In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
data = pd.read_csv('youtube_data.csv' , encoding='latin1')
data
```

Out[2]:

	rank	Youtuber	subscribers	video views	category	Title	uploads	Country of origin	Country	Abbreviation	...	sub
0	1	T-Series	245000000.0	2.280000e+11	Music	T-Series	20082	India	india	IN	...	
1	2	YouTube Movies	170000000.0	0.000000e+00	Film & Animation	youtubemovies	1	United States	United States	US	...	
2	3	MrBeast	166000000.0	2.836884e+10	Entertainment	MrBeast	741	United States	United States	US	...	
3	4	Cocomelon - Nursery Rhymes	162000000.0	1.640000e+11	Education	Cocomelon - Nursery Rhymes	966	United States	United States	US	...	
4	5	SET India	159000000.0	1.480000e+11	Shows	SET India	116536	India	India	IN	...	
...	...	...	...	...	...	...	...	...	...	...	...	
1001	779	The Dodo	14200000.0	9.964117e+09	Pets & Animals	Dorukhan Gï¿½ï¿½ï	8	NaN	NaN	NaN	...	
1002	787	Supercar Blondie	14100000.0	5.405563e+09	Autos & Vehicles	Supercar Blondie	855	United Arab Emirates	United Arab Emirates	AE	...	
1003	871	Just For Laughs Gags	13300000.0	7.406629e+09	Comedy	Just For Laughs Gags	6916	United States	United States	US	...	
1004	872	Kabita's Kitchen	13300000.0	2.831276e+09	Howto & Style	Kabita's Kitchen	1489	India	India	IN	...	
1005	873	BanderitaX	13300000.0	4.129249e+09	Gaming	BanderitaX	1640	Saudi Arabia	Saudi Arabia	SA	...	

1006 rows × 29 columns

In [3]:
```
data.columns
```

Out[3]:
```
Index(['rank', 'Youtuber', 'subscribers', 'video views', 'category', 'Title',
       'uploads', 'Country of origin', 'Country', 'Abbreviation',
       'channel_type', 'video_views_rank', 'country_rank', 'channel_type_rank',
       'video_views_for_the_last_30_days', 'lowest_monthly_earnings',
       'highest_monthly_earnings', 'lowest_yearly_earnings',
       'highest_yearly_earnings', 'subscribers_for_last_30_days',
       'created_year', 'created_month', 'created_date',
       'Gross tertiary education enrollment (%)', 'Population',
       'Unemployment rate', 'Urban_population', 'Latitude', 'Longitude'],
      dtype='object')
```

In [4]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1006 entries, 0 to 1005
Data columns (total 29 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   rank                                    1006 non-null   int64
 1   Youtuber                                1006 non-null   object
 2   subscribers                             1003 non-null   float64
 3   video views                             1006 non-null   float64
 4   category                                951 non-null    object
 5   Title                                   1006 non-null   object
 6   uploads                                 1006 non-null   int64
 7   Country of origin                       881 non-null    object
 8   Country                                 881 non-null    object
 9   Abbreviation                            881 non-null    object
 10  channel_type                            974 non-null    object
 11  video_views_rank                        1005 non-null   float64
 12  country_rank                            887 non-null    float64
 13  channel_type_rank                       971 non-null    float64
 14  video_views_for_the_last_30_days        949 non-null    float64
 15  lowest_monthly_earnings                 1006 non-null   float64
 16  highest_monthly_earnings                1006 non-null   float64
 17  lowest_yearly_earnings                  1006 non-null   float64
 18  highest_yearly_earnings                 1006 non-null   float64
 19  subscribers_for_last_30_days            666 non-null    float64
 20  created_year                            1001 non-null   float64
 21  created_month                           994 non-null    object
 22  created_date                            1001 non-null   float64
 23  Gross tertiary education enrollment (%) 880 non-null    float64
 24  Population                              880 non-null    float64
 25  Unemployment rate                       880 non-null    float64
 26  Urban_population                        880 non-null    float64
 27  Latitude                                880 non-null    float64
 28  Longitude                               880 non-null    float64
dtypes: float64(19), int64(2), object(8)
memory usage: 228.0+ KB
```

