Data and ETL Process

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df_tracks= pd.read_csv('Spotify.csv')
df_tracks
```

	genre	artist_name	track_name	track_id	popularity	acoust
0	Movie	Henri Salvador	C'est beau de faire un Show	0BRjO6ga9RKCKjfDqeFgWV	0	
1	Movie	Martin & les fées	Perdu d'avance (par Gad Elmaleh)	0BjC1NfoEOOusryehmNudP	1	
2	Movie	Joseph Williams	Don't Let Me Be Lonely Tonight	0CoSDzoNIKCRs124s9uTVy	3	
3	Movie	Henri Salvador	Dis-moi Monsieur Gordon Cooper	0Gc6TVm52BwZD07Ki6tIvf	0	
4	Movie	Fabien Nataf	Ouverture	0IuslXpMROHdEPvSl1fTQK	4	
...	...	...	...	...	...	
232711	Soul	Slave	Son Of Slide	2XGLdVl7lGeq8ksM6Al7jT	39	
232712	Soul	Jr Thomas & The Volcanos	Burning Fire	1qWZdkBl4UVPj9lK6HuuFM	38	
232713	Soul	Muddy Waters	(I'm Your) Hoochie Coochie Man	2ziWXUmQLrXTiYjCg2fZ2t	47	
232714	Soul	R.LUM.R	With My Words	6EFsue2YbIG4Qkq8Zr9Rir	44	
232715	Soul	Mint Condition	You Don't Have To Hurt No More	34XO9RwPMKjbvRry54QzWn	35	

232716 rows × 17 columns

```
#find Null values
pd.isnull(df_tracks).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
Emotions            0
dtype: int64
```

```
df_tracks['duration_ms'] = (df_tracks['duration_ms'] / 1000)
df_tracks.rename({'duration_ms': 'duration_sec'}, axis=1, inplace=True)
df_tracks.head()
```

	genre	artist_name	track_name	track_id	popularity	acousticness	danceability	du
0	Movie	Henri Salvador	C'est beau de faire un Show	0BRjO6ga9RKCKjfDqeFgWV	0	0.611	0.389	
1	Movie	Martin & les fées	Perdu d'avance (par Gad Elmaleh)	0BjC1NfoEOOusryehmNudP	1	0.246	0.590	
2	Movie	Joseph Williams	Don't Let Me Be Lonely Tonight	0CoSDzoNIKCRs124s9uTVy	3	0.952	0.663	
3	Movie	Henri	Dis-moi Monsieur	0Gc6TVm52BwZD07Ki6tIvf	0	0.703	0.240	

```
#Remove all the spaces
df_tracks['artist_name'] = df_tracks['artist_name'].str.replace(" ", "")
df_tracks
```

	genre	artist_name	track_name	track_id	popularity	acousticness	da
0	Movie	HenriSalvador	C'est beau de faire un Show	0BRjO6ga9RKCKjfDqeFgWV	0	0.61100	
1	Movie	Martin&lesfées	Perdu d'avance (par Gad Elmaleh)	0BjC1NfoEOOusryehmNudP	1	0.24600	
2	Movie	JosephWilliams	Don't Let Me Be Lonely Tonight	0CoSDzoNIKCRs124s9uTVy	3	0.95200	
3	Movie	HenriSalvador	Dis-moi Monsieur Gordon Cooper	0Gc6TVm52BwZD07Ki6tIvf	0	0.70300	
4	Movie	FabienNataf	Ouverture	0IuslXpMROHdEPvSl1fTQK	4	0.95000	
...	...	...	...	...	...	...	
232711	Soul	Slave	Son Of Slide	2XGLdVl7lGeq8ksM6Al7jT	39	0.00384	
232712	Soul	JrThomas&TheVolcanos	Burning Fire	1qWZdkBl4UVPj9lK6HuuFM	38	0.03290	
232713	Soul	MuddyWaters	(I'm Your) Hoochie Coochie Man	2ziWXUmQLrXTiYjCg2fZ2t	47	0.90100	
232714	Soul	R.LUM.R	With My Words	6EFsue2YbIG4Qkq8Zr9Rir	44	0.26200	
232715	Soul	MintCondition	You Don't Have To Hurt No More	34XO9RwPMKjbvRry54QzWn	35	0.09730	

232716 rows × 17 columns

Data Analysis

1. Every genre will have all unique song two genre will not have one song

```
sorted_df = df_tracks.sort_values('popularity', ascending = False)
df_onetracks=sorted_df.drop_duplicates(subset='track_name',keep='first')
df_onetracks
```

	genre	artist_name	track_name	track_id	popularity	acousticness	danc
9027	Dance	ArianaGrande	7 rings	14msK75pk3pA33pzPVNtBF	100	0.5780	
9026	Dance	ArianaGrande	break up with your girlfriend, i'm bored	4kV4N9D1iKVxx1KLvtTpjS	99	0.0421	
86946	Rap	PostMalone	Wow.	6MWtB6iiXyIwun0YzU6DFP	99	0.1630	
107904	Pop	DaddyYankee	Con Calma	5w9c2J52mkdntKOmRLeM2m	98	0.1100	
107801	Pop	PostMalone	Sunflower - Spider-Man: Into the Spider-Verse	3KkXRkHbMCARz0aVfEt68P	97	0.5560	
...	...	...	...	...	...	...	
195730	Movie	Chorus	Anandabhairavi	1J7VPB980Q6XdsuwIeEgNW	0	0.7100	
195731	Movie	LucieDolene	Quel temps fait-il à Paris ? (From "Les vacanc...	1JeSn51usJJZENvNOjUyjZ	0	0.9920	
195734	Movie	LeopoldStokowski	Tchaikovsky: Aurora's Wedding Ballet Music: Ac...	1P87O9nlFeciqOfc6ahUl8	0	0.9640	
195736	Movie	Chorus	Om Gam Ganapathaye Namah	1VH7vTIlikS9gHEKrWkpQa	0	0.9770	
0	Movie	HenriSalvador	C'est beau de faire un Show	0BRjO6ga9RKCKjfDqeFgWV	0	0.6110	

2. sorting down according to the basis of the genre

```
df = df_onetracks.groupby(['genre', 'artist_name'])
df.first()
```

		track_name	track_id	popularity	acousticness	danceabili
genre	artist_name					

3. find the songs of the artist

```
df= df_onetracks.groupby('artist_name')
df.get_group(input('Enter the artist name'))
```

```
Enter the artist nameArianaGrande
```

	genre	artist_name	track_name	track_id	popularity	acousticness	danceabil
9027	Dance	ArianaGrande	7 rings	14msK75pk3pA33pzPVNtBF	100	0.5780	0.
9026	Dance	ArianaGrande	break up with your girlfriend, i'm bored	4kV4N9D1iKVxx1KLvtTpjS	99	0.0421	0.
107809	Pop	ArianaGrande	thank u, next	2rPE9A1vEgShuZxxzR2tZH	95	0.2800	0.
107806	Pop	ArianaGrande	needy	1TEL6MlSSVLSdhOSddidlJ	92	0.7800	0.
107812	Pop	ArianaGrande	bloodline	2hloaUoRonYssMuqLCBLTX	91	0.0815	0.
...	...	...	...	...	...	...	
17652	Dance	ArianaGrande	You'll Never Know	4Pqlj0WOfPAq4QAvisjgpd	50	0.0554	0.
21148	Dance	ArianaGrande	Side To Side - Slushii Remix	27WOAkJrAYagnYj7Y2tsFw	50	0.0761	0.
20287	Dance	ArianaGrande	Not Just On Christmas	6JJ4W13A4Q5yeK4g6OROmg	49	0.1800	0.
18750	Dance	ArianaGrande	Lovin' It	7EpKfPAURnG9OCVer0S30N	48	0.1080	0.
19403	Dance	ArianaGrande	Better Left Unsaid	5Pnny78GESkBSLnxFmhRYZ	46	0.0442	0.

69 rows × 17 columns

4. Descriptive Statistics

```
df_onetracks.describe().transpose()
```

	count	mean	std	min	25%	50%	75%	max
popularity	148601.0	36.544290	17.905777	0.00000	25.0000	37.000000	49.000	100.000
acousticness	148601.0	0.415961	0.370475	0.00000	0.0490	0.307000	0.810	0.996
danceability	148601.0	0.536211	0.193302	0.05690	0.4060	0.554000	0.681	0.989
duration_sec	148601.0	237.954180	136.163322	15.38700	177.3730	219.853000	270.773	5552.917
energy	148601.0	0.551045	0.280243	0.00002	0.3290	0.588000	0.788	0.999
instrumentalness	148601.0	0.181188	0.330073	0.00000	0.0000	0.000083	0.126	0.999
liveness	148601.0	0.228605	0.215132	0.01050	0.0978	0.131000	0.283	1.000
loudness	148601.0	-10.394912	6.617151	-52.45700	-13.3050	-8.335000	-5.677	3.744
speechiness	148601.0	0.130553	0.209288	0.02220	0.0371	0.049600	0.104	0.967
tempo	148601.0	116.840135	31.391908	30.37900	91.7370	114.672000	138.091	242.903
Emotions	148601.0	0.448472	0.269878	0.00000	0.2150	0.437000	0.667	1.000

5. most of the songs in which Genre

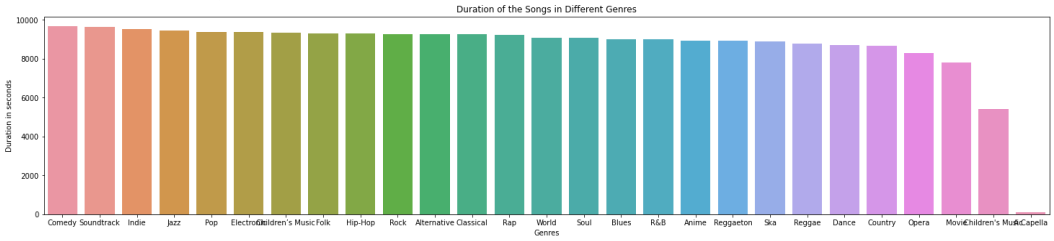
```
df_numbercharteds=df_tracks.groupby('genre').count().sort_values('track_name', ascending=False)
df_numbercharteds=df_numbercharteds.reset_index()
df_numbercharteds
```

	genre	artist_name	track_name	track_id	popularity	acousticness	danceability	duration_sec
0	Comedy	9680	9680	9680	9680	9680	9680	9680
1	Soundtrack	9646	9646	9646	9646	9646	9646	9646
2	Indie	9543	9543	9543	9543	9543	9543	9543
3	Jazz	9441	9441	9441	9441	9441	9441	9441
4	Pop	9385	9385	9385	9385	9385	9385	9385
5	Electronic	9377	9377	9377	9377	9377	9377	9377
6	Children's Music	9353	9353	9353	9353	9353	9353	9353
7	Folk	9299	9299	9299	9299	9299	9299	9299
8	Hip-Hop	9295	9295	9295	9295	9295	9295	9295
9	Rock	9272	9272	9272	9272	9272	9272	9272
10	Alternative	9263	9263	9263	9263	9263	9263	9263
11	Classical	9256	9256	9256	9256	9256	9256	9256
12	Rap	9231	9231	9231	9231	9231	9231	9231
13	World	9095	9095	9095	9095	9095	9095	9095
14	Soul	9089	9089	9089	9089	9089	9089	9089
15	Blues	9023	9023	9023	9023	9023	9023	9023
16	R&B	8992	8992	8992	8992	8992	8992	8992
17	Anime	8936	8936	8936	8936	8936	8936	8936
18	Reggaeton	8927	8927	8927	8927	8927	8927	8927
19	Ska	8874	8874	8874	8874	8874	8874	8874
20	Reggae	8771	8771	8771	8771	8771	8771	8771
21	Dance	8700	8700	8700	8700	8700	8700	8700
22	Country	8664	8664	8664	8664	8664	8664	8664
23	Opera	8276	8276	8276	8276	8276	8276	8276
24	Movie	7806	7806	7806	7806	7806	7806	7806
25	Children's Music	5403	5403	5403	5403	5403	5403	5403

6. Most songs in which genre

```
plt.figure(figsize=(25,5))
plt.title("Duration of the Songs in Different Genres")
sns.color_palette("rocket", as_cmap= True)
sns.barplot(x='genre', y='duration_sec', data=df_numbercharteds)
plt.ylabel("Duration in seconds")
plt.xlabel("Genres")
```

```
Text(0.5, 0, 'Genres')
```



7. to determine who has most duration

```
df_numberchart=df_onetracks.groupby('artist_name').sum().sort_values('duration_sec', ascending=False)
df_numberchart=df_numberchart.reset_index()
df_numberchart
```

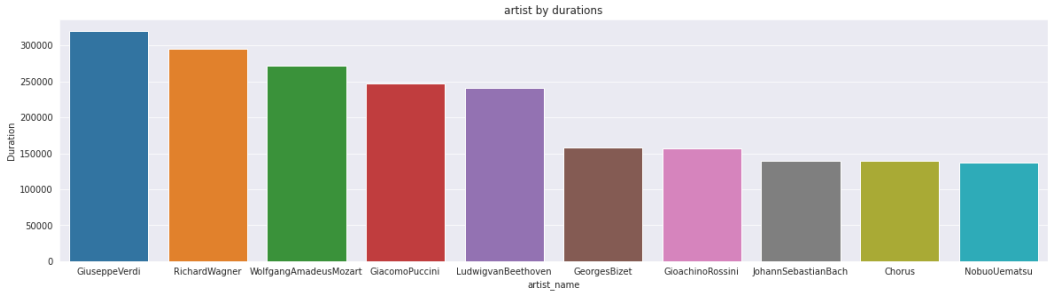
	artist_name	popularity	acousticness	danceability	duration_sec	energy	instrum
0	GiuseppeVerdi	17257	1239.9760	395.6277	320154.235	203.743380	1
1	RichardWagner	9459	727.8960	168.8469	295230.840	118.535219	3
2	WolfgangAmadeusMozart	17751	768.5180	257.5442	271693.972	95.973640	4
3	GiacomoPuccini	14617	1043.5980	282.9923	246990.302	199.157510	1
4	LudwigvanBeethoven	14572	572.7670	193.0588	240604.507	63.416320	4
...	...	...	...	...	...	...	
13693	Emma-JeanThackray	41	0.2970	0.6520	41.667	0.355000	
13694	AdamGussow	33	0.7510	0.5360	35.973	0.229000	
13695	MokaOnly	47	0.4440	0.6800	34.750	0.274000	
13696	Thomas&Friends	50	0.0793	0.9430	30.891	0.505000	
13697	TheLittleSingers	0	0.9240	0.5000	30.840	0.276000	

13698 rows × 12 columns

8. to determine who has most duration

```
sns.set_style(style = "darkgrid")
plt.figure(figsize=(20,5))
famous= df_numberchart.sort_values('duration_sec', ascending = False).head(10)
sns.barplot(x='artist_name', y='duration_sec', data = famous).set(title= "artist by durations")
plt.ylabel("Duration")
```

```
Text(0, 0.5, 'Duration')
```



9. How many songs each artists have

```
df_numbercharted=df_onetracks.groupby('artist_name').count().sort_values('track_name', ascending=False)
df_numbercharted=df_numbercharted.reset_index()
df_numbercharted
```

	artist_name	genre	track_name	track_id	popularity	acousticness	danceability	dura
0	GiuseppeVerdi	1298	1298	1298	1298	1298	1298	

10. The Most Popular Artist

```
df_numberchartedss=df_onetracks.groupby('artist_name').mean().sort_values('popularity', ascending=False)
df_numberchartedss=df_numberchartedss.reset_index()
df_numberchartedss
```

	artist_name	popularity	acousticness	danceability	duration_sec	energy	instrumentalnes
0	PedroCapó	87.0	0.26100	0.7815	209.1330	0.7035	0.00000
1	MarioBautista	85.0	0.02750	0.8040	218.4700	0.7050	0.00013
2	MauyRicky	83.0	0.08245	0.7220	230.0265	0.7115	0.00000
3	KrisKrossAmsterdam	82.0	0.15100	0.7340	163.6360	0.7650	0.00000
4	SofiaReyes	82.0	0.16500	0.7920	201.5260	0.8950	0.00000
...	...	...	...	...	...	...	
13693	TölzerKnabenchor	0.0	0.71700	0.7270	200.2670	0.6770	0.00000
13694	JackGrunsky	0.0	0.39920	0.6698	228.6760	0.4818	0.00000
13695	NurseryRhymesClub	0.0	0.38600	0.4580	147.8790	0.2680	0.86800
13696	TheSippyCups	0.0	0.87500	0.6360	120.2000	0.8480	0.00000
13697	KarlLeury	0.0	0.16300	0.8330	232.2670	0.4950	0.00000

13698 rows × 12 columns

```
artist = df_onetracks['artist_name'].unique()
len(artist)
```

```
13698
```

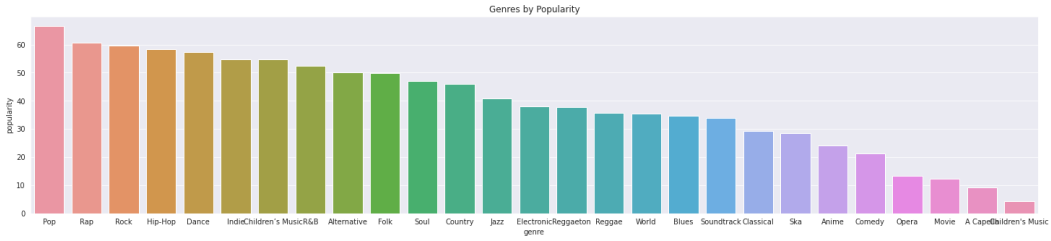
11. Which genre has more popularity

```
df_number=df_tracks.groupby('genre').mean().sort_values('popularity', ascending=False)
df_number=df_number.reset_index()
df_number
```

	genre	popularity	acousticness	danceability	duration_sec	energy	instrumentalness	livene
0	Pop	66.591263	0.224823	0.640238	220.855250	0.642219	0.016601	0.1799
1	Rap	60.533745	0.168078	0.697253	219.849773	0.650532	0.009318	0.1989
2	Rock	59.619392	0.196429	0.538292	237.003227	0.683670	0.053288	0.1869
3	Hip-Hop	58.423131	0.176172	0.718808	219.981913	0.643275	0.011200	0.2011
4	Dance	57.274943	0.152884	0.638193	226.261345	0.698085	0.035453	0.1877
5	Indie	54.701561	0.331214	0.566821	224.150374	0.581002	0.085317	0.1689
	Children's							

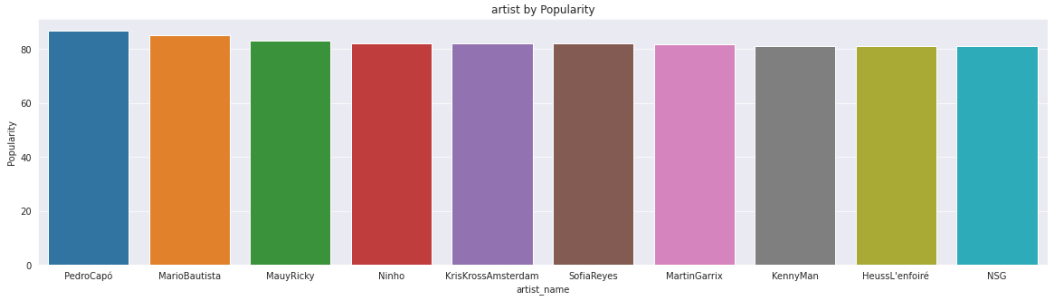
```
sns.set_style(style = "darkgrid")
plt.figure(figsize=(25,5))
famous= df_number.sort_values("popularity", ascending = False)
sns.barplot(x='genre', y='popularity', data = famous).set(title= "Genres by Popularity")
```

```
[Text(0.5, 1.0, 'Genres by Popularity')]
```

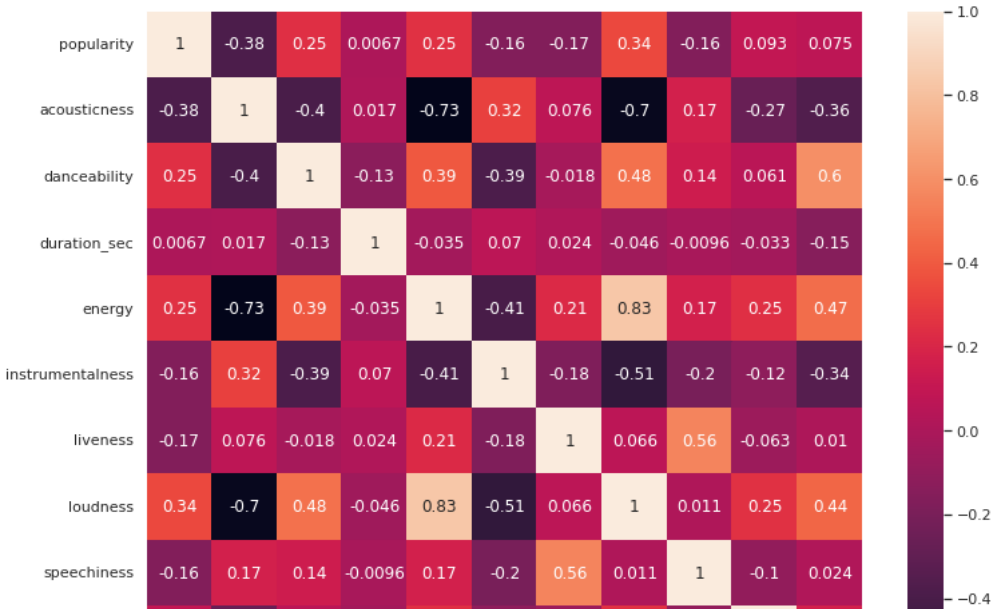



```
sns.set_style(style = "darkgrid")
plt.figure(figsize=(20,5))
famous= df_numberchartedss.sort_values('popularity', ascending = False).head(10)
sns.barplot(x='artist_name', y='popularity', data = famous).set(title= "artist by Popularity")
plt.ylabel("Popularity")
```

```
Text(0, 0.5, 'Popularity')
```

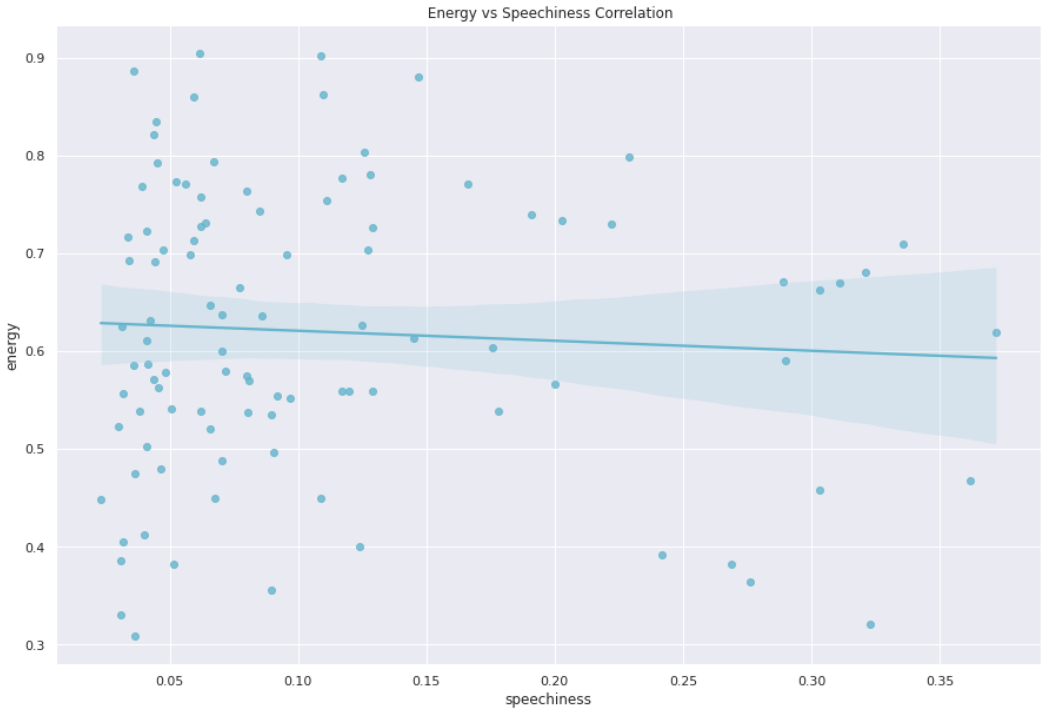



```
import seaborn as sn
sn.set(rc = {'figure.figsize':(12,10)})
sn.heatmap(df_onetracks.corr(), annot=True)
plt.show()
```




```
import matplotlib.pyplot as plt
import seaborn as sns
df_onetrackss = df_onetracks.head(100)
plt.figure(figsize=(15,10))
sns.regplot(data=df_onetrackss, y='energy', x='speechiness', color='c').set(title=' Energy vs Speechiness Correlation')
```

```
[Text(0.5, 1.0, ' Energy vs Speechiness Correlation')]
```




```
plt.figure(figsize=(18,6))
sns.regplot(data = df_onetracks, y= "loudness", x = "energy", color = "C").set(title="Loudness vs Energy Correlation")
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/regression.py:359: MatplotlibDeprecationWarning: Support
  color = mpl.colors.rgb2hex(mpl.colors.colorConverter.to_rgb(color))
[Text(0.5, 1.0, 'Loudness vs Energy Correlation')]
```