In [1]:

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

In [2]:

```
# loading the dataset
df=pd.read_csv('amazon.csv')
```

In [3]:

```
df
```

Out[3]:

	product_id	product_name	category	discounted_price	actual_price	disc
0	B07JW9H4J1	Wayona Nylon Braided USB to Lightning Fast Cha...	Computers&Accessories|Accessories&Peripherals|...	₹399	₹1,099	
1	B098NS6PVG	Ambrane Unbreakable 60W / 3A Fast Charging 1.5...	Computers&Accessories|Accessories&Peripherals|...	₹199	₹349	
2	B096MSW6CT	Sounce Fast Phone Charging Cable & Data Sync U...	Computers&Accessories|Accessories&Peripherals|...	₹199	₹1,899	
3	B08HDJ86NZ	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	Computers&Accessories|Accessories&Peripherals|...	₹329	₹699	
4	B08CF3B7N1	Portronics Konnect L 1.2M Fast Charging 3A 8 P...	Computers&Accessories|Accessories&Peripherals|...	₹154	₹399	
...	...	...	...	...	...	
1460	B08L7J3T31	Noir Aqua - 5pcs PP Spun Filter + 1 Spanner | ...	Home&Kitchen|Kitchen&HomeAppliances|WaterPurif...	₹379	₹919	
1461	B01M6453MB	Prestige Delight PRWO Electric Rice Cooker (1 ...	Home&Kitchen|Kitchen&HomeAppliances|SmallKitch...	₹2,280	₹3,045	
1462	B009P2LIL4	Bajaj Majesty RX10 2000 Watts Heat Convector R...	Home&Kitchen|Heating,Cooling&AirQuality|RoomHe...	₹2,219	₹3,080	
1463	B00J5DYCCA	Havells Ventil Air DSP 230mm Exhaust Fan (Pist...	Home&Kitchen|Heating,Cooling&AirQuality|Fans|E...	₹1,399	₹1,890	
1464	[illegible]	Borosil Jumbo 1000-Watt	[illegible]	[illegible]	[illegible]	

	product_id	product_name	category	discounted_price	actual_price	disc
1464	B01486F4G6	Grill Sandwich Maker (...	Home&Kitchen|Kitchen&HomeAppliances|SmallKitch...	₹2,863	₹3,690	

1465 rows × 16 columns

In [4]:

```
df.shape
```

Out[4]:

```
(1465, 16)
```

In [24]:

```
df.head()
```

Out[24]:

	product_id	product_name	category	discounted_price	actual_price	discount_
0	B07JW9H4J1	Wayona Nylon Braided USB to Lightning Fast Cha...	Computers&Accessories|Accessories&Peripherals|...	₹399	₹1,099	
1	B098NS6PVG	Ambrane Unbreakable 60W / 3A Fast Charging 1.5...	Computers&Accessories|Accessories&Peripherals|...	₹199	₹349	
2	B096MSW6CT	Sounce Fast Phone Charging Cable & Data Sync U...	Computers&Accessories|Accessories&Peripherals|...	₹199	₹1,899	
3	B08HDJ86NZ	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	Computers&Accessories|Accessories&Peripherals|...	₹329	₹699	
4	B08CF3B7N1	Portronics Konnect L 1.2M Fast Charging 3A 8 P...	Computers&Accessories|Accessories&Peripherals|...	₹154	₹399	

In [7]:

```
df.dtypes
```

Out[7]:

```
product_id              object
product_name            object
category                object
discounted_price        object
actual_price            object
discount_percentage     object
rating                  object
rating_count            object
about_product           object
user_id                 object
user_name               object
review_id               object
review_title            object
review_content          object
img_link                object
product_link            object
dtype: object
```

In [8]:

```
df.describe(include='all')
```

Out[8]:

	product_id	product_name	category	discounted_price	actual_price	disco
count	1465	1465	1465	1465	1465	
unique	1351	1337	211	550	449	
top	B07JW9H4J1	Fire-Boltt Ninja Call Pro Plus 1.83" Smart Wat...	Computers&Accessories|Accessories&Peripherals|...	₹199	₹999	
freq	3	5	233	53	120	

In [10]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1465 entries, 0 to 1464
Data columns (total 16 columns):
 #   Column               Non-Null Count  Dtype 
---  ------               --------------  ----- 
 0   product_id           1465 non-null   object
 1   product_name         1465 non-null   object
 2   category             1465 non-null   object
 3   discounted_price     1465 non-null   object
 4   actual_price         1465 non-null   object
 5   discount_percentage  1465 non-null   object
 6   rating               1465 non-null   object
 7   rating_count         1463 non-null   object
 8   about_product        1465 non-null   object
 9   user_id              1465 non-null   object
 10  user_name            1465 non-null   object
 11  review_id            1465 non-null   object
 12  review_title         1465 non-null   object
 13  review_content       1465 non-null   object
 14  img_link             1465 non-null   object
 15  product_link         1465 non-null   object
dtypes: object(16)
memory usage: 183.3+ KB
```

In [11]:

```
df.columns
```

Out[11]:

```
Index(['product_id', 'product_name', 'category', 'discounted_price',
       'actual_price', 'discount_percentage', 'rating', 'rating_count',
       'about_product', 'user_id', 'user_name', 'review_id', 'review_title',
       'review_content', 'img_link', 'product_link'],
      dtype='object')
```

In [38]:

```
df['discounted_price'] = df['discounted_price'].replace('[₹,]', '', regex=True)
```

In [41]:

```
df['discounted_price']=df['discounted_price'].astype(float)
```

In [47]:

```
df['discounted_price'].dtype
```

Out[47]:

```
dtype('float64')
```

In [48]:

```
df['actual_price']=df['actual_price'].str.replace('₹','').str.replace(',','')
```

In [56]:

```
df['actual_price']=pd.to_numeric(df['actual_price'],errors='coerce')
df['actual_price'].dtype
```

Out[56]:

```
dtype('float64')
```

In [57]:

```
df['rating']=pd.to_numeric(df['rating'],errors='coerce')
```

In [58]:

```
df['rating'].dtype
```

Out[58]:

```
dtype('float64')
```

In [59]:

```
df['rating_count']=df['rating_count'].str.replace(',','')
```

In [61]:

```
df['rating_count']=pd.to_numeric(df['rating_count'],errors='coerce')
```

In [62]:

```
df['rating_count']
```

Out[62]:

```
0       24269.0
1       43994.0
2        7928.0
3       94363.0
4       16905.0
         ...
1460     1090.0
1461     4118.0
1462      468.0
1463     8031.0
1464     6987.0
Name: rating_count, Length: 1465, dtype: float64
```

In [64]:

```
df['discount_percentage']=df['discount_percentage'].str.replace('%','')
```

In [65]:

```
df['discount_percentage']=df['discount_percentage'].astype(int)
```

In [66]:

```
df.dtypes
```

Out[66]:

```
product id              object
```

```
product_name            object
category                object
discounted_price       float64
actual_price           float64
discount_percentage      int32
rating                 float64
rating_count           float64
about_product           object
user_id                 object
user_name               object
review_id               object
review_title            object
review_content          object
img_link                object
product_link            object
dtype: object
```

In [73]:

```
categorical=[feature for feature in df.columns if df[feature].dtype == 'O']
```

In [70]:

```
numerical=[i for i in df.columns if df[i].dtype!='O']
```

In [71]:

```
numerical
```

Out[71]:

```
['discounted_price',
 'actual_price',
 'discount_percentage',
 'rating',
 'rating_count']
```

In [74]:

```
categorical
```

Out[74]:

```
['product_id',
 'product_name',
 'category',
 'about_product',
 'user_id',
 'user_name',
 'review_id',
 'review_title',
 'review_content',
 'img_link',
 'product_link']
```

"1.What is the average rating for each product category?

In [97]:

```
df['primary_category']=df['category'].apply(lambda x:x.split('|')[0])
```

In [98]:

```
average=df.groupby('primary_category')['rating'].mean().sort_values(ascending=False)
```

In [99]:

```
average
```

Out[99]:

```
primary_category
OfficeProducts            4.309677
Toys&Games                4.300000
HomeImprovement           4.250000
Computers&Accessories     4.154967
Electronics               4.081749
Home&Kitchen              4.040716
Health&PersonalCare       4.000000
MusicalInstruments        3.900000
Car&Motorbike             3.800000
Name: rating, dtype: float64
```

In [ ]:

```
# insights>> Officeproducts has high ratinng
```

2.What are the top rating_count products by category?

In [122]:

```
top_rating=df.groupby('primary_category')['rating_count'].count().sort_values(ascending=
False)
```

In [127]:

```
top_rating
```

Out[127]:

```
primary_category
Electronics               526
Computers&Accessories     451
Home&Kitchen              448
OfficeProducts             31
HomeImprovement             2
MusicalInstruments          2
Car&Motorbike               1
Health&PersonalCare         1
Toys&Games                  1
Name: rating_count, dtype: int64
```

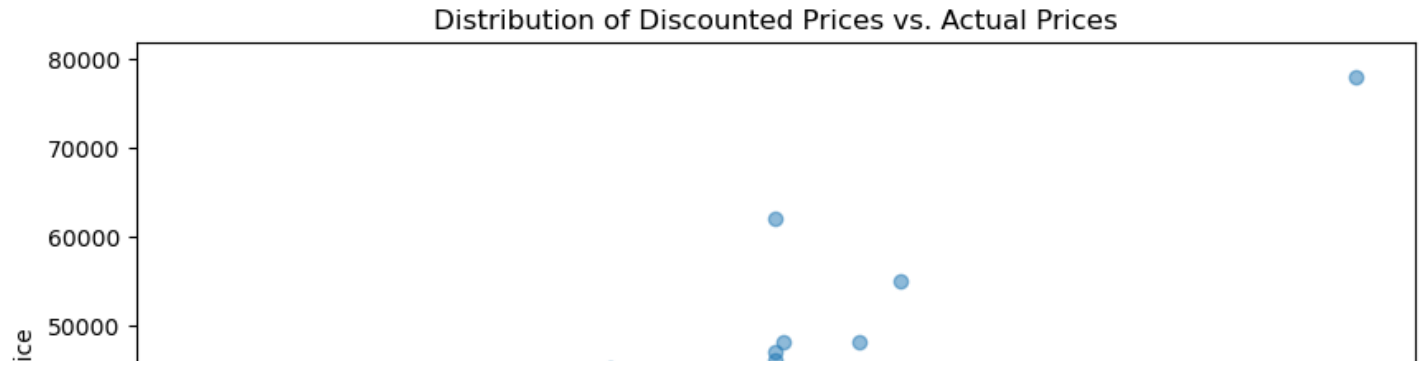
In [ ]:

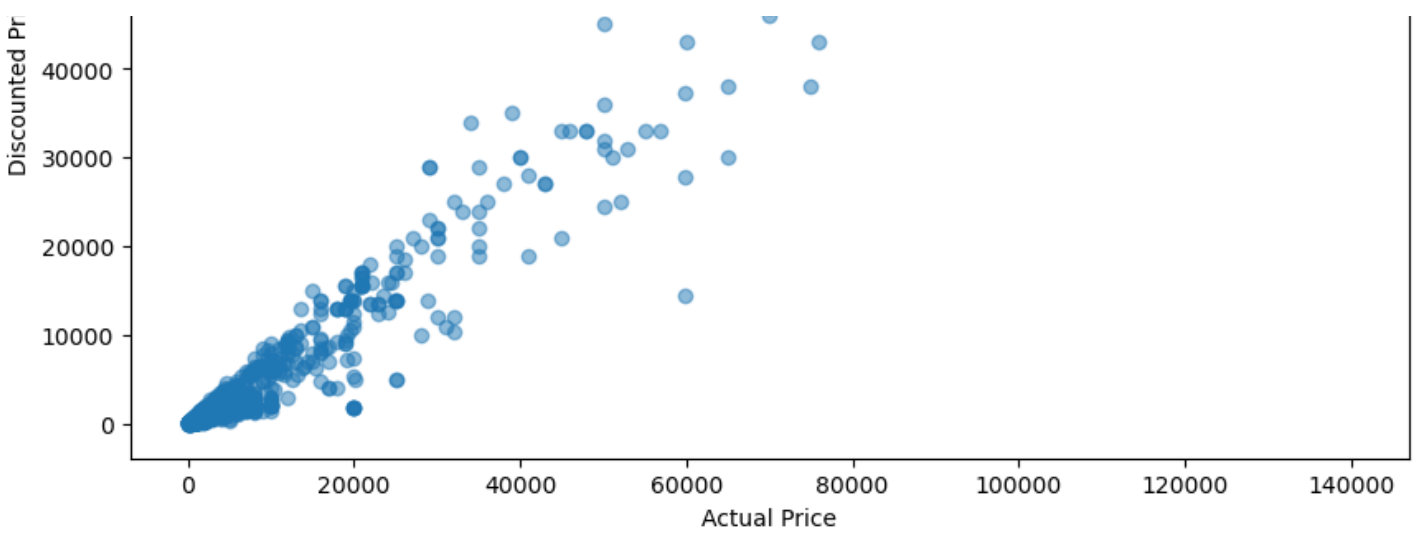
```
# insights>> top rating_count of the product is Electronics
```

3What is the distribution of discounted prices vs. actual prices?

In [141]:

```
# Create a scatter plot
plt.figure(figsize=(10, 6))
plt.scatter(x=df['actual_price'],y=df['discounted_price'], alpha=0.5)
plt.title('Distribution of Discounted Prices vs. Actual Prices')
plt.xlabel('Actual Price')
plt.ylabel('Discounted Price')
plt.show()
```





4. How does the average discount percentage vary across categories?

In [151]:

```
average_percentage=df.groupby('primary_category')['discount_percentage'].mean().sort_values(ascending=False)
```

In [152]:

```
average_percentage
```

Out[152]:

```
primary_category
HomeImprovement           57.500000
Computers&Accessories     54.024283
Health&PersonalCare       53.000000
Electronics               50.828897
MusicalInstruments        46.000000
Car&Motorbike             42.000000
Home&Kitchen              40.120536
OfficeProducts            12.354839
Toys&Games                 0.000000
Name: discount_percentage, dtype: float64
```

In [ ]:

```
# insights >> Home Improvement has high discount percentage
```

5. What are the most popular product names?

In [173]:

```
df['product_name'].value_counts().sort_values(ascending=False)[:10].reset_index()
```

Out[173]:

	index	product_name
0	Fire-Boltt Ninja Call Pro Plus 1.83" Smart Wat...	5
1	Fire-Boltt Phoenix Smart Watch with Bluetooth ...	4
2	Ambrane 2 in 1 Type-C & Micro USB Cable with 6...	3
3	Pinnaclz Original Combo of 2 Micro USB Fast Ch...	3
4	Portronics Konnect L POR-1081 Fast Charging 3A...	3
5	boAt Micro USB 55 Tangle-free, Sturdy Micro US...	3
6	MI Usb Type-C Cable Smartphone (Black)	3
7	pTron Solero TB301 3A Type-C Data and Fast Cha...	3

index	product_name
8 Portronics Konnect L 1.2M Fast Charging 3A 8 P...	3
9 boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	3

In [ ]:

```
# >> Top 10 popular product names
```

6. What are the most popular product keywords?

In [175]:

```
df['product_id'].value_counts().sort_values(ascending=False)[:10].reset_index()
```

Out[175]:

	index	product_id
0	B07JW9H4J1	3
1	B09CMP1SC8	3
2	B085DTN6R2	3
3	B08CF3D7QR	3
4	B096MSW6CT	3
5	B08WRWPM22	3
6	B08Y1TFSP6	3
7	B01GGKYKQM	3
8	B08CF3B7N1	3
9	B098NS6PVG	3

In [ ]:

```
# top 10 popular product keywords
```

7. What are the most popular product reviews?

In [203]:

```
product_review=df.groupby('primary_category')['review_id'].count().sort_values(ascending
=False)
```

In [199]:

```
product_review
```

Out[199]:

```
primary_category
Electronics               526
Computers&Accessories     453
Home&Kitchen              448
OfficeProducts             31
HomeImprovement             2
MusicalInstruments          2
Car&Motorbike               1
Health&PersonalCare         1
Toys&Games                  1
Name: review_id, dtype: int64
```

8. What is the correlation between discounted_price and rating?

In [207]:

```
correlation=df[['discounted_price','rating']].corr()
```

In [211]:

```
sns.heatmap(correlation,annot=True)
plt.title('correlation between discounted_price and rating')
plt.show()
```

9. What are the Top 5 categories based on the highest ratings?

In [224]:

```
h_rating=df.groupby('primary_category')['rating'].mean().sort_values(ascending=False)[:5
].reset_index()
```

In [225]:

```
h_rating
```

Out[225]:

	primary_category	rating
0	OfficeProducts	4.309677
1	Toys&Games	4.300000
2	HomeImprovement	4.250000
3	Computers&Accessories	4.154967
4	Electronics	4.081749

In [ ]:

```
# insights>> Top 5 highest rating categories
```

10. Identify any potential areas for improvement or optimization based on the data analysis

In [226]:

```
df
```

Out[226]:

	product_id	product_name	category	discounted_price	actual_price	disc
0	B07JW9H4J1	Wayona Nylon Braided USB to Lightning Fast Cha...	Computers&Accessories|Accessories&Peripherals|...	399.0	1099.0	
1	B098NS6PVG	Ambrane Unbreakable 60W / 3A Fast Charging 1.5...	Computers&Accessories|Accessories&Peripherals|...	199.0	349.0	
2	B096MSW6CT	Sounce Fast Phone Charging Cable & Data Sync U...	Computers&Accessories|Accessories&Peripherals|...	199.0	1899.0	
3	B08HDJ86NZ	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	Computers&Accessories|Accessories&Peripherals|...	329.0	699.0	
4	B08CF3B7N1	Portronics Konnect L 1.2M Fast Charging 3A 8 P...	Computers&Accessories|Accessories&Peripherals|...	154.0	399.0	
...	...	...	...	...	...	
1460	B08L7J3T31	Noir Aqua - 5pcs PP Spun Filter + 1 Spanner | ...	Home&Kitchen|Kitchen&HomeAppliances|WaterPurif...	379.0	919.0	
1461	B01M6453MB	Prestige Delight PRWO Electric Rice Cooker (1 ...	Home&Kitchen|Kitchen&HomeAppliances|SmallKitch...	2280.0	3045.0	
1462	B009P2LIL4	Bajaj Majesty RX10 2000 Watts Heat Convector R...	Home&Kitchen|Heating,Cooling&AirQuality|RoomHe...	2219.0	3080.0	
1463	B00J5DYCCA	Havells Ventil Air DSP 230mm Exhaust Fan (Pist...	Home&Kitchen|Heating,Cooling&AirQuality|Fans|E...	1399.0	1890.0	
1464	B01486F4G6	Borosil Jumbo 1000-Watt Grill Sandwich Maker (...	Home&Kitchen|Kitchen&HomeAppliances|SmallKitch...	2863.0	3690.0	

1465 rows × 17 columns

In [243]:

```
df.isna().sum()
```

Out[243]:

```
product_id            0
product_name          0
category              0
discounted_price      0
actual_price          0
discount_percentage   0
```

```
discount_percentage    0
rating                 0
rating_count           0
about_product          0
user_id                0
user_name              0
review_id              0
review_title           0
review_content         0
img_link               0
product_link           0
primary_category       0
dtype: int64
```

In [ ]:

```
# >> there are 1 null value in rating and 2 null values in rating count.So we can fill wi
th 0
```

In [240]:

```
df['rating'].fillna('0',inplace=True)
```

In [242]:

```
df['rating_count'].fillna('0',inplace=True)
```

In [251]:

```
df.drop_duplicates()
```

Out[251]:

	product_id	product_name	category	discounted_price	actual_price	disc
0	B07JW9H4J1	Wayona Nylon Braided USB to Lightning Fast Cha...	Computers&Accessories|Accessories&Peripherals|...	399.0	1099.0	
1	B098NS6PVG	Ambrane Unbreakable 60W / 3A Fast Charging 1.5...	Computers&Accessories|Accessories&Peripherals|...	199.0	349.0	
2	B096MSW6CT	Sounce Fast Phone Charging Cable & Data Sync U...	Computers&Accessories|Accessories&Peripherals|...	199.0	1899.0	
3	B08HDJ86NZ	boAt Deuce USB 300 2 in 1 Type-C & Micro USB S...	Computers&Accessories|Accessories&Peripherals|...	329.0	699.0	
4	B08CF3B7N1	Portronics Konnect L 1.2M Fast Charging 3A 8 P...	Computers&Accessories|Accessories&Peripherals|...	154.0	399.0	
...	...	...	...	...	...	
1460	B08L7J3T31	Noir Aqua - 5pcs PP Spun Filter + 1 Spanner | ...	Home&Kitchen|Kitchen&HomeAppliances|WaterPurif...	379.0	919.0	
1461	B01M6453MB	Prestige Delight PRWO Electric Rice Cooker (1 ...	Home&Kitchen|Kitchen&HomeAppliances|SmallKitch...	2280.0	3045.0	

	product_id	product_name	category	discounted_price	actual_price	disc
1462	B009P2LIL4	Bajaj Majesty RX10 2000 Watts Heat Convector R...	Home&Kitchen|Heating,Cooling&AirQuality|RoomHe...	2219.0	3080.0	
1463	B00J5DYCCA	Havells Ventil Air DSP 230mm Exhaust Fan (Pist...	Home&Kitchen|Heating,Cooling&AirQuality|Fans|E...	1399.0	1890.0	
1464	B01486F4G6	Borosil Jumbo 1000-Watt Grill Sandwich Maker (...	Home&Kitchen|Kitchen&HomeAppliances|SmallKitch...	2863.0	3690.0	

1465 rows × 17 columns

In [272]:

```
df.describe()
```

Out[272]:

	discounted_price	actual_price	discount_percentage	rating	rating_count
count	1465.000000	1465.000000	1465.000000	1465.000000	1465.000000
mean	3125.310874	5444.990635	47.691468	4.093788	18270.564505
std	6944.304394	10874.826864	21.635905	0.310598	42729.995315
min	39.000000	39.000000	0.000000	0.000000	0.000000
25%	325.000000	800.000000	32.000000	4.000000	1173.000000
50%	799.000000	1650.000000	50.000000	4.100000	5178.000000
75%	1999.000000	4295.000000	63.000000	4.300000	17325.000000
max	77990.000000	139900.000000	94.000000	5.000000	426973.000000

In [ ]:

```
# >> there is no any duplicate value in the dataset
```

Spotify Data:Popular Hip-Hop Artists and Tracks

In [265]:

```
data=pd.read_csv('spotify.csv')
data
```

Out[265]:

	Artist	Track Name	Popularity	Duration (ms)	Track ID
0	Drake	Rich Baby Daddy (feat. Sexyy Red & SZA)	92	319191	1yeB8MUNeLo9Ek1UEpsyz6
1	Drake	One Dance	91	173986	1zi7xx7UVEFkmKfv06H8x0
2	Drake	IDGAF (feat. Yeat)	90	260111	2YSzYUF3jWqb9YP9VXmpjE
3	Drake	First Person Shooter (feat. J. Cole)	88	247444	7aqfrAY2p9BUSiupwk3svU
4	Drake	Jimmy Cooks (feat. 21 Savage)	88	218364	3F5CgOj3wFlRv51JsHbxhe
...	...	...	...	...	...
435	French Montana	Splash Brothers	44	221863	3fBsEOnzwtlkpS0LxXAZhN
436	Fat Joe	All The Way Up (feat. Infared)	64	191900	7EzwtgfW7khBrpvaNPtMoT

	Artist	Track Name	Popularity	Duration (ms)	Track ID
437	A$AP Ferg	Work REMIX (feat. A$AP Rocky, French Montana, ...	69	220000	7xVLFuuYdAvcTfcP3JG3dS
438	Diddy	Another One Of Me (feat. 21 Savage)	65	220408	4hGmQboiou09EwhcTWa0H6
439	Rick Ross	Stay Schemin	68	267720	0nq6sfr8z1R5KJ4XUk396e

440 rows × 5 columns

In [266]:

```
data.shape
```

Out[266]:

```
(440, 5)
```

In [268]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 440 entries, 0 to 439
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   Artist          440 non-null    object
 1   Track Name      440 non-null    object
 2   Popularity      440 non-null    int64 
 3   Duration (ms)   440 non-null    int64 
 4   Track ID        440 non-null    object
dtypes: int64(2), object(3)
memory usage: 17.3+ KB
```

In [269]:

```
data.dtypes
```

Out[269]:

```
Artist           object
Track Name       object
Popularity        int64
Duration (ms)     int64
Track ID         object
dtype: object
```

In [271]:

```
data.describe(include='all')
```

Out[271]:

	Artist	Track Name	Popularity	Duration (ms)	Track ID
count	440	440	440.000000	440.000000	440
unique	115	412	NaN	NaN	413
top	Drake	Annihilate (Spider-Man: Across the Spider-Vers...	NaN	NaN	39MK3d3fonIP8Mz9oHCTBB
freq	20	3	NaN	NaN	3
mean	NaN	NaN	75.736364	206810.040909	NaN
std	NaN	NaN	9.886534	53576.930289	NaN
min	NaN	NaN	29.000000	81666.000000	NaN
25%	NaN	NaN	70.000000	172778.500000	NaN
50%	NaN	NaN	77.000000	201866.000000	NaN
75%	NaN	NaN	83.000000	235119.750000	NaN
max	NaN	NaN	97.000000	501648.000000	NaN

1.Load the dataframe and ensure data quality by checking for missing values and duplicate rows. Handle missing values and remove duplicate rows if necessary.

In [274]:

```
data.isna().sum()
```

Out[274]:

```
Artist           0
Track Name       0
Popularity       0
Duration (ms)    0
Track ID         0
dtype: int64
```

In [ ]:

```
# >insights There is no any null values
```

In [281]:

```
data.duplicated().sum()
```

Out[281]:

```
27
```

In [ ]:

```
# there are 27 duplicated values in the dataset
```

In [286]:

```
data.drop_duplicates(inplace=True)
```

In [287]:

```
data
```

Out[287]:

	Artist	Track Name	Popularity	Duration (ms)	Track ID
0	Drake	Rich Baby Daddy (feat. Sexyy Red & SZA)	92	319191	1yeB8MUNeLo9Ek1UEpsyz6
1	Drake	One Dance	91	173986	1zi7xx7UVEFkmKfv06H8x0
2	Drake	IDGAF (feat. Yeat)	90	260111	2YSzYUF3jWqb9YP9VXmpjE
3	Drake	First Person Shooter (feat. J. Cole)	88	247444	7aqfrAY2p9BUSiupwk3svU
4	Drake	Jimmy Cooks (feat. 21 Savage)	88	218364	3F5CgOj3wFlRv51JsHbxhe
...	...	...	...	...	...
433	French Montana	Stand United	54	163971	01CHrTerCzyRpMl1MzQ4fz
434	Jason Derulo	Tip Toe (feat. French Montana)	65	187521	0TY3jVGwGDwDabLyQLVRQQ
436	Fat Joe	All The Way Up (feat. Infared)	64	191900	7EzwtgfW7khBrpvaNPtMoT
437	A$AP Ferg	Work REMIX (feat. A$AP Rocky, French Montana, ...	69	283693	7xVLFuuYdAvcTfcP3IG3dS
438	Diddy	Another One Of Me (feat. 21 Savage)	65	220408	4hGmQboiou09EwhcTWa0H6

413 rows × 5 columns

In [288]:

```
data.shape
```

Out[288]:

```
(413, 5)
```

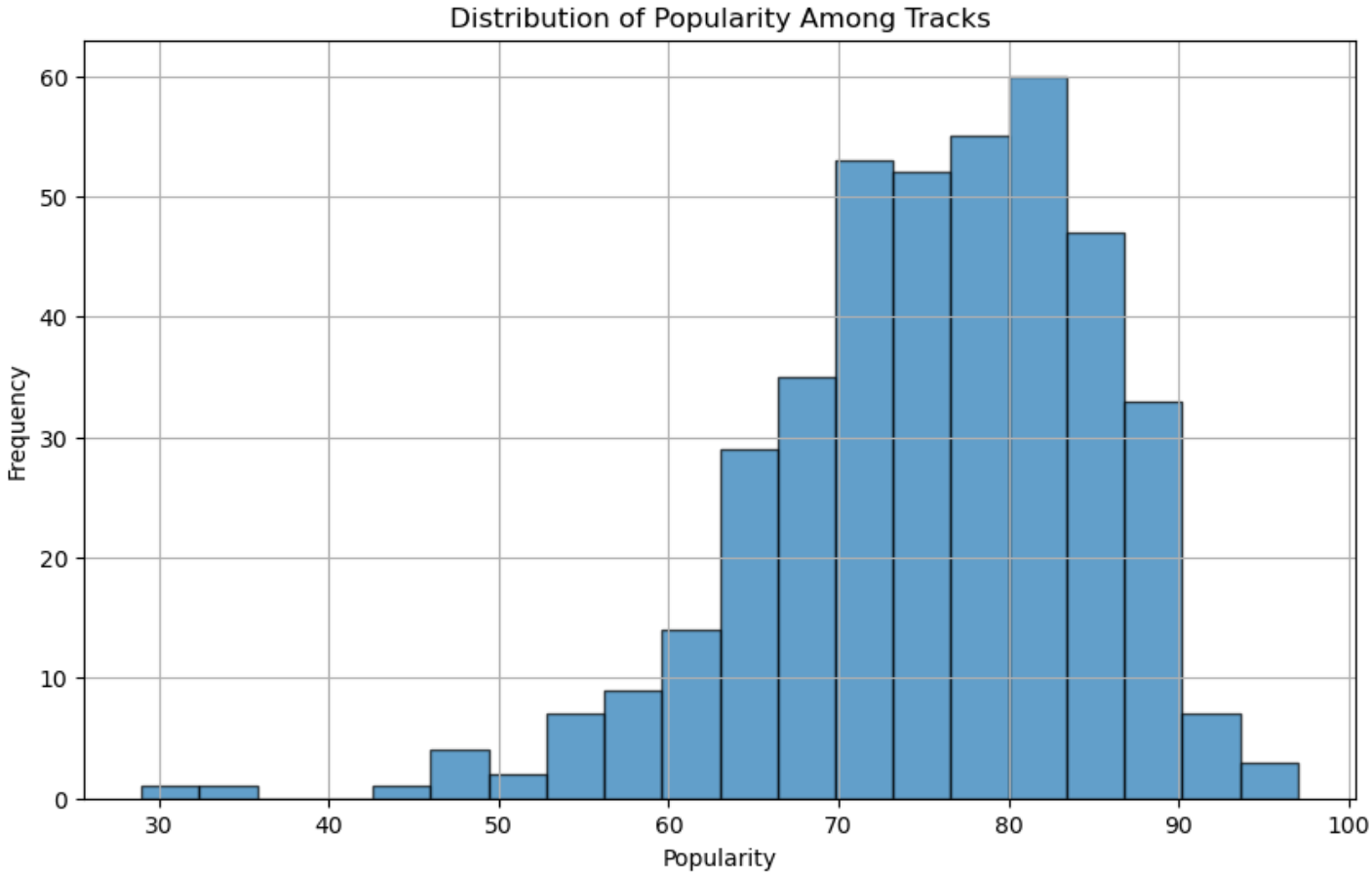
In [ ]:

```
# After removing the duplicate values the shape of the dataset is (413,5)
```

2.What is the distribution of popularity among the tracks in the dataset? Visualize it using a histogram.

In [293]:

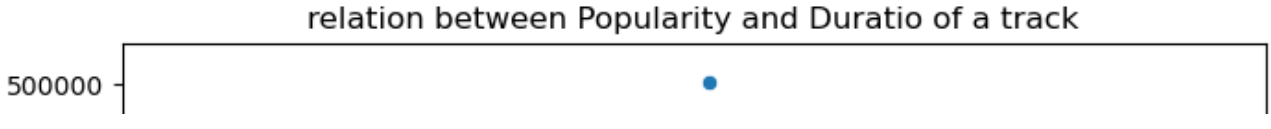
```
# Plot the distribution of popularity
plt.figure(figsize=(10, 6))
plt.hist(data['Popularity'], bins=20, edgecolor='k', alpha=0.7)
plt.title('Distribution of Popularity Among Tracks')
plt.xlabel('Popularity')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```

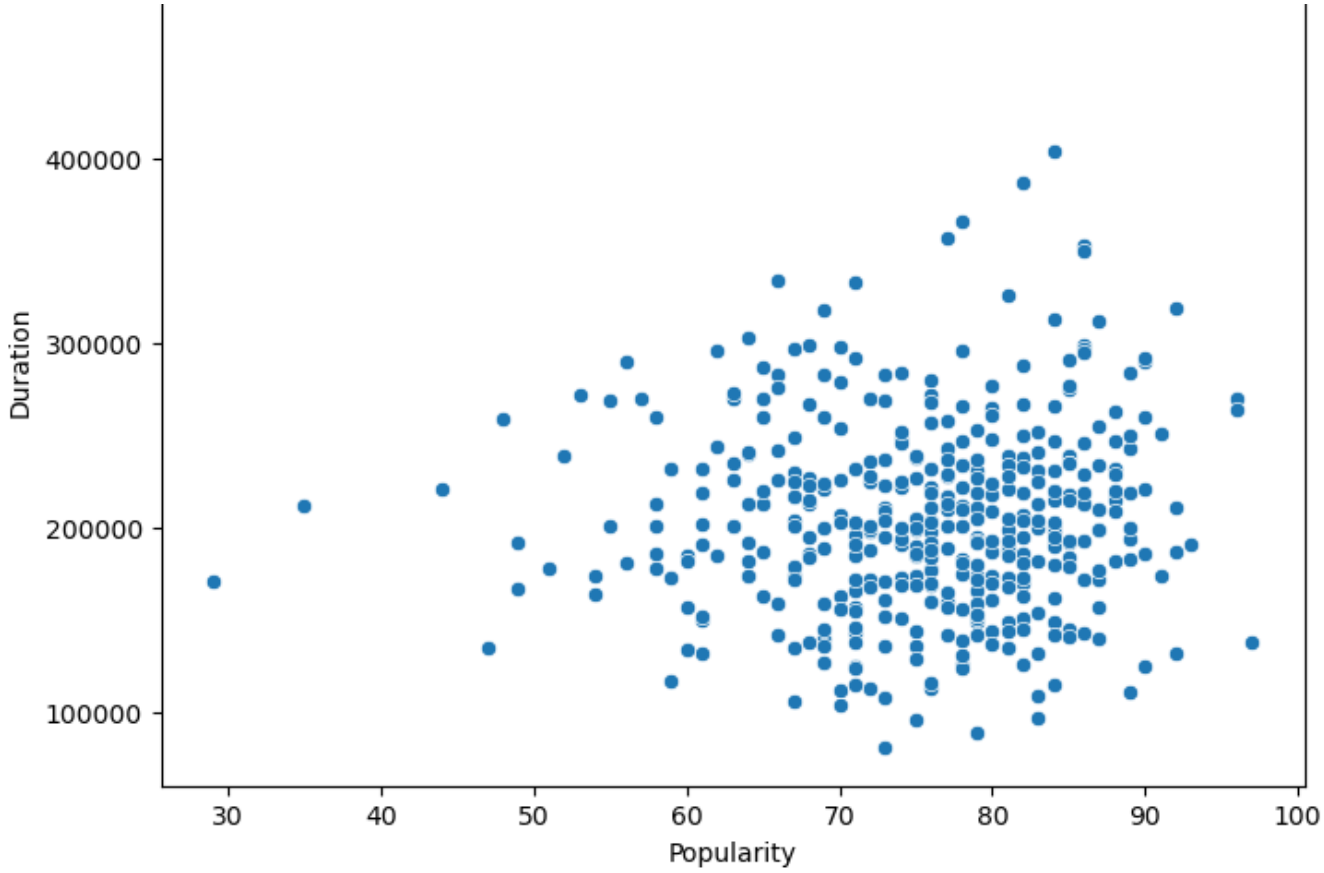


3.Is there any relationship between the popularity and the duration of tracks? Explore this using a scatter plot.

In [296]:

```
plt.figure(figsize=(8,6))
sns.scatterplot(data=data,x='Popularity',y='Duration (ms)')
plt.xlabel('Popularity')
plt.ylabel('Duration')
plt.title('relation between Popularity and Duratio of a track')
plt.show()
```





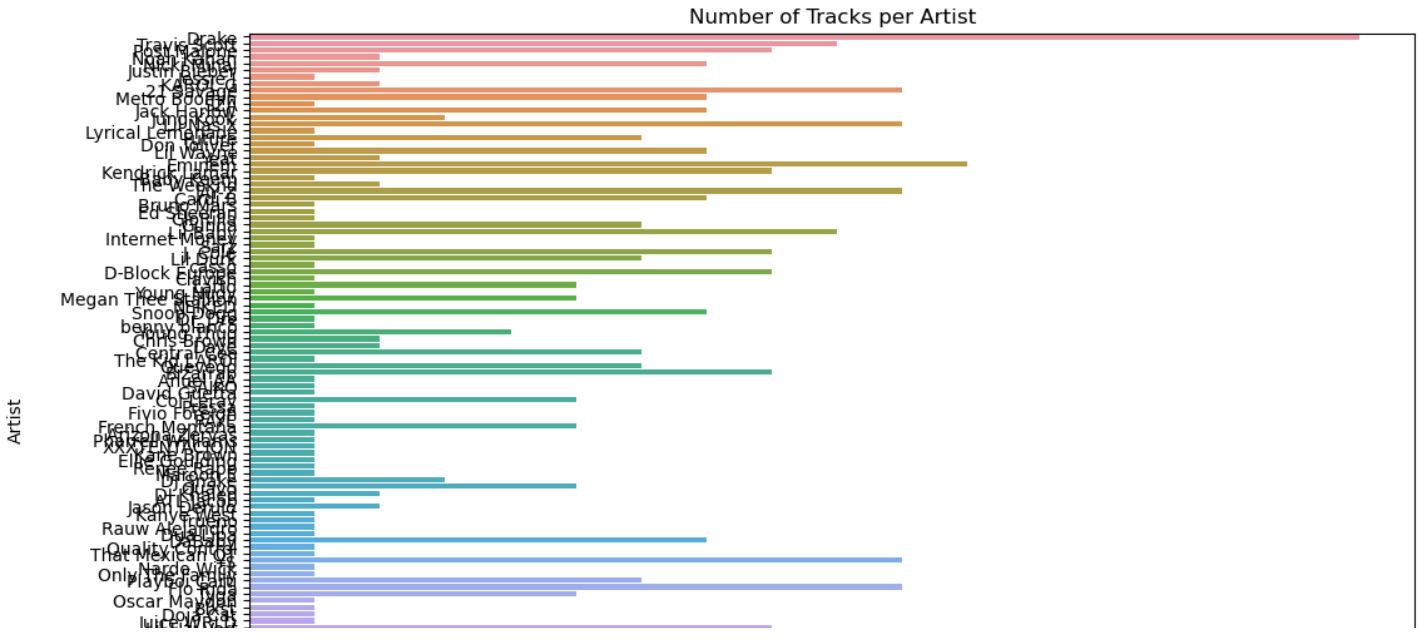
In [303]:

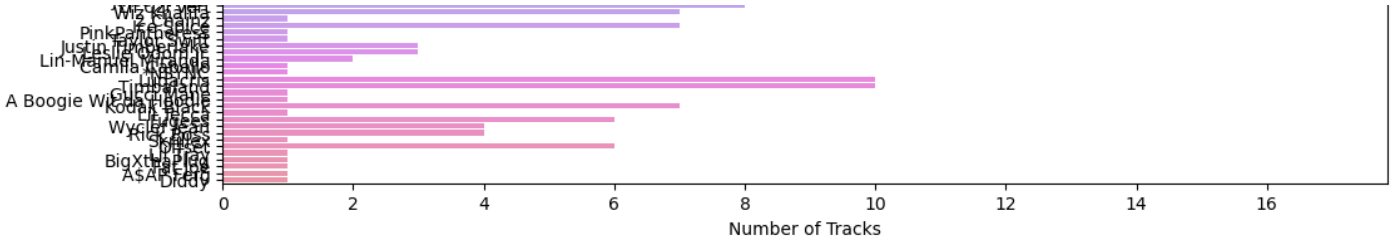
```
artist_track_counts = data['Artist'].value_counts()
print(artist_track_counts.head())
```

```
Drake          17
Eminem         11
Flo Rida       10
Ludacris       10
Timbaland      10
Name: Artist, dtype: int64
```

In [306]:

```
# Plot the count of tracks for each artist
plt.figure(figsize=(12, 8))
sns.countplot(data=data, y='Artist')
plt.title('Number of Tracks per Artist')
plt.xlabel('Number of Tracks')
plt.ylabel('Artist')
plt.show()
```





5.What are the top 5 least popular tracks in the dataset? Provide the artist name and track name for each.

In [353]:

```
data_sorted=data['Popularity'].value_counts().sort_values(ascending=True)
least_scored=df_sorted[['Artist', 'Track Name', 'Popularity']].head(5)
least_scored
```

Out[353]:

	Artist	Track Name	Popularity
207	**Pressa**	**Attachments (feat. Coi Leray)**	**29**
231	**Justin Bieber**	**Intentions**	**35**
413	**French Montana**	**Splash Brothers**	**44**
225	**Lil Baby**	**On Me - Remix**	**47**
407	**Wyclef Jean**	**911 (feat. Mary J. Blige)**	**48**

6.Among the top 5 most popular artists, which artist has the highest popularity on average?Calculate and display the average popularity for each artist

In [378]:

```
sorted_value=data.groupby('Artist')['Popularity'].mean()
```

In [396]:

```
average=sorted_value.sort_values(ascending=False)[:5]
```

In [414]:

```
average
```

Out[414]:

```
Artist
cassö           92.000000
Trueno          89.000000
David Guetta    87.000000
Travis Scott    86.555556
¥$              85.100000
Name: Popularity, dtype: float64
```

7.For the top 5 most popular artists, what are their most popular tracks? List the track name for each artist.

In [413]:

```
# Assuming your dataframe is called 'df'
sorted_by_sales = data.sort_values(by='Popularity',ascending=False)  # Sort by sales (as
cending)

# Get artist and track name for the top 5 least popular tracks
top5 = sorted_by_sales.head(5)[['Artist','Popularity' ,'Track Name']]
```

```
top5
```

Out[413]:

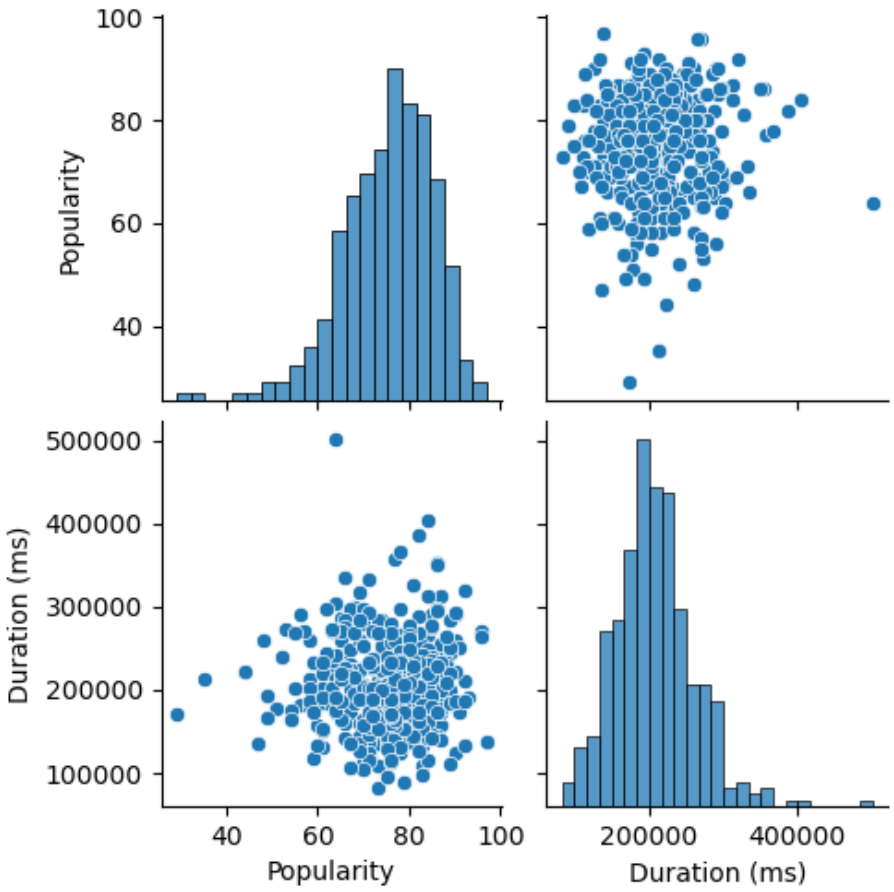
	Artist	Popularity	Track Name
40	Jack Harlow	97	Lovin On Me
70	21 Savage	96	redrum
260	¥$	96	CARNIVAL
30	Travis Scott	93	FE!N (feat. Playboi Carti)
140	cassö	92	Prada

8.Visualize relationships between multiple numerical variables simultaneously using a pair plot.

In [419]:

```
plt.figure(figsize=(8,8))
sns.pairplot(data)
plt.show()
```

```
<Figure size 800x800 with 0 Axes>
```



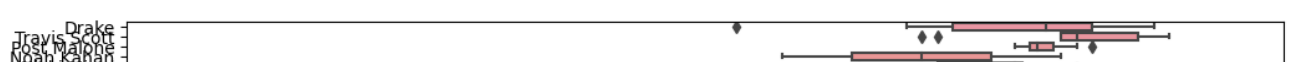
9.Does the duration of tracks vary significantly across different artists? Explore this visually using a box plot or violin plot.

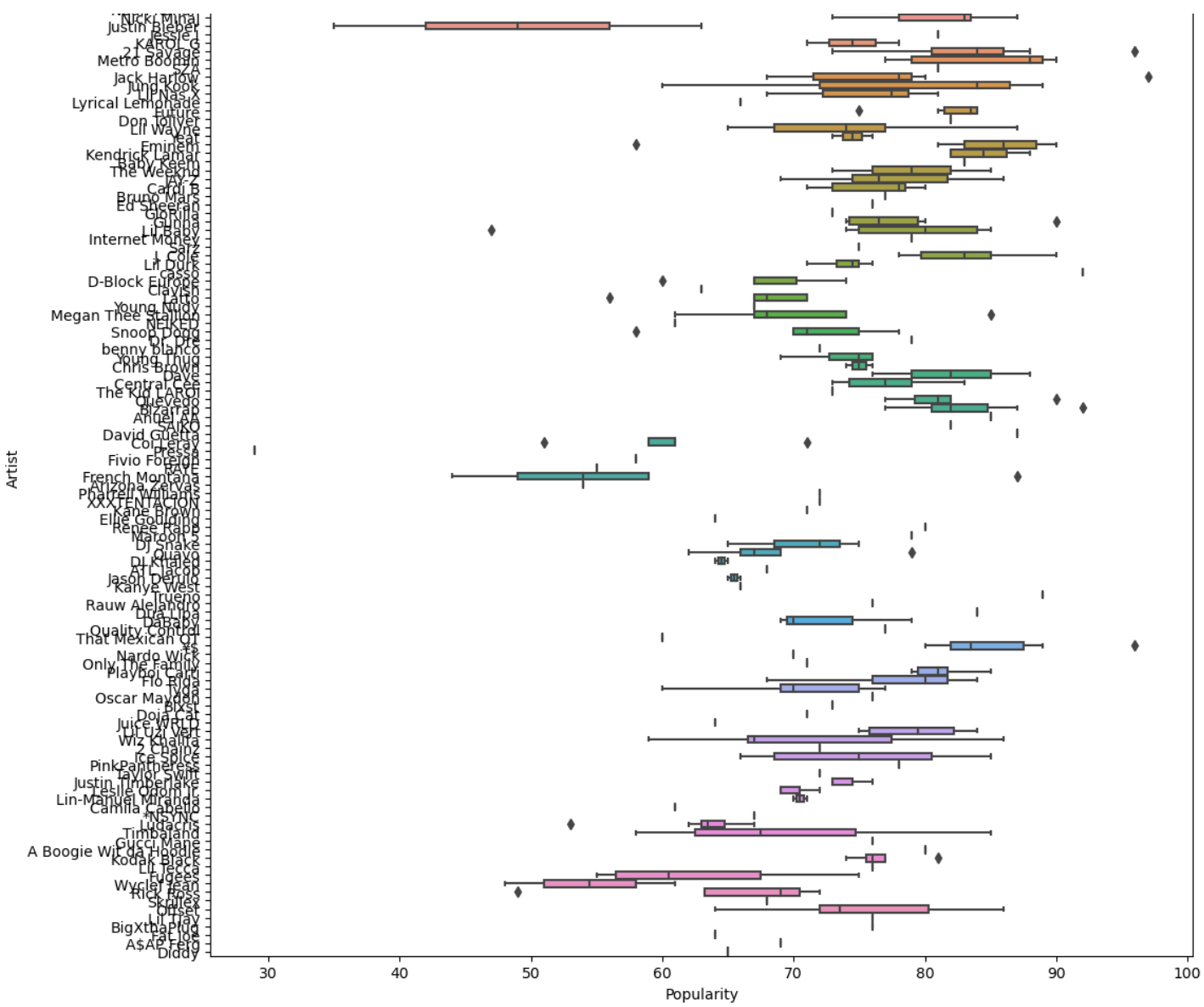
In [424]:

```
plt.figure(figsize=(12,12))
sns.boxplot(data=data,x='Popularity',y='Artist')
```

Out[424]:

```
<Axes: xlabel='Popularity', ylabel='Artist'>
```





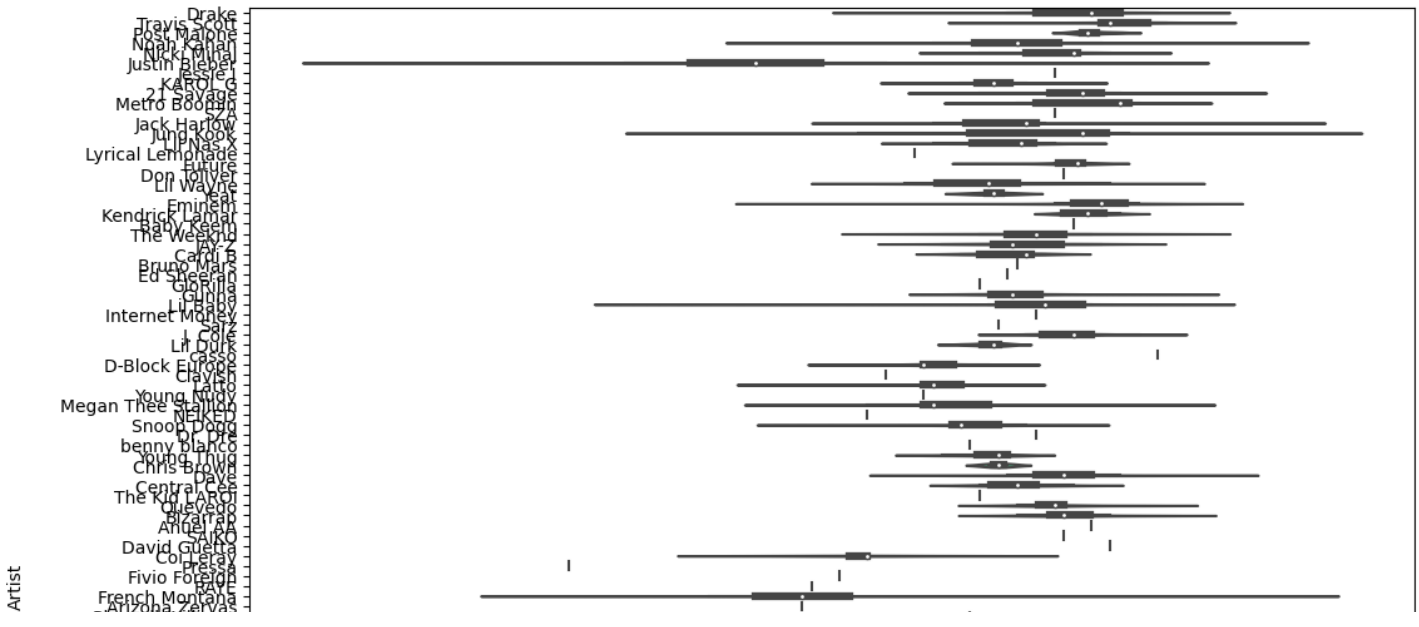
10.How does the distribution of track popularity vary for different artists? Visualize this using a swarm plot or a violin plot.

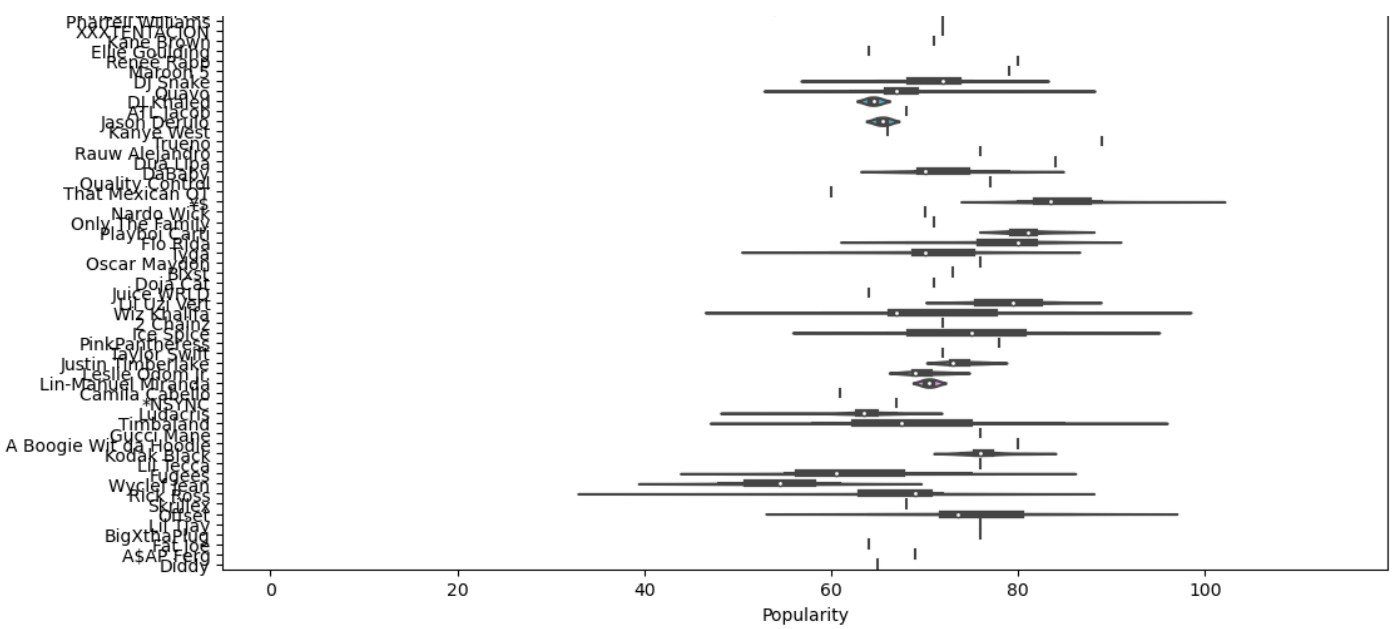
In [429]:

```
plt.figure(figsize=(12,12))
sns.violinplot(data=data,x='Popularity',y='Artist')
```

Out[429]:

```
<Axes: xlabel='Popularity', ylabel='Artist'>
```




Pharrell Williams
XXXTENTACION
Kane Brown
Elle Goulding
Renee Rapp
Maroon 5
DJ Snake
Quavo
DJ Khaled
Ari Lacob
Jason Derulo
Kanye West
Trueno
Rauw Alejandro
Dua Lipa
Dababy
Quality Control
That Mexican OT
¥$
Nardo Wick
Only The Family
PlayThatCarti
Flo Rida
Tyga
Oscar Maydon
BXSH
Doja Cat
Juice WRLD
Lil Uzi Vert
Wiz Khalifa
J Balvin
Ice Spice
PinkPantheress
Taylor Swift
Justin Timberlake
Leslie Odom Jr.
Lin-Manuel Miranda
Camila Cabello
*NSYNC
Ludacris
Timbaland
Gucci Mane
A Boogie Wit da Hoodie
Kodak Black
Lil Tecca
Rudeus
Wyclef Jean
Rick Ross
Skrillex
Offset
Lil Tjay
BigXthaPlug
Fat Joe
A$AP Ferg
Diddy
0
20
40
60
80
100
Popularity

In [ ]:

In [ ]: