 ↪set(title='Popularity vs Acousticness Corealtions')
```

```
[ ]: [Text(0.5, 1.0, 'Popularity vs Acousticness Corealtions')]
```



```
[ ]: sns.displot(years,discrete=True,aspect=2,height=5,kind='hist').
      ↪set(title="Number of Songs year wise")
```

```
[ ]: <seaborn.axisgrid.FacetGrid at 0x7b6a8b388580>
```



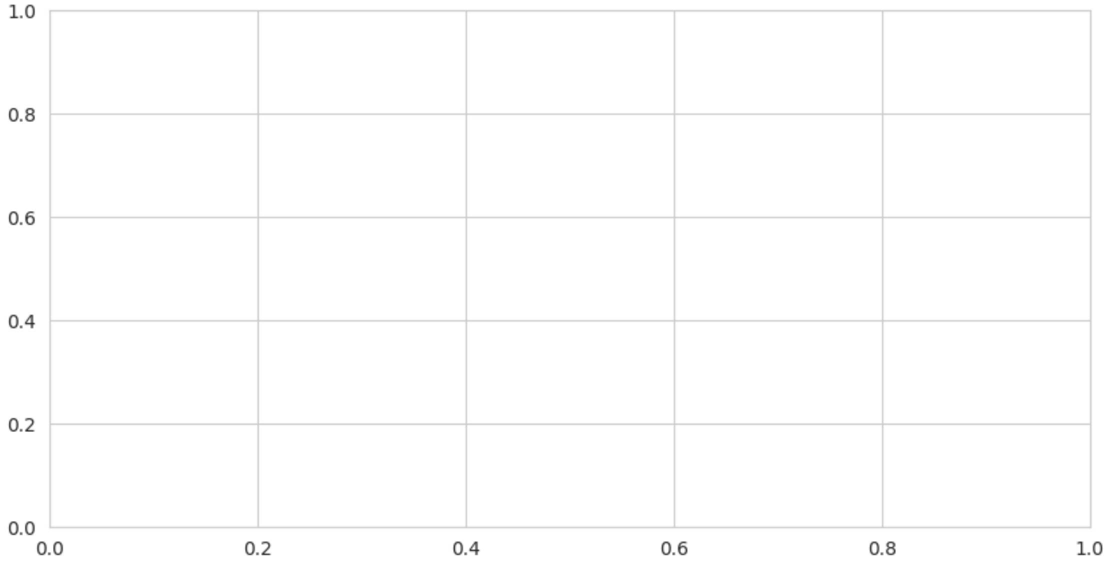
```
[ ]: df['dates']=df.index.get_level_values('release_date')
     df['dates']=pd.to_datetime(df.dates)
     years=df.dates.dt.year
```

```
[ ]: total_dr=df.duration_ms
     sns.set_style(style='whitegrid')
     fig_dims=(10,5)
     fig, ax=plt.subplots(figsize=fig_dims)
     fig=sns.lineplot(x=years,y=total_dr,ax=ax).set(title="Years vs Duration")
     plt.xticks(rotattion=60)
```

```
/usr/local/lib/python3.10/dist-packages/seaborn/_base.py:1007: FutureWarning:
reindexing with a non-unique Index is deprecated and will raise in a future
version.
  comp_data.insert(0, var, comp_col)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-62-09a8140c4445> in <cell line: 5>()
      3 fig_dims=(10,5)
      4 fig, ax=plt.subplots(figsize=fig_dims)
----> 5 fig=sns.lineplot(x=years,y=total_dr,ax=ax).set(title="Years vs Duration")
      6 plt.xticks(rotattion=60)

/usr/local/lib/python3.10/dist-packages/seaborn/relational.py in lineplot(data,␣
  ↪x, y, hue, size, style, units, weights, palette, hue_order, hue_norm, sizes,␣
  ↪size_order, size_norm, dashes, markers, style_order, estimator, errorbar,␣
  ↪n_boot, seed, orient, sort, err_style, err_kws, legend, ci, ax, **kwargs)
    508     kwargs["color"] = _default_color(ax.plot, hue, color, kwargs)
    509
--> 510     p.plot(ax, kwargs)
    511     return ax
    512

/usr/local/lib/python3.10/dist-packages/seaborn/relational.py in plot(self, ax,␣
  ↪kws)
    273         # Loop over the semantic subsets and add to the plot
    274         grouping_vars = "hue", "size", "style"
--> 275         for sub_vars, sub_data in self.iter_data(grouping_vars,␣
  ↪from_comp_data=True):
    276
    277             if self.sort:

/usr/local/lib/python3.10/dist-packages/seaborn/_base.py in iter_data(self,␣
  ↪grouping_vars, reverse, from_comp_data, by_facet, allow_empty, dropna)
    900
    901         if from_comp_data:
--> 902             data = self.comp_data
    903         else:
    904             data = self.plot_data

/usr/local/lib/python3.10/dist-packages/seaborn/_base.py in comp_data(self)
   1005                 else:
   1006                     comp_col = pd.Series(dtype=float, name=var)
-> 1007                 comp_data.insert(0, var, comp_col)
   1008 
   1009             self._comp_data = comp_data

/usr/local/lib/python3.10/dist-packages/pandas/core/frame.py in insert(self,␣
 ↪loc, column, value, allow_duplicates)
   4819             raise TypeError("loc must be int")
   4820 
-> 4821         value = self._sanitize_column(value)
   4822         self._mgr.insert(loc, column, value)
   4823 

/usr/local/lib/python3.10/dist-packages/pandas/core/frame.py in␣
 ↪_sanitize_column(self, value)
   4910             return _reindex_for_setitem(value, self.index)
   4911         elif is_dict_like(value):
-> 4912             return _reindex_for_setitem(Series(value), self.index)
   4913 
   4914         if is_list_like(value):

/usr/local/lib/python3.10/dist-packages/pandas/core/frame.py in␣
 ↪_reindex_for_setitem(value, index)
  12023         if not value.index.is_unique:
  12024             # duplicate axis
> 12025             raise err
  12026 
  12027         raise TypeError(

/usr/local/lib/python3.10/dist-packages/pandas/core/frame.py in␣
 ↪_reindex_for_setitem(value, index)
  12018     # GH#4107
  12019     try:
> 12020         reindexed_value = value.reindex(index)._values
  12021     except ValueError as err:
  12022         # raised in MultiIndex.from_tuples, see test_insert_error_msmgs

/usr/local/lib/python3.10/dist-packages/pandas/core/series.py in reindex(self,␣
 ↪*args, **kwargs)
   5092                 )
   5093             kwargs.update({"index": index})
```

```
-> 5094         return super().reindex(**kwargs)
   5095
   5096     @overload

/usr/local/lib/python3.10/dist-packages/pandas/core/generic.py in reindex(self,␣
 ↪*args, **kwargs)
   5287
   5288         # perform the reindex on the axes
-> 5289         return self._reindex_axes(

   5290             axes, level, limit, tolerance, method, fill_value, copy
   5291         ).__finalize__(self, method="reindex")

/usr/local/lib/python3.10/dist-packages/pandas/core/generic.py in␣
 ↪_reindex_axes(self, axes, level, limit, tolerance, method, fill_value, copy)
   5307
   5308             axis = self._get_axis_number(a)
-> 5309             obj = obj._reindex_with_indexers(

   5310                 {axis: [new_index, indexer]},
   5311                 fill_value=fill_value,

/usr/local/lib/python3.10/dist-packages/pandas/core/generic.py in␣
 ↪_reindex_with_indexers(self, reindexers, fill_value, copy, allow_dups)
   5353
   5354             # TODO: speed up on homogeneous DataFrame objects (see␣
 ↪_reindex_multi)
-> 5355             new_data = new_data.reindex_indexer(

   5356                 index,
   5357                 indexer,

/usr/local/lib/python3.10/dist-packages/pandas/core/internals/managers.py in␣
 ↪reindex_indexer(self, new_axis, indexer, axis, fill_value, allow_dups, copy,␣
 ↪only_slice, use_na_proxy)
    735         # some axes don't allow reindexing with dups
    736         if not allow_dups:
--> 737             self.axes[axis]._validate_can_reindex(indexer)
    738
    739         if axis >= self.ndim:

/usr/local/lib/python3.10/dist-packages/pandas/core/indexes/base.py in␣
 ↪_validate_can_reindex(self, indexer)
   4314         # trying to reindex on an axis with duplicates
   4315         if not self._index_as_unique and len(indexer):
-> 4316             raise ValueError("cannot reindex on an axis with duplicate␣
 ↪labels")
   4317
   4318     def reindex(
ValueError: cannot reindex on an axis with duplicate labels
```



```
[ ]: pip install --user seaborn==0.11.0
```

```
Collecting seaborn==0.11.0
  Downloading seaborn-0.11.0-py3-none-any.whl (283 kB)
                           283.1/283.1

kB 3.6 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.15 in
/usr/local/lib/python3.10/dist-packages (from seaborn==0.11.0) (1.25.2)
Requirement already satisfied: scipy>=1.0 in /usr/local/lib/python3.10/dist-
packages (from seaborn==0.11.0) (1.11.4)
Requirement already satisfied: pandas>=0.23 in /usr/local/lib/python3.10/dist-
packages (from seaborn==0.11.0) (1.5.3)
Requirement already satisfied: matplotlib>=2.2 in
/usr/local/lib/python3.10/dist-packages (from seaborn==0.11.0) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=2.2->seaborn==0.11.0)
(1.2.0)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-
packages (from matplotlib>=2.2->seaborn==0.11.0) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=2.2->seaborn==0.11.0)
(4.50.0)
Requirement already satisfied: kiwisolver>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=2.2->seaborn==0.11.0)
```

```
(1.4.5)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=2.2->seaborn==0.11.0)
(24.0)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-
packages (from matplotlib>=2.2->seaborn==0.11.0) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=2.2->seaborn==0.11.0)
(3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in
/usr/local/lib/python3.10/dist-packages (from matplotlib>=2.2->seaborn==0.11.0)
(2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-
packages (from pandas>=0.23->seaborn==0.11.0) (2023.4)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-
packages (from python-dateutil>=2.7->matplotlib>=2.2->seaborn==0.11.0) (1.16.0)
Installing collected packages: seaborn
Successfully installed seaborn-0.11.0
```

```
[ ]: df['years'].dtypes
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
/usr/local/lib/python3.10/dist-packages/pandas/core/indexes/base.py in␣
 ↪get_loc(self, key, method, tolerance)
   3801             try:
-> 3802                 return self._engine.get_loc(casted_key)
   3803             except KeyError as err:

/usr/local/lib/python3.10/dist-packages/pandas/_libs/index.pyx in pandas._libs.
 ↪index.IndexEngine.get_loc()

/usr/local/lib/python3.10/dist-packages/pandas/_libs/index.pyx in pandas._libs.
 ↪index.IndexEngine.get_loc()

pandas/_libs/hashtable_class_helper.pxi in pandas._libs.hashtable.
 ↪PyObjectHashTable.get_item()

pandas/_libs/hashtable_class_helper.pxi in pandas._libs.hashtable.
 ↪PyObjectHashTable.get_item()

KeyError: 'years'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
<ipython-input-58-ba2926af7469> in <cell line: 1>()
```

```
----> 1 df['years'].dtypes

/usr/local/lib/python3.10/dist-packages/pandas/core/frame.py in␣
 ↪__getitem__(self, key)
   3805             if self.columns.nlevels > 1:
   3806                 return self._getitem_multilevel(key)
-> 3807             indexer = self.columns.get_loc(key)
   3808             if is_integer(indexer):
   3809                 indexer = [indexer]

/usr/local/lib/python3.10/dist-packages/pandas/core/indexes/base.py in␣
 ↪get_loc(self, key, method, tolerance)
   3802                 return self._engine.get_loc(casted_key)
   3803             except KeyError as err:
-> 3804                 raise KeyError(key) from err
   3805             except TypeError:
   3806                 # If we have a listlike key, _check_indexing_error will␣
 ↪raise

KeyError: 'years'
```

[ ]:
```
df_genre=pd.read_csv("SpotifyFeatures.csv",encoding="unicode_escape")
```

[ ]:
```
df_genre.head()
```

[ ]:
```
  ï»¿genre         artist_name                        track_name  \
0    Movie      Henri Salvador       C'est beau de faire un Show
1    Movie  Martin & les fÃ©es  Perdu d'avance (par Gad Elmaleh)
2    Movie     Joseph Williams    Don't Let Me Be Lonely Tonight
3    Movie      Henri Salvador    Dis-moi Monsieur Gordon Cooper
4    Movie        Fabien Nataf                         Ouverture

                 track_id  popularity  acousticness  danceability  \
0  0BRj06ga9RKCKjfDqeFgWV           0         0.611         0.389
1  0BjC1NfoEOOusryehmNudP           1         0.246         0.590
2  0CoSDzoNIKCRs124s9uTVy           3         0.952         0.663
3  0Gc6TVm52BwZD07Ki6tIvf           0         0.703         0.240
4  0IuslXpMROHdEPvSl1fTQK           4         0.950         0.331

   duration_ms  energy  instrumentalness key  liveness  loudness   mode  \
0        99373   0.910             0.000  C#    0.3460    -1.828  Major
1       137373   0.737             0.000  F#    0.1510    -5.559  Minor
2       170267   0.131             0.000   C    0.1030   -13.879  Minor
3       152427   0.326             0.000  C#    0.0985   -12.178  Major
4        82625   0.225             0.123   F    0.2020   -21.150  Major

   speechiness    tempo time_signature  valence
```

```
0       0.0525  166.969            4/4    0.814
1       0.0868  174.003            4/4    0.816
2       0.0362   99.488            5/4    0.368
3       0.0395  171.758            4/4    0.227
4       0.0456  140.576            4/4    0.390
```

```
[ ]: df_genre.describe()
```

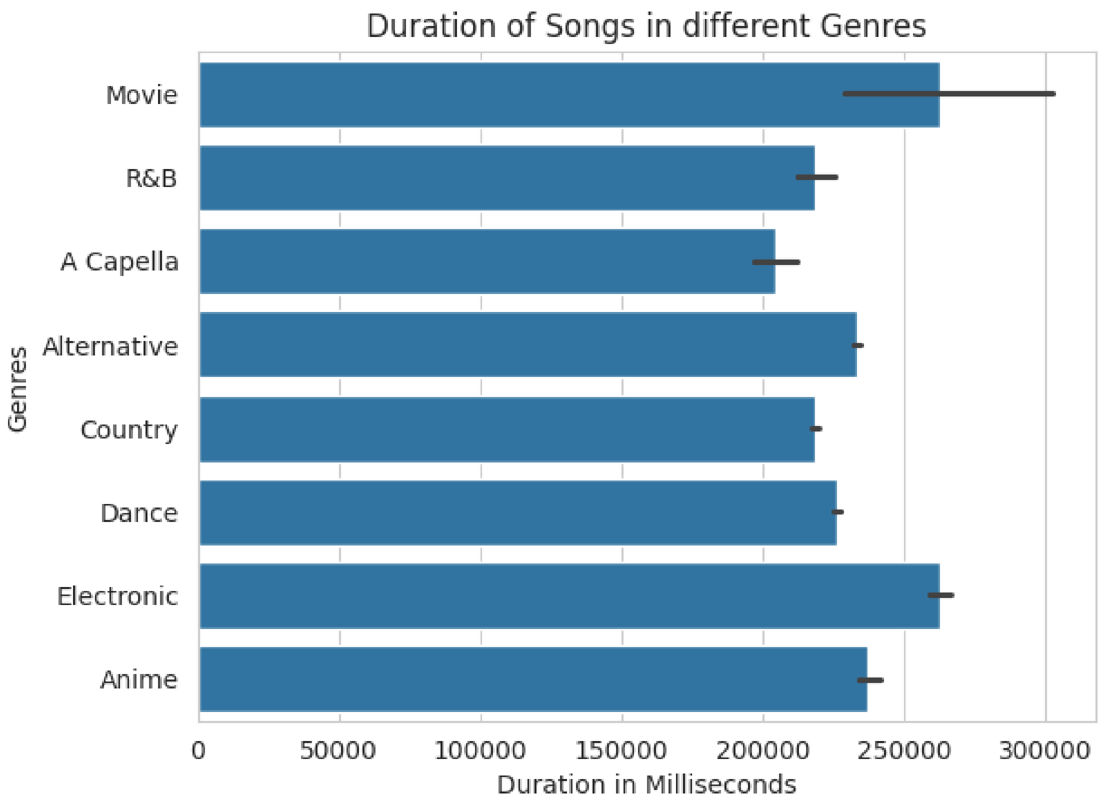
```
[ ]:          popularity  acoustiness  danceability  duration_ms        energy  \
     count  29485.000000  29485.000000  29485.000000  2.948500e+04  29485.000000
     mean      48.079023      0.185312      0.583156  2.344866e+05      0.692810
     std       12.760919      0.248963      0.146904  8.726672e+04      0.204764
     min        0.000000      0.000001      0.061700  1.880000e+04      0.001540
     25%       41.000000      0.008500      0.489000  1.957070e+05      0.554000
     50%       48.000000      0.063600      0.590000  2.233850e+05      0.723000
     75%       56.000000      0.272000      0.686000  2.590000e+05      0.862000
     max      100.000000      0.996000      0.987000  4.830606e+06      0.999000

            instrumentalness      liveness      loudness   speechiness  \
     count      29485.000000  29485.000000  29485.000000  29485.000000
     mean           0.099874      0.196430     -6.728825      0.083502
     std            0.242260      0.158856      3.322457      0.087804
     min            0.000000      0.014900    -37.286000      0.022300
     25%            0.000000      0.096700     -8.208000      0.035200
     50%            0.000030      0.131000     -6.028000      0.049300
     75%            0.010700      0.257000     -4.511000      0.089100
     max            0.994000      0.996000      1.893000      0.961000

                   tempo       valence
     count  29484.000000  29484.000000
     mean     122.796272      0.469736
     std       28.850826      0.232536
     min       32.244000      0.000000
     25%       99.999000      0.289000
     50%      122.005000      0.456000
     75%      141.954000      0.643000
     max      220.169000      0.992000
```

```
[ ]: df_genre.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29485 entries, 0 to 29484
Data columns (total 18 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   ï»¿genre           29485 non-null  object
 1   artist_name       29485 non-null  object
 2   track_name        29485 non-null  object
 3   track_id          29485 non-null  object
 4   popularity        29485 non-null  int64
 5   acousticness      29485 non-null  float64
 6   danceability      29485 non-null  float64
 7   duration_ms       29485 non-null  int64
 8   energy            29485 non-null  float64
 9   instrumentalness  29485 non-null  float64
 10  key               29485 non-null  object
 11  liveness          29485 non-null  float64
 12  loudness          29485 non-null  float64
 13  mode              29485 non-null  object
 14  speechiness       29485 non-null  float64
 15  tempo             29484 non-null  float64
 16  time_signature    29484 non-null  object
 17  valence           29484 non-null  float64
dtypes: float64(9), int64(2), object(7)
memory usage: 4.0+ MB
```

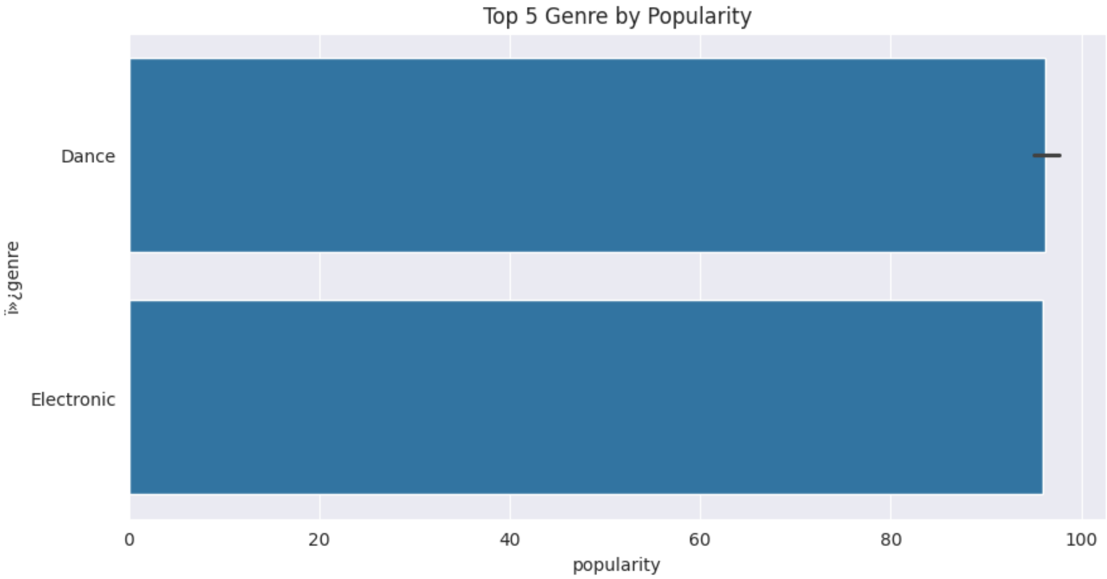
```
[ ]: plt.title("Duration of Songs in different Genres")
     sns.color_palette("rocket",as_cmap=True)
     sns.barplot(y='ï»¿genre',x='duration_ms',data=df_genre)
     plt.xlabel("Duration in Milliseconds")
     plt.ylabel("Genres")
```

```
[ ]: Text(0, 0.5, 'Genres')
```



```
[ ]: sns.set_style(style="darkgrid")
     plt.figure(figsize=(10,5))
     famous=df_genre.sort_values("popularity",ascending=False).head(10)
     sns.barplot(y="ï»¿genre",x="popularity",data=famous).set(title="Top 5 Genre by␣
      ↪Popularity")
```

```
[ ]: [Text(0.5, 1.0, 'Top 5 Genre by Popularity')]
```


Top 5 Genre by Popularity
ï»¿genre
Dance
Electronic
0
20
40
60
80
100
popularity

```
[ ]:
```

```
[ ]:
```