

CORRELATION MATRIX

```
In [19]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import PolynomialFeatures
```

```
In [2]: df_spotify=pd.read_csv(r"C:\Users\arumu\Downloads\spotify dataset.csv")
df_spotify
```

```
Out[2]:
```

	track_id	track_name	track_artist	track_popularity	track_album_id	track_album_name	tra
0	6f807x0ima9a1j3VPbc7VN	I Don't Care (with Justin Bieber) - Loud Luxur...	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSI2Cx	I Don't Care (with Justin Bieber) [Loud Luxury...	
1	0r7CVbZTWZgbTCYdfa2P31	Memories - Dillon Francis Remix	Maroon 5	67	63rPSO264uRjW1X5E6cWv6	Memories (Dillon Francis Remix)	
2	1z1Hg7Vb0AhHDiEmnDE79I	All the Time - Don Diablo Remix	Zara Larsson	70	1HoSmj2eLcsrR0vE9gThr4	All the Time (Don Diablo Remix)	
3	75FpbthrwQmzHIBJLuGdC7	Call You Mine - Keanu Silva Remix	The Chainsmokers	60	1nqYsOef1yKKuGOVchbsk6	Call You Mine - The Remixes	
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved - Future Humans Remix	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2zsQ	Someone You Loved (Future Humans Remix)	
...
32828	7bxnKAamR3snQ1VGLuVfC1	City Of Lights - Official Radio Edit	Lush & Simon	42	2azRoBBWEEYhqV6sb7JrT	City Of Lights (Vocal Mix)	
32829	5Aevni09Em4575077nkWHz	Closer - Sultan & Ned Shepard Remix	Tegan and Sara	20	6kD6KLxj7s8eCE3ABvAyf5	Closer Remixed	
32830	7ImMqPP3Q1yfUHvsdn7wEo	Sweet Surrender - Radio Edit	Starkillers	14	0ltWNSY9JgxolZO4VzuCa6	Sweet Surrender (Radio Edit)	
32831	2m69mhnfQ1Oq6IGtXuYhgX	Only For You - Maor Levi Remix	Mat Zo	15	1fGrOkHnHJcStl14zNx8Jy	Only For You (Remixes)	
32832	29zWqhca3zt5NsckZqDf6c	Typhoon - Original Mix	Julian Calor	27	0X3mUOm6MhxR7PzxG95rAo	Typhoon/Storm	

32833 rows × 23 columns

```
In [14]: # Select numeric features
numeric_features = df_spotify.select_dtypes(include=['float64', 'int64']).columns
X = df_spotify[numeric_features]
```

```
In [15]: poly = PolynomialFeatures(degree=2, include_bias=False)
X_poly = poly.fit_transform(X)
```

```
In [16]: # polynomial features
poly_feature_names = poly.get_feature_names_out(numeric_features)
X_poly_df = pd.DataFrame(X_poly, columns=poly_feature_names)
```

```
In [17]: correlation_matrix_poly = X_poly_df.corr()
```

```
In [18]: plt.figure(figsize=(14, 10))
sns.heatmap(correlation_matrix_poly, annot=False, cmap='coolwarm', fmt='.2f', linewidths=0.5)
plt.title('Correlation Matrix of Polynomial Features (Degree 2)', fontsize=16)
plt.show()
```

Correlation Matrix of Polynomial Features (Degree 2)