In [5]:
```
data.describe()
```

Out[5]:

	rank	subscribers	video views	uploads	video_views_rank	country_rank	channel_type_rank	video_views_f
count	1006.000000	1.003000e+03	1.006000e+03	1006.000000	1.005000e+03	887.000000	971.000000	
mean	497.472167	2.319501e+07	1.112411e+10	9168.335984	5.607670e+05	384.289741	742.311020	
std	288.738758	1.783047e+07	1.424148e+10	34028.189437	1.368886e+06	1227.359768	1938.126477	
min	1.000000	1.230000e+07	0.000000e+00	0.000000	1.000000e+00	1.000000	1.000000	
25%	247.250000	1.450000e+07	4.281427e+09	194.000000	3.220000e+02	11.000000	26.000000	
50%	498.500000	1.770000e+07	7.751292e+09	726.500000	9.190000e+02	50.000000	65.000000	
75%	748.750000	2.475000e+07	1.357357e+10	2606.500000	3.645000e+03	123.000000	139.000000	
max	995.000000	2.450000e+08	2.280000e+11	301308.000000	4.057944e+06	7741.000000	7741.000000	

8 rows × 21 columns

In [6]:
```
data.isna().any()
```

Out[6]:
```
rank                                    False
Youtuber                                False
subscribers                              True
video views                             False
category                                 True
Title                                   False
uploads                                 False
Country of origin                        True
Country                                  True
Abbreviation                             True
channel_type                             True
video_views_rank                         True
country_rank                             True
channel_type_rank                        True
video_views_for_the_last_30_days         True
lowest_monthly_earnings                 False
highest_monthly_earnings                False
lowest_yearly_earnings                  False
highest_yearly_earnings                 False
subscribers_for_last_30_days             True
created_year                             True
created_month                            True
created_date                             True
Gross tertiary education enrollment (%)  True
Population                               True
Unemployment rate                        True
Urban_population                         True
Latitude                                 True
Longitude                                True
dtype: bool
```

What are the top 10 YouTube channels based on the number of subscribers?

In [7]:
```
top_10_subscribers = data.nlargest(10, 'subscribers')[['Youtuber', 'subscribers']]
print('<------------------------------------------------>')
print("Top 10 YouTube channels based on subscribers:")
print(top_10_subscribers)
print('<------------------------------------------------>')
```

```
<------------------------------------------------>
Top 10 YouTube channels based on subscribers:
                    Youtuber  subscribers
0                    T-Series  245000000.0
1               YouTube Movies  170000000.0
2                     MrBeast  166000000.0
3  Cocomelon - Nursery Rhymes  162000000.0
4                   SET India  159000000.0
5                       Music  119000000.0
6         ýýý Kids Diana Show  112000000.0
7                   PewDiePie  111000000.0
8                 Like Nastya  106000000.0
9               Vlad and Niki   98900000.0
<------------------------------------------------>
```

Which category has the highest average number of subscribers?

In [8]:
```
avg_subscribers_by_category = data.groupby('category')['subscribers'].mean().idxmax()
print('<------------------------------------------------>')
print("Category with the highest average subscribers:", avg_subscribers_by_category)
print('<------------------------------------------------>')
```

```
<------------------------------------------------>
Category with the highest average subscribers: Shows
<------------------------------------------------>
```

How many videos, on average, are uploaded by YouTube channels in each category?

In [9]:
```
avg_videos_by_category = data.groupby('category')['uploads'].mean()
print('<------------------------------------------------>')
print("Average videos uploaded by category:")
print(avg_videos_by_category)
print('<------------------------------------------------>')
```

```
<---------------------------------------------->
Average videos uploaded by category:
category
Autos & Vehicles           1550.666667
Comedy                     1202.557143
Education                  3087.086957
Entertainment             12052.445378
Film & Animation           2861.844444
Gaming                     4285.273684
Howto & Style              1695.500000
Movies                     3553.000000
Music                      2325.945813
News & Politics          112484.384615
Nonprofits & Activism    102912.000000
People & Blogs             9256.793893
Pets & Animals             3562.800000
Science & Technology       2114.058824
Shows                     27443.692308
Sports                    19129.833333
Trailers                   6839.000000
Travel & Events             766.000000
Name: uploads, dtype: float64
<---------------------------------------------->
```

What are the top 5 countries with the highest number of YouTube channels?

In [10]:
```
top_5_countries = data['Country'].value_counts().nlargest(5)
print('<---------------------------------------------->')
print("Top 5 countries with the highest number of YouTube channels:")
print(top_5_countries)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Top 5 countries with the highest number of YouTube channels:
Country
United States     315
India             169
Brazil             62
United Kingdom     44
Mexico             33
Name: count, dtype: int64
<---------------------------------------------->
```

What is the distribution of channel types (individual vs. brand) across different categories?

In [11]:
```
channel_type_distribution = data.groupby(['category', 'channel_type']).size()
print('<---------------------------------------------->')
print("Distribution of channel types across categories:")
print(channel_type_distribution)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Distribution of channel types across categories:
category          channel_type
Autos & Vehicles  Autos             2
                  Entertainment     1
Comedy            Comedy           39
                  Entertainment    20
                  Film              1
                                   ..
Sports            Entertainment     1
                  Sports           11
Trailers          Entertainment     1
                  Music             1
Travel & Events   Entertainment     1
Length: 79, dtype: int64
<---------------------------------------------->
```

Is there a correlation between the number of subscribers and total video views for YouTube channels?

In [12]:
```
subscribers_views_correlation = data['subscribers'].corr(data['video views'])
print('<---------------------------------------------->')
print("Correlation between subscribers and total video views:", subscribers_views_correlation)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Correlation between subscribers and total video views: 0.7481786016237687
<---------------------------------------------->
```

How do the monthly earnings vary between individual and brand YouTube channels?

In [13]:
```
monthly_earnings_variation = data.groupby('channel_type')[['lowest_monthly_earnings', 'highest_monthly_earnings
```

```
print('<------------------------------------------------>')
print("Monthly earnings variation between individual and brand channels:")
print(monthly_earnings_variation)
print('<------------------------------------------------>')
```

```
<------------------------------------------------>
Monthly earnings variation between individual and brand channels:
              lowest_monthly_earnings  highest_monthly_earnings
channel_type
Animals                 176566.666667              2.833333e+06
Autos                    44150.157500              7.000026e+05
Comedy                   45869.729038              7.315022e+05
Education                50188.000000              8.051820e+05
Entertainment            43972.672007              7.027955e+05
Film                     29086.429524              4.646072e+05
Games                    21677.960300              3.480582e+05
Howto                    14665.648649              2.339649e+05
Music                    35866.994977              5.726846e+05
News                     43756.875000              7.016400e+05
Nonprofit                24400.000000              3.904000e+05
People                   41823.474902              6.691181e+05
Sports                   50778.571429              8.197357e+05
Tech                     13782.352941              2.203588e+05
<------------------------------------------------>
```

What is the overall trend in subscribers gained in the last 30 days across all channels?

In [14]:
```
overall_subscribers_trend = data['subscribers_for_last_30_days'].sum()
print('<------------------------------------------------>')
print("Overall trend in subscribers gained in the last 30 days:", overall_subscribers_trend)
print('<------------------------------------------------>')
```

```
<------------------------------------------------>
Overall trend in subscribers gained in the last 30 days: 232794874.0
<------------------------------------------------>
```

Are there any outliers in terms of yearly earnings from YouTube channels?

In [15]:
```
yearly_earnings_outliers = data[(data['lowest_yearly_earnings'] < 0) | (data['highest_yearly_earnings'] < 0)]
print('<------------------------------------------------>')
print("Channels with negative yearly earnings:")
print(yearly_earnings_outliers)
print('<------------------------------------------------>')
```

```
<------------------------------------------------>
Channels with negative yearly earnings:
Empty DataFrame
Columns: [rank, Youtuber, subscribers, video views, category, Title, uploads, Country of origin, Country, Abbrev
iation, channel_type, video_views_rank, country_rank, channel_type_rank, video_views_for_the_last_30_days, lowes
t_monthly_earnings, highest_monthly_earnings, lowest_yearly_earnings, highest_yearly_earnings, subscribers_for_l
ast_30_days, created_year, created_month, created_date, Gross tertiary education enrollment (%), Population, Une
mployment rate, Urban_population, Latitude, Longitude]
Index: []

[0 rows x 29 columns]
<------------------------------------------------>
```

What is the distribution of channel creation dates? Is there any trend over time?

In [16]:
```
channel_creation_dates_distribution = data.groupby(['created_year', 'created_month']).size()
print('<------------------------------------------------>')
print("Distribution of channel creation dates:")
print(channel_creation_dates_distribution)
print('<------------------------------------------------>')
```

```
<------------------------------------------------>
Distribution of channel creation dates:
created_year  created_month
1970.0        Jan              1
2005.0        Dec              3
              Jun              2
              Nov              9
              Oct              6
                              ..
2021.0        Sep              2
2022.0        Apr              1
              Jun              2
              Mar              1
              May              1
Length: 198, dtype: int64
<------------------------------------------------>
```

Is there a relationship between gross tertiary education enrollment and the number of YouTube channels in a

country?

In [17]:
```
education_enrollment_relationship = data[['Gross tertiary education enrollment (%)', 'Country']].drop_duplicates
print('<------------------------------------------------>')
print("Relationship between education enrollment and number of channels:")
print(education_enrollment_relationship)
print('<------------------------------------------------>')
```

```
<------------------------------------------------>
Relationship between education enrollment and number of channels:
     Gross tertiary education enrollment (%)               Country
0                                       28.1                 india
1                                       88.2         United States
4                                       28.1                 India
5                                        NaN                   NaN
7                                       63.2                 Japan
8                                       81.9                Russia
13                                      94.3           South Korea
16                                      60.0        United Kingdom
19                                      68.9                Canada
23                                      51.3                Brazil
31                                      90.0             Argentina
44                                      88.5                 Chile
50                                      41.4                  Cuba
52                                      29.4           El Salvador
56                                       9.0              Pakistan
60                                      35.5           Philippines
61                                      49.3              Thailand
63                                      55.3              Colombia
69                                      65.4              Barbados
71                                      40.2                Mexico
76                                      36.8  United Arab Emirates
81                                      88.9                 Spain
85                                      68.0          Saudi Arabia
117                                     36.3             Indonesia
132                                     23.9                Turkey
151                                     79.3             Venezuela
158                                     54.4                Kuwait
165                                     34.4                Jordan
167                                     85.0           Netherlands
190                                     84.8             Singapore
191                                    113.1             Australia
212                                     61.9                 Italy
294                                     70.2               Germany
315                                     65.6                France
371                                     67.0                Sweden
377                                      9.7           Afghanistan
379                                     82.7               Ukraine
388                                     88.1                Latvia
423                                     59.6           Switzerland
489                                     28.5               Vietnam
503                                     45.1              Malaysia
507                                     50.6                 China
517                                     16.2                  Iraq
637                                     35.2                 Egypt
663                                      NaN               Andorra
725                                     44.9               Ecuador
749                                     35.9               Morocco
754                                     70.7                  Peru
808                                     20.6            Bangladesh
876                                     88.2               Finland
899                                      7.6                 Samoa
<------------------------------------------------>
```

How does the unemployment rate vary among the top 10 countries with the highest number of YouTube channels?

In [18]:
```
top_10_countries_unemployment = data.groupby('Country')['Unemployment rate'].mean().nlargest(10)
print('<------------------------------------------------>')
print("Unemployment rate among top 10 countries with most YouTube channels:")
print(top_10_countries_unemployment)
print('<------------------------------------------------>')
```

```
<---------------------------------------------->
Unemployment rate among top 10 countries with most YouTube channels:
Country
Jordan           14.72
United States    14.70
Spain            13.96
Turkey           13.49
Iraq             12.82
Brazil           12.08
Afghanistan      11.12
Egypt            10.76
Barbados         10.33
Italy             9.89
Name: Unemployment rate, dtype: float64
<---------------------------------------------->
```

What is the average urban population percentage in countries with YouTube channels?

In [19]:
```
avg_urban_population = data['Urban_population'].mean()
print('<---------------------------------------------->')
print("Average urban population percentage in countries with YouTube channels:", avg_urban_population)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Average urban population percentage in countries with YouTube channels: 223974718.82045454
<---------------------------------------------->
```

Are there any patterns in the distribution of YouTube channels based on latitude and longitude coordinates?

In [20]:
```
latitude_longitude_patterns = data.groupby(['Latitude', 'Longitude']).size()
print('<---------------------------------------------->')
print("Patterns in distribution of YouTube channels based on latitude and longitude:")
print(latitude_longitude_patterns)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Patterns in distribution of YouTube channels based on latitude and longitude:
Latitude    Longitude
-38.416097  -63.616672      13
-35.675147  -71.542969       3
-25.274398   133.775136      9
-14.235004  -51.925280      62
-13.759029  -172.104629      1
-9.189967   -75.015152       1
-1.831239   -78.183406       2
-0.789275    113.921327     28
 1.352083    103.819836      4
 4.210484    101.975766      1
 4.570868   -74.297333      11
 6.423750   -66.589730       1
 12.879721   121.774017     12
 13.193887  -59.543198       1
 13.794185  -88.896530       1
 14.058324   108.277199      3
 15.870032   100.992541     18
 20.593684   78.962880     170
 21.521757  -77.781167       1
 23.424076   53.847818       8
 23.634501  -102.552784     33
 23.684994   90.356331       1
 23.885942   45.079162      10
 26.820553   30.802498       2
 29.311660   47.481766       1
 30.375321   69.345116       6
 30.585164   36.238414       3
 31.791702  -7.092620        1
 33.223191   43.679291       2
 33.939110   67.709953       1
 35.861660   104.195397      1
 35.907757   127.766922     17
 36.204824   138.252924      5
 37.090240  -95.712891     315
 38.963745   35.243322       4
 40.463667  -3.749220       22
 41.871940   12.567380       2
 46.227638   2.213749        5
 46.818188   8.227512        1
 48.379433   31.165580       8
 51.165691   10.451526       6
 52.132633   5.291266        3
 55.378051  -3.435973       44
 56.130366  -106.346771     15
 56.879635   24.603189       1
 60.128161   18.643501       4
 61.524010   105.318756     16
 61.924110   25.748151       1
dtype: int64
<---------------------------------------------->
```

What is the correlation between the number of subscribers and the population of a country?

In [21]:
```
subscribers_population_correlation = data['subscribers'].corr(data['Population'])
print('<---------------------------------------------->')
print("Correlation between subscribers and population of a country:", subscribers_population_correlation)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Correlation between subscribers and population of a country: 0.08279259673577884
<---------------------------------------------->
```

How do the top 10 countries with the highest number of YouTube channels compare in terms of their total population?

In [22]:
```
top_10_countries_population = data.groupby('Country')['Population'].mean().nlargest(10)
print('<---------------------------------------------->')
print("Total population of top 10 countries with most YouTube channels:")
print(top_10_countries_population)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Total population of top 10 countries with most YouTube channels:
Country
China            1.397715e+09
India            1.366418e+09
india            1.366418e+09
United States    3.282395e+08
Indonesia        2.702039e+08
Pakistan         2.165653e+08
Brazil           2.125594e+08
Bangladesh       1.673108e+08
Russia           1.443735e+08
Japan            1.262266e+08
Name: Population, dtype: float64
<---------------------------------------------->
```

Is there a correlation between the number of subscribers gained in the last 30 days and the unemployment rate in a country?

In [23]:
```
subscribers_unemployment_correlation = data['subscribers_for_last_30_days'].corr(data['Unemployment rate'])
print('<---------------------------------------------->')
print("Correlation between subscribers gained in last 30 days and unemployment rate:", subscribers_unemployment_
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Correlation between subscribers gained in last 30 days and unemployment rate: -0.022031381943234354
<---------------------------------------------->
```

How does the distribution of video views for the last 30 days vary across different channel types?

In [24]:
```
video_views_distribution = data.groupby('channel_type')['video_views_for_the_last_30_days']
print('<---------------------------------------------->')
print("Distribution of video views for the last 30 days across channel types:")
print(video_views_distribution)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Distribution of video views for the last 30 days across channel types:
<pandas.core.groupby.generic.SeriesGroupBy object at 0x000001DBD678EA40>
<---------------------------------------------->
```

Are there any seasonal trends in the number of videos uploaded by YouTube channels?

In [25]:
```
seasonal_videos_trends = data.groupby('created_month')['uploads'].mean()
print('<---------------------------------------------->')
print("Seasonal trends in number of videos uploaded:")
print(seasonal_videos_trends)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Seasonal trends in number of videos uploaded:
created_month
Apr     2499.929577
Aug    15912.265060
Dec     3577.154930
Feb    11732.582090
Jan     8012.594059
Jul     5603.808989
Jun    13059.413333
Mar     4339.232558
May     8568.717647
Nov     8841.582418
Oct    16688.500000
Sep    12182.113402
Name: uploads, dtype: float64
<---------------------------------------------->
```

What is the average number of subscribers gained per month since the creation of YouTube channels?

In [26]:
```
data['months_since_creation'] = (pd.to_datetime('now') - pd.to_datetime(data['created_date'])).dt.days / 30
avg_subscribers_per_month = data['subscribers'] / data['months_since_creation']
print('<---------------------------------------------->')
print("Average subscribers gained per month since channel creation:")
print(avg_subscribers_per_month)
print('<---------------------------------------------->')
```

```
<---------------------------------------------->
Average subscribers gained per month since channel creation:
0       370893.677146
1       257354.796387
2       251299.389413
3       245243.982439
4       240702.427209
            ...
1001     21496.694757
1002     21345.309583
1003     20134.228188
1004     20134.228188
1005     20134.228188
Length: 1006, dtype: float64
<---------------------------------------------->
```

In [ ]: