In [ ]:
```
#importing necessary libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [12]:
```
df_tracks=pd.read_csv('E:/SQL/spotify.csv')
df_tracks.head()
```

Out[12]:

	uri	artist_names	track_name	peak_rank	weeks_on_chart	c
0	spotify:track:02MWAaffLxlfxAUY7c5dvx	Glass Animals	Heat Waves	1	65	
1	spotify:track:5PjdY0CKGZdEuoNab3yDmX	The Kid LAROI, Justin Bieber	STAY (with Justin Bieber)	1	37	
2	spotify:track:3IAfUEeaXRX9s9UdKOJrFI	Anitta	Envolver	3	3	
3	spotify:track:1HhNoOuqm1a5MXYEgAFl8o	Imagine Dragons, JID, Arcane, League of Legends	Enemy (with JID) - from the series Arcane Leag...	3	21	
4	spotify:track:4fouWK6XVHhzl78KzQ1UjL	GAYLE	abcdefu	1	19	

In [13]:
```
#to check the totoal number of null values in each column of the dataset
pd.isnull(df_tracks).sum()
```

Out[13]:
```
uri                0
artist_names       0
track_name         0
peak_rank          0
weeks_on_chart     0
danceability       0
energy             0
key                0
loudness           0
mode               0
speechiness        0
acousticness       0
instrumentalness   0
liveness           0
tempo              0
time_signature     0
duration_ms        0
dtype: int64
```

In [14]:
```
df_tracks.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 646 entries, 0 to 645
Data columns (total 17 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   uri               646 non-null    object
 1   artist_names      646 non-null    object
 2   track_name        646 non-null    object
 3   peak_rank         646 non-null    int64
 4   weeks_on_chart    646 non-null    int64
 5   danceability      646 non-null    float64
 6   energy            646 non-null    float64
 7   key               646 non-null    int64
 8   loudness          646 non-null    float64
 9   mode              646 non-null    int64
 10  speechiness       646 non-null    float64
 11  acousticness      646 non-null    float64
 12  instrumentalness  646 non-null    float64
 13  liveness          646 non-null    float64
 14  tempo             646 non-null    float64
 15  time_signature    646 non-null    int64
 16  duration_ms       646 non-null    int64
dtypes: float64(8), int64(6), object(3)
memory usage: 85.9+ KB
```

In [15]:
```
#shows the highest number of weeks retained by a song in the topsongs chart
sort_df=df_tracks.sort_values('weeks_on_chart',ascending=False).head(10)
sort_df
```

Out[15]:

	uri	artist_names	track_name	peak_rank	weeks_on_cha
139	spotify:track:7qiZfU4dY1lWllzX7mPBI3	Ed Sheeran	Shape of You	1	2
198	spotify:track:5uCax9HTNlzGybIStD3vDh	James Arthur	Say You Won't Let Go	7	2
91	spotify:track:0pqnGHJpmpxLKifKRmU6WP	Imagine Dragons	Believer	16	2
150	spotify:track:0tgVpDi06FyKpA1z0VMD4v	Ed Sheeran	Perfect	4	2
397	spotify:track:6gBFPUFcJLzWGx4lenP6h2	Travis Scott	goosebumps	13	2
484	spotify:track:7m9OqQk4RVRkw9JJdeAw96	XXXTENTACION	Jocelyn Flores	7	2
102	spotify:track:0u2P5u6lvoDfwTYjAADbn4	Billie Eilish, Khalid	lovely (with Khalid)	24	2
177	spotify:track:285pBltuF7vW8TeWk8hdRR	Juice WRLD	Lucid Dreams	3	1
146	spotify:track:4u7EnebtmKWzUH433cf5Qv	Queen	Bohemian Rhapsody - Remastered 2011	11	1
93	spotify:track:0RiRZpuVRbi7oqRdSMwhQY	Post Malone, Swae Lee	Sunflower - Spider-Man: Into the Spider-Verse	1	1

In [16]:
```
df_tracks.describe().transpose()
```

Out[16]:

	count	mean	std	min	25%	50%
peak_rank	646.0	65.922601	57.004503	1.000000	15.000000	51.500000
weeks_on_chart	646.0	19.498452	37.813937	1.000000	1.000000	5.000000
danceability	646.0	0.674426	0.151527	0.193000	0.569000	0.700000
energy	646.0	0.640836	0.165478	0.021800	0.532500	0.652000
key	646.0	5.086687	3.622314	0.000000	1.000000	5.000000
loudness	646.0	-6.356034	2.627462	-31.160000	-7.716250	-5.930500
mode	646.0	0.577399	0.494356	0.000000	0.000000	1.000000
speechiness	646.0	0.109797	0.102167	0.023200	0.041325	0.063450
acousticness	646.0	0.255699	0.262595	0.000017	0.040400	0.155500
instrumentalness	646.0	0.013686	0.077836	0.000000	0.000000	0.000002
liveness	646.0	0.176478	0.133792	0.026400	0.093100	0.120000
tempo	646.0	121.091571	28.267951	66.165000	98.428500	119.966000
time_signature	646.0	3.927245	0.351256	1.000000	4.000000	4.000000
duration_ms	646.0	203629.859133	54966.666438	36935.000000	169901.500000	196388.500000

In [17]:
```
#finding the track name with the respective artist name using iloc
df_tracks[["track_name","artist_names"]].iloc[30]
```

Out[17]:
```
track_name      Another Love
artist_names       Tom Odell
Name: 30, dtype: object
```

In [ ]:
```

```

In [ ]:
```

```

In [ ]:
```

```

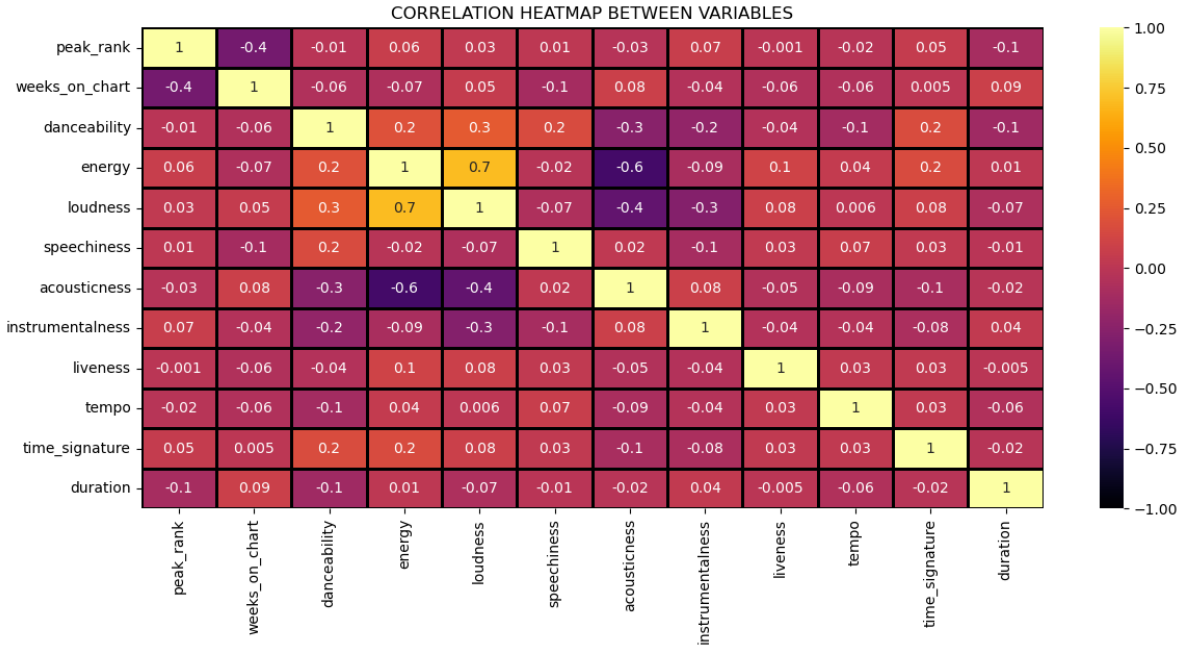
In [18]:
```
#converting track duration in milliseconds to minutes
df_tracks["duration"]=df_tracks["duration_ms"].apply(lambda x:(x*1.66667e-5))
df_tracks.drop("duration_ms",inplace=True,axis=1)
```

In [19]:
```
df_tracks.duration.head()
```

Out[19]:
```
0    3.980091
1    2.363438
2    3.230106
3    2.889689
4    2.810039
Name: duration, dtype: float64
```

In [48]:
```
corr_df=df_tracks.drop({"key","mode"},axis=1).corr(method="pearson")
plt.figure(figsize=(14,6))
heatmap=sns.heatmap(corr_df,annot=True,fmt=".1g",vmin=-1,vmax=1,center=0,cmap="inf
heatmap.set_title("CORRELATION HEATMAP BETWEEN VARIABLES")
heatmap.set_xticklabels(heatmap.get_xticklabels(),rotation=90)
```

Out[48]:
```
[Text(0.5, 0, 'peak_rank'),
 Text(1.5, 0, 'weeks_on_chart'),
 Text(2.5, 0, 'danceability'),
 Text(3.5, 0, 'energy'),
 Text(4.5, 0, 'loudness'),
 Text(5.5, 0, 'speechiness'),
 Text(6.5, 0, 'acousticness'),
 Text(7.5, 0, 'instrumentalness'),
 Text(8.5, 0, 'liveness'),
 Text(9.5, 0, 'tempo'),
 Text(10.5, 0, 'time_signature'),
 Text(11.5, 0, 'duration')]
```

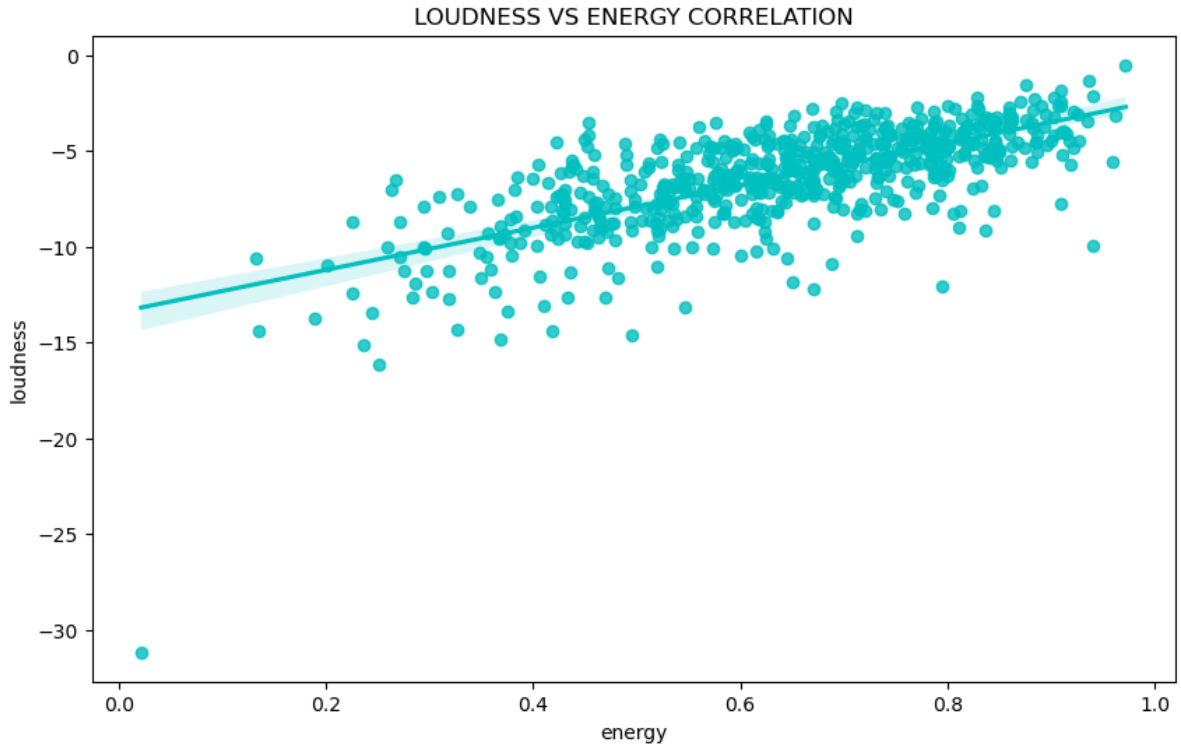


In [39]:
```
#shows total number of dataset present in the excel
sample_df=df_tracks.sample(int(len(df_tracks)))
print(len(sample_df))
```

646

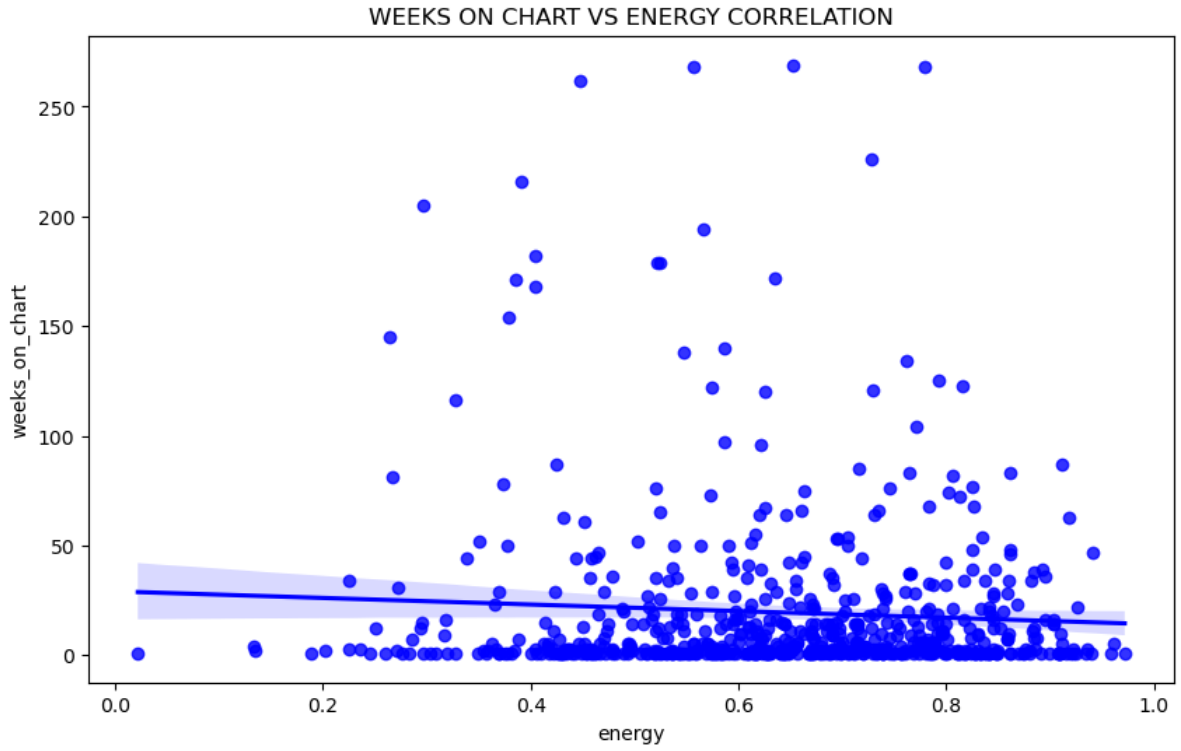
In [45]:
```
plt.figure(figsize=(10,6))
sns.regplot(data= sample_df,y="loudness",x="energy",color="c").set(title="LOUDNESS
```

Out[45]:
```
[Text(0.5, 1.0, 'LOUDNESS VS ENERGY CORRELATION')]
```



In [46]:
```
plt.figure(figsize=(10,6))
sns.regplot(data= sample_df,y="weeks_on_chart",x="energy",color="b").set(title="WE
```

Out[46]:
```
[Text(0.5, 1.0, 'WEEKS ON CHART VS ENERGY CORRELATION')]
```



In [ ]:

In [ ]: