untitled11

July 16, 2025

0.0.1 Netflix Data: Cleaning, Analysis and Visualization

[2]:
```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
```

[4]:
```
# Load the Dataset
data = pd.read_csv('netflix1.csv')
```

[8]:
```
data.head()
```

[8]:
```
  show_id     type                              title         director  \
0      s1    Movie               Dick Johnson Is Dead  Kirsten Johnson
1      s3  TV Show                          Ganglands  Julien Leclercq
2      s6  TV Show                      Midnight Mass    Mike Flanagan
3     s14    Movie  Confessions of an Invisible Girl    Bruno Garotti
4      s8    Movie                            Sankofa     Haile Gerima

         country date_added  release_year rating  duration  \
0  United States  9/25/2021          2020  PG-13    90 min
1         France  9/24/2021          2021  TV-MA  1 Season
2  United States  9/24/2021          2021  TV-MA  1 Season
3         Brazil  9/22/2021          2021  TV-PG    91 min
4  United States  9/24/2021          1993  TV-MA   125 min

                                        listed_in
0                                   Documentaries
1  Crime TV Shows, International TV Shows, TV Act…
2                TV Dramas, TV Horror, TV Mysteries
3                Children & Family Movies, Comedies
4   Dramas, Independent Movies, International Movies
```

0.0.2 Data Cleaning

[6]:
```
# Check for missing values
data.isnull().sum()
```

[6]:
```
show_id         0
type            0
title           0
director        0
country         0
date_added      0
release_year    0
rating          0
duration        0
listed_in       0
dtype: int64
```

[8]:
```
# Drop Dublicats if any
data.drop_duplicates(inplace=True)
```

[17]:
```
data.columns
```

[17]:
```
Index(['show_id', 'type', 'title', 'director', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in'],
      dtype='object')
```

[10]:
```
# Drop row with missing critical information
data.dropna(subset=['director','title','country'],inplace=True)
```

[12]:
```
# Convert data_added to datetime
data['date_added'] = pd.to_datetime(data['date_added'])
```

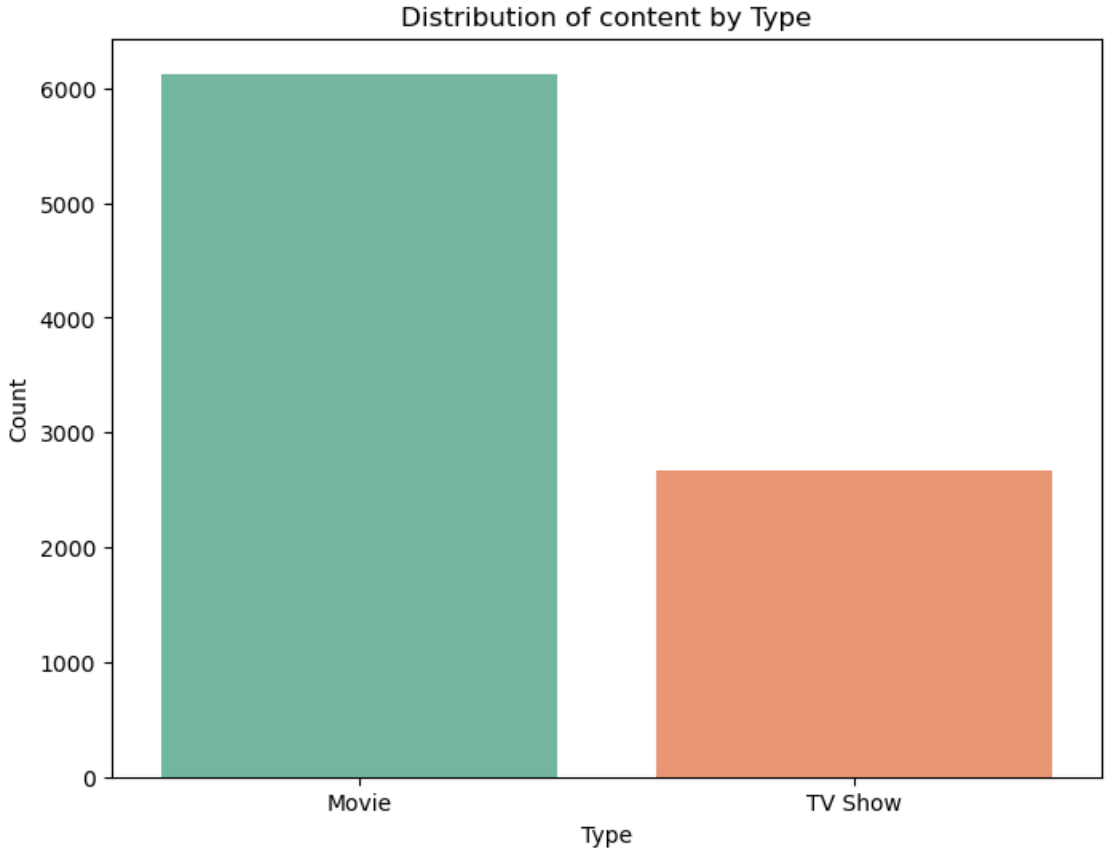
[14]:
```
# Show data types to confirm changes
print(data.dtypes)
```

```
show_id                 object
type                    object
title                   object
director                object
country                 object
date_added      datetime64[ns]
release_year             int64
rating                  object
duration                object
listed_in               object
dtype: object
```

0.0.3 Exploratory Data Analysis(EDA)

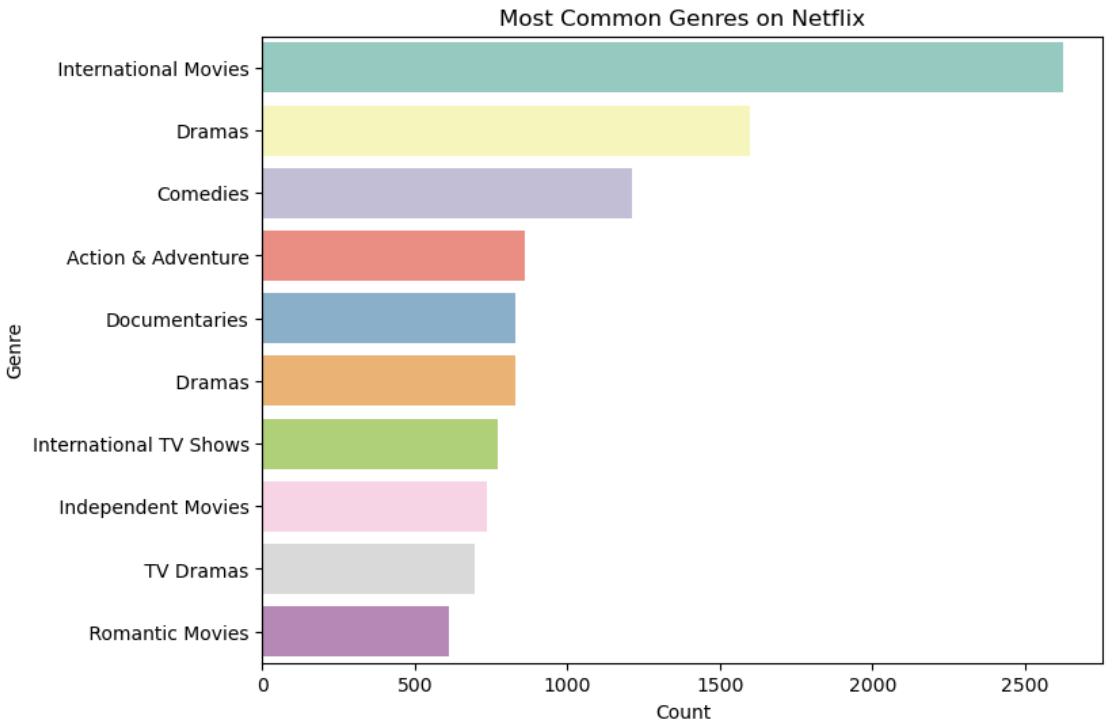
[16]:
```
# Content Type Distribution (Movies vs. TV Shows)
# Count the number of Movies and TV Show
type_counts = data['type'].value_counts()

# Plot the Distribution
plt.figure(figsize=(8,6))
sns.barplot(x=type_counts.index,y=type_counts.values,palette='Set2')
plt.title('Distribution of content by Type')
plt.xlabel('Type')
plt.ylabel('Count')
plt.show()
```

```
C:\Users\aa\AppData\Local\Temp\ipykernel_8232\1021420799.py:7: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.barplot(x=type_counts.index,y=type_counts.values,palette='Set2')
```



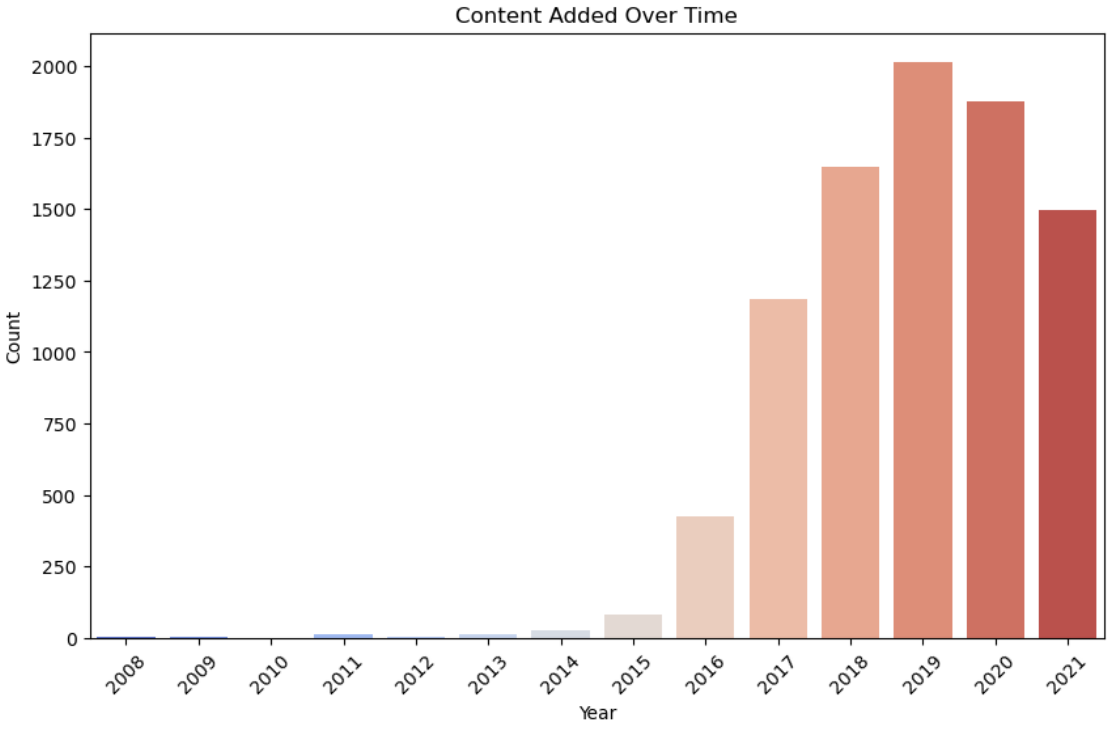
[24]:
```
# Most Commmon Genres
# Split the 'listed_in' column and count genre
data['genres'] = data['listed_in'].apply(lambda x: x.split(','))
all_genres = sum(data['genres'],[])
genre_counts = pd.Series(all_genres).value_counts().head(10)

# Plot the most common genre
plt.figure(figsize=(8,6))
sns.barplot(x=genre_counts.values,y=genre_counts.index,palette='Set3')
plt.title('Most Common Genres on Netflix')
plt.xlabel('Count')
plt.ylabel('Genre')
plt.show()
```

```
C:\Users\aa\AppData\Local\Temp\ipykernel_8608\3669360106.py:9: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same
effect.

  sns.barplot(x=genre_counts.values,y=genre_counts.index,palette='Set3')
```

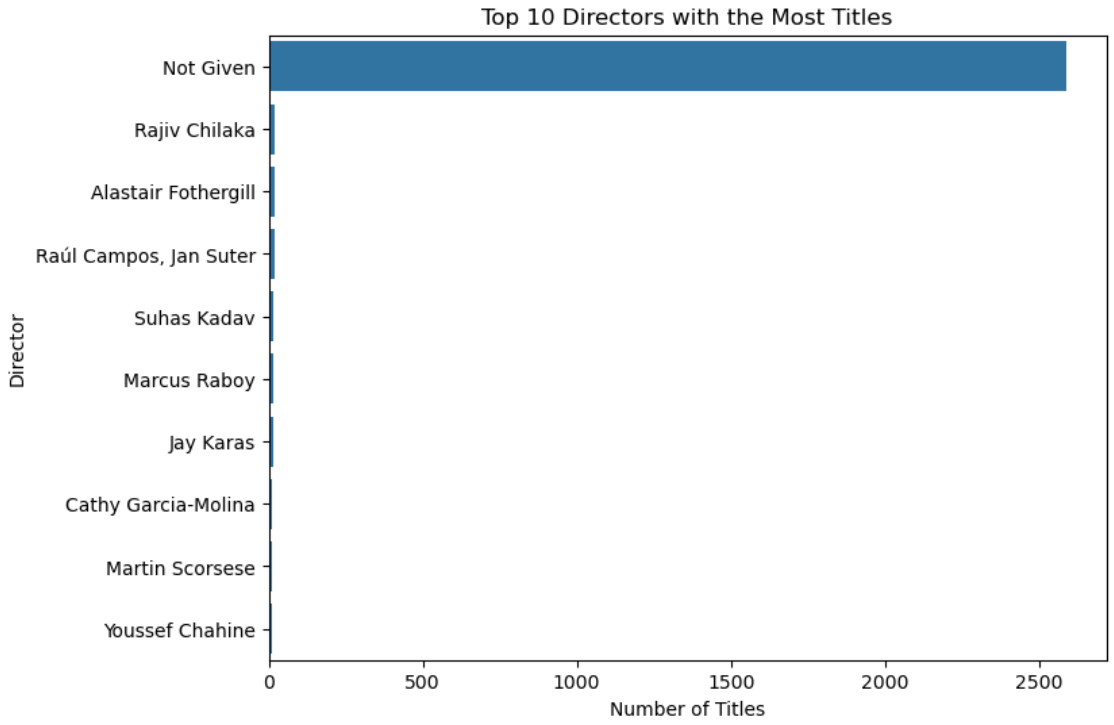


[18]:
```
# Content Added Over Time
# Extract Year and Month from 'date_added'
data['year_added'] = data['date_added'].dt.year
data['month_added'] = data['date_added'].dt.month

# Plot content added over the years
plt.figure(figsize=(10,6))
sns.countplot(x='year_added',data=data,palette='coolwarm')
plt.title('Content Added Over Time')
plt.xlabel('Year')
plt.ylabel('Count')
plt.xticks(rotation=45)
plt.show()
```

```
C:\Users\aa\AppData\Local\Temp\ipykernel_8232\1534635845.py:8: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.countplot(x='year_added',data=data,palette='coolwarm')
```



[20]:
```
# Top 10 Directions with the Most titles
# Counts Titles by Director
top_directors = data['director'].value_counts().head(10)

# Plot the Directors
plt.figure(figsize=(8,6))
sns.barplot(x=top_directors.values,y=top_directors.index)
plt.title('Top 10 Directors with the Most Titles')
plt.xlabel('Number of Titles')
plt.ylabel('Director')
plt.show()
```



[22]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8790 entries, 0 to 8789
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   show_id       8790 non-null   object
 1   type          8790 non-null   object
 2   title         8790 non-null   object
 3   director      8790 non-null   object
 4   country       8790 non-null   object
```

```
 5   date_added    8790 non-null   datetime64[ns]
 6   release_year  8790 non-null   int64
 7   rating        8790 non-null   object
 8   duration      8790 non-null   object
 9   listed_in     8790 non-null   object
 10  year_added    8790 non-null   int32
 11  month_added   8790 non-null   int32
dtypes: datetime64[ns](1), int32(2), int64(1), object(8)
memory usage: 755.5+ KB
```

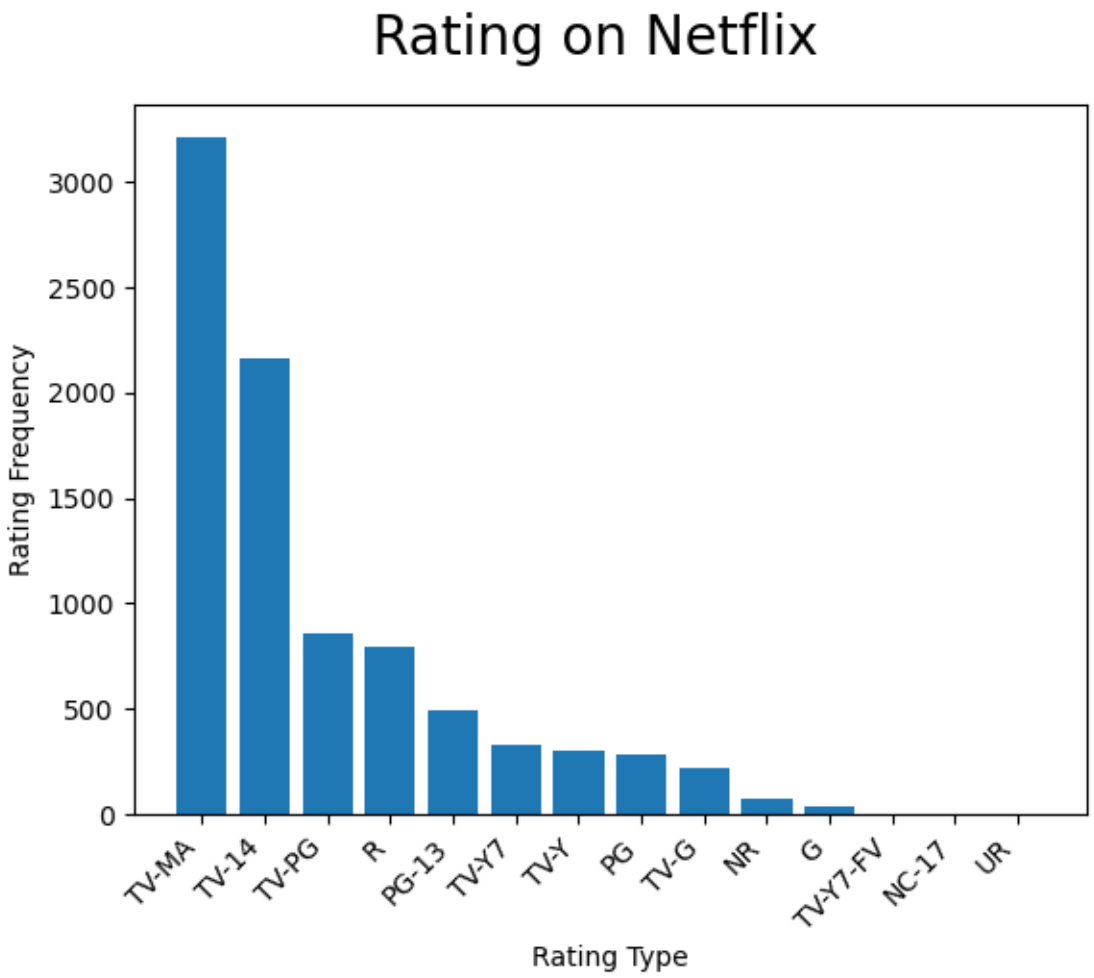
[24]:
```
data.shape
```

[24]:
```
(8790, 12)
```

[26]:
```
# Visual reprentation of rating frequenncy of movie and Tv show
data['rating'].value_counts()
```

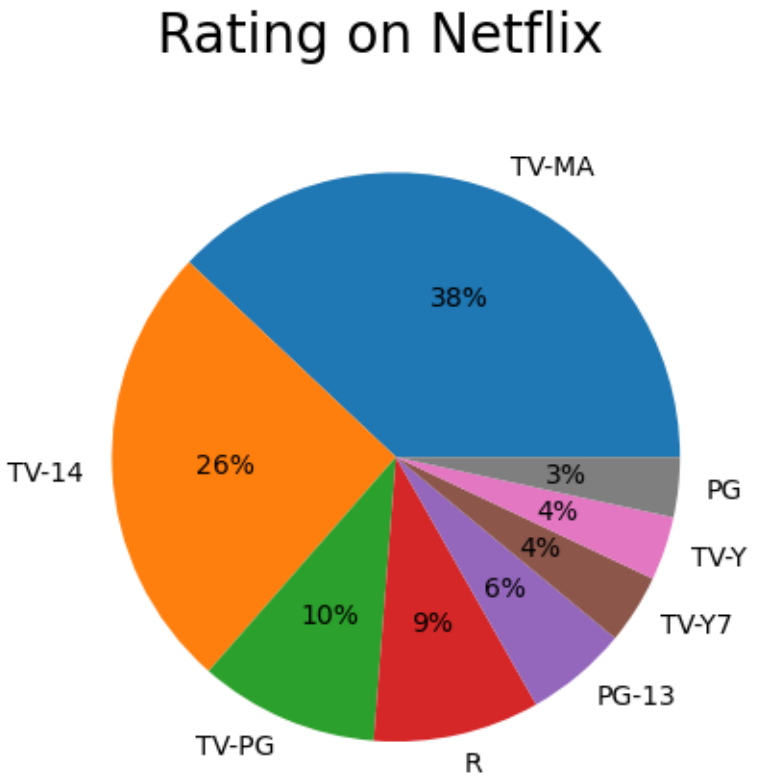
[26]:
```
rating
TV-MA       3205
TV-14       2157
TV-PG        861
R            799
PG-13        490
TV-Y7        333
TV-Y         306
PG           287
TV-G         220
NR            79
G             41
TV-Y7-FV       6
NC-17          3
UR             3
Name: count, dtype: int64
```

[28]:
```
ratings = data['rating'].value_counts().reset_index().
  ↪sort_values(by='count',ascending=False)
plt.bar(ratings['rating'],ratings['count'])
plt.xticks(rotation=45,ha='right')
plt.xlabel('Rating Type')
plt.ylabel('Rating Frequency')
plt.suptitle('Rating on Netflix',fontsize=20)
plt.show()
```



[30]:
```
plt.pie(ratings['count'][:8],labels=ratings['rating'][:8],autopct='%.0f%%')
plt.suptitle('Rating on Netflix',fontsize=20)
```

[30]: Text(0.5, 0.98, 'Rating on Netflix')

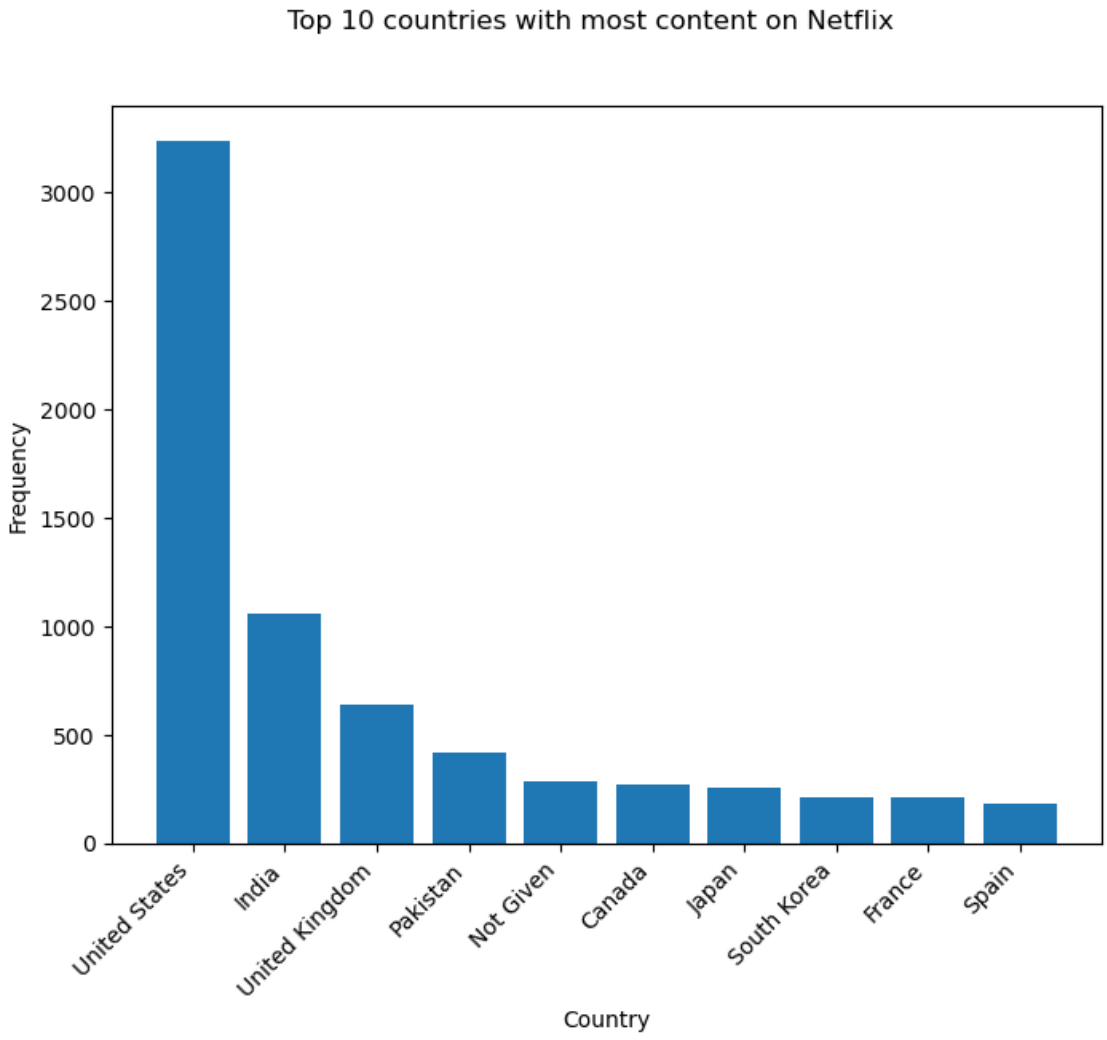


[32]:
```
data['country'].value_counts()
```

[32]:
```
country
United States     3240
India             1057
United Kingdom     638
Pakistan           421
Not Given          287
                  ...
Iran                 1
West Germany         1
Greece               1
Zimbabwe             1
Soviet Union         1
Name: count, Length: 86, dtype: int64
```

[34]:
```
# Top 10 Country with most content on Netflix
top_ten_countries = data['country'].value_counts().reset_index().
 ↪sort_values(by='count',ascending=False)[:10]

# Plot
```

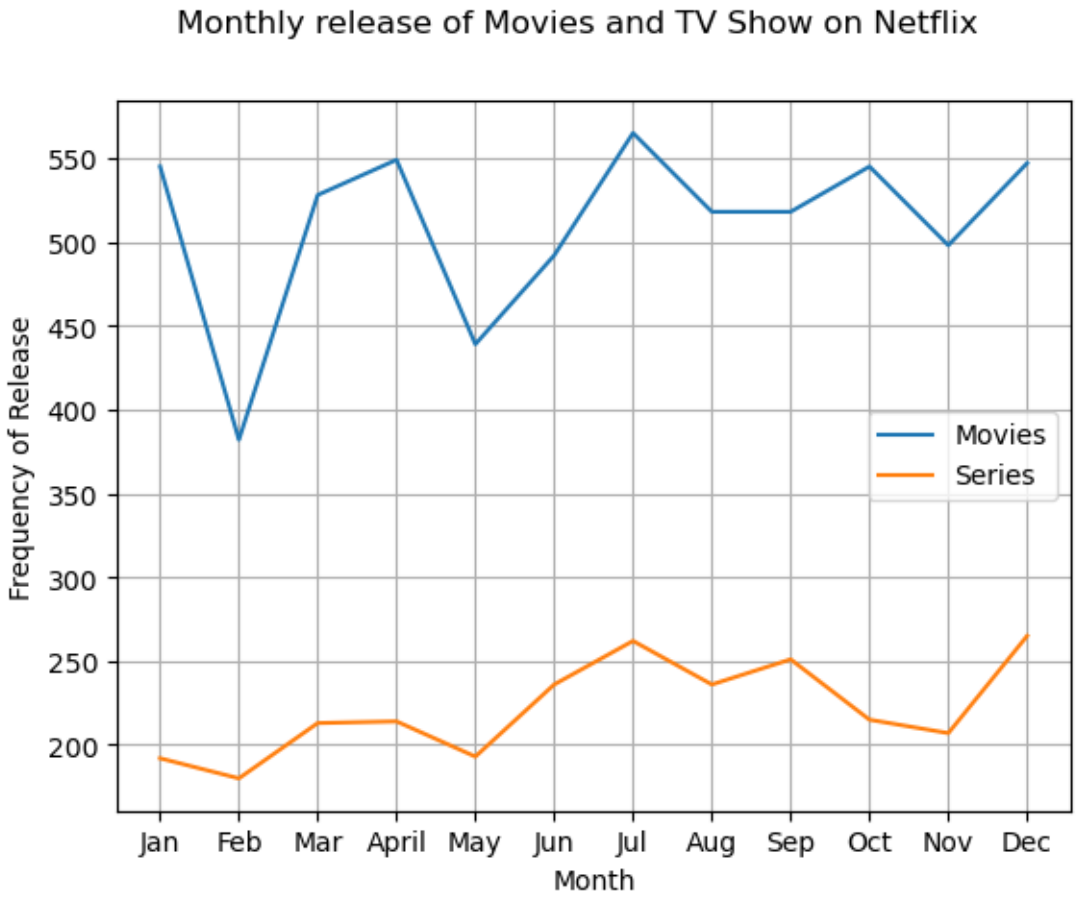
```
plt.figure(figsize=(8,6))
plt.bar(top_ten_countries['country'],top_ten_countries['count'])
plt.xticks(rotation=45,ha='right')
plt.xlabel('Country')
plt.ylabel('Frequency')
plt.suptitle('Top 10 countries with most content on Netflix')
plt.show()
```



[36]:
```
data['year'] = data['date_added'].dt.year
data['month'] = data['date_added'].dt.month
data['day'] = data['date_added'].dt.day
```

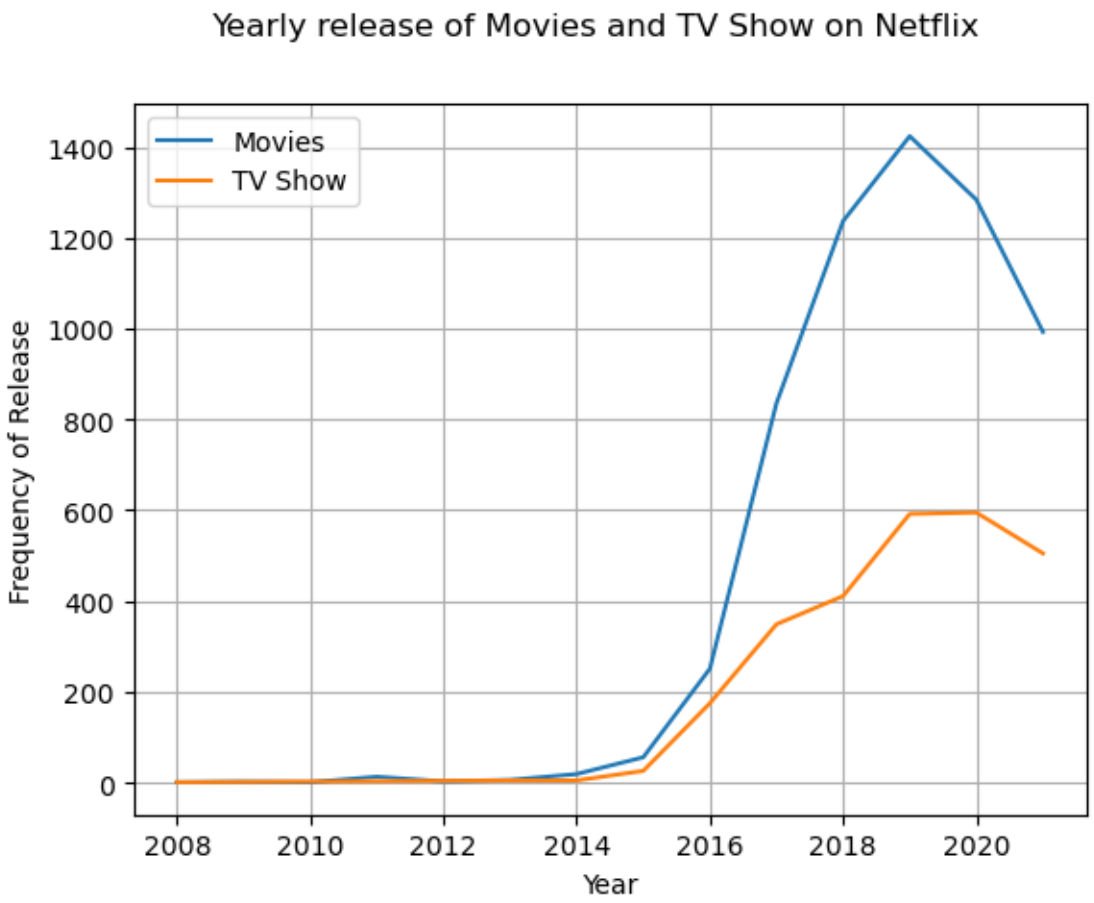
[38]:
```
# Monthly releases of Movies and TV Show on Netflix
monthly_movie_release = data[data['type']=='Movie']['month'].value_counts().
  ↪sort_index()
```

```
monthly_series_release = data[data['type']=='TV Show']['month'].value_counts().
  ↪sort_index()
plt.plot(monthly_movie_release.index,monthly_movie_release.
  ↪values,label='Movies')
plt.plot(monthly_series_release.index,monthly_series_release.
  ↪values,label='Series')
plt.xlabel('Month')
plt.ylabel('Frequency of Release')
plt.
  ↪xticks(range(1,13),['Jan','Feb','Mar','April','May','Jun','Jul','Aug','Sep','Oct','Nov','De
plt.legend()
plt.grid(True)
plt.suptitle('Monthly release of Movies and TV Show on Netflix')
plt.show()
```



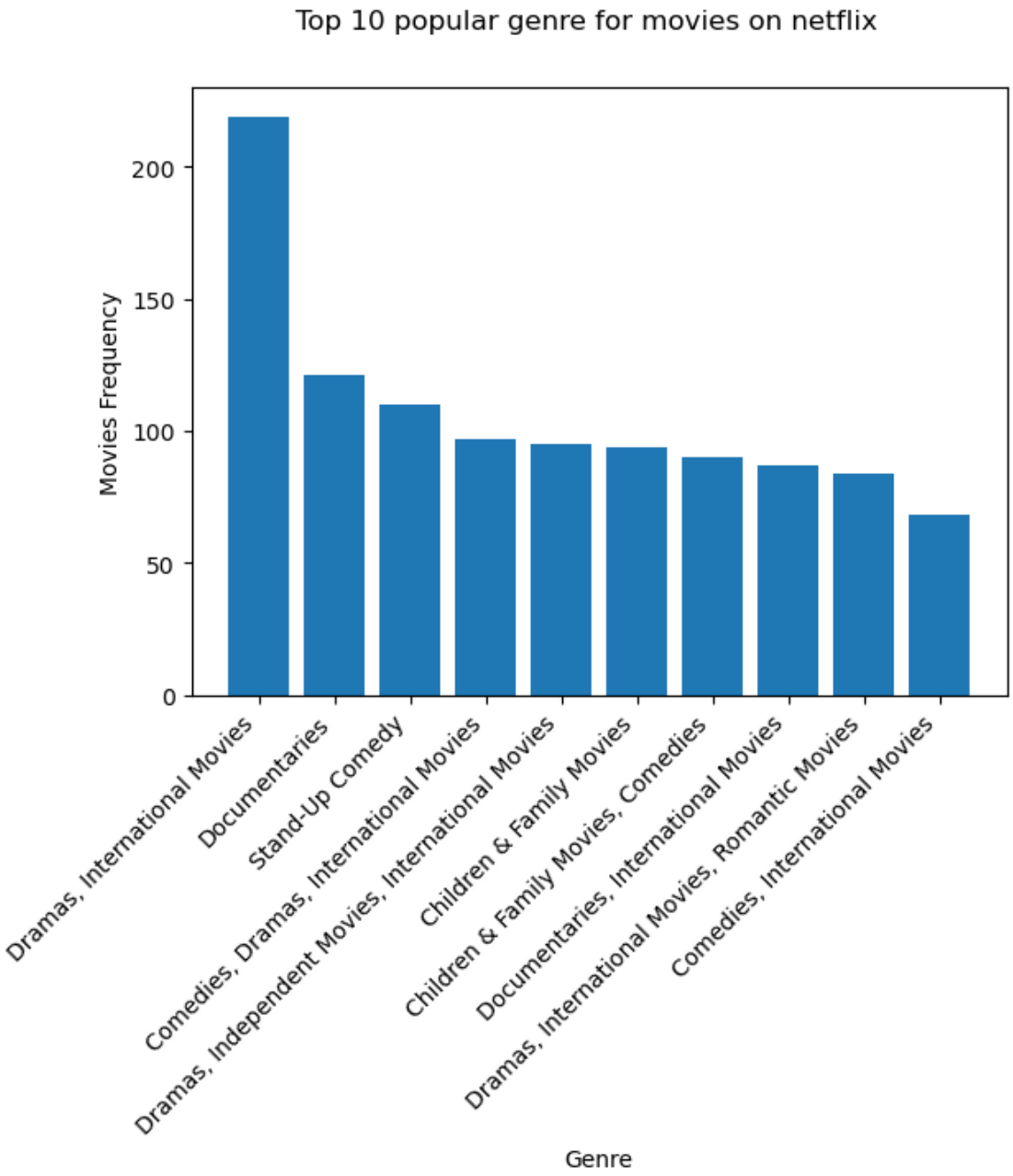
[40]:
```
# Yearly releases of Movies and TV Show on Netflix
yearly_movie_release = data[data['type']=='Movie']['year'].value_counts().
  ↪sort_index()
```

```
yearly_series_release = data[data['type']=='TV Show']['year'].value_counts().
  ↪sort_index()
plt.plot(yearly_movie_release.index,yearly_movie_release.values,label='Movies')
plt.plot(yearly_series_release.index,yearly_series_release.values,label='TV␣
  ↪Show')
plt.xlabel('Year')
plt.ylabel('Frequency of Release')
plt.legend()
plt.grid(True)
plt.suptitle('Yearly release of Movies and TV Show on Netflix')
plt.show()
```



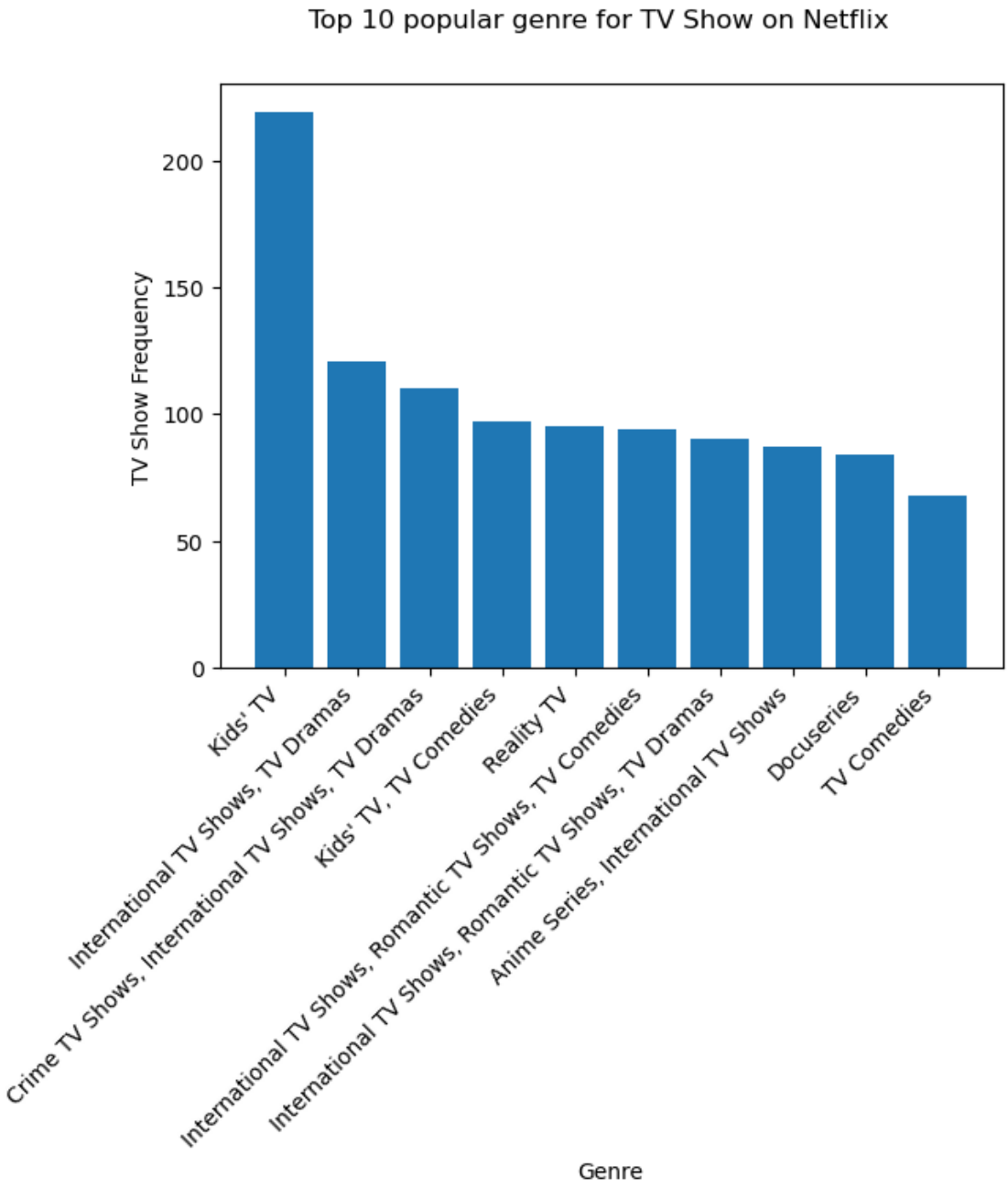
[42]:
```
# Top 10 popular movie genre
popular_movie_genre = data[data['type']=='Movie'].groupby('listed_in').size().
  ↪sort_values(ascending=False)[:10]
popular_series_genre = data[data['type']=='TV Show'].groupby('listed_in').
  ↪size().sort_values(ascending=False)[:10]
plt.bar(popular_movie_genre.index,popular_series_genre.values)
```

```
plt.xticks(rotation=45,ha='right')
plt.xlabel('Genre')
plt.ylabel('Movies Frequency')
plt.suptitle('Top 10 popular genre for movies on netflix')
plt.show()
```

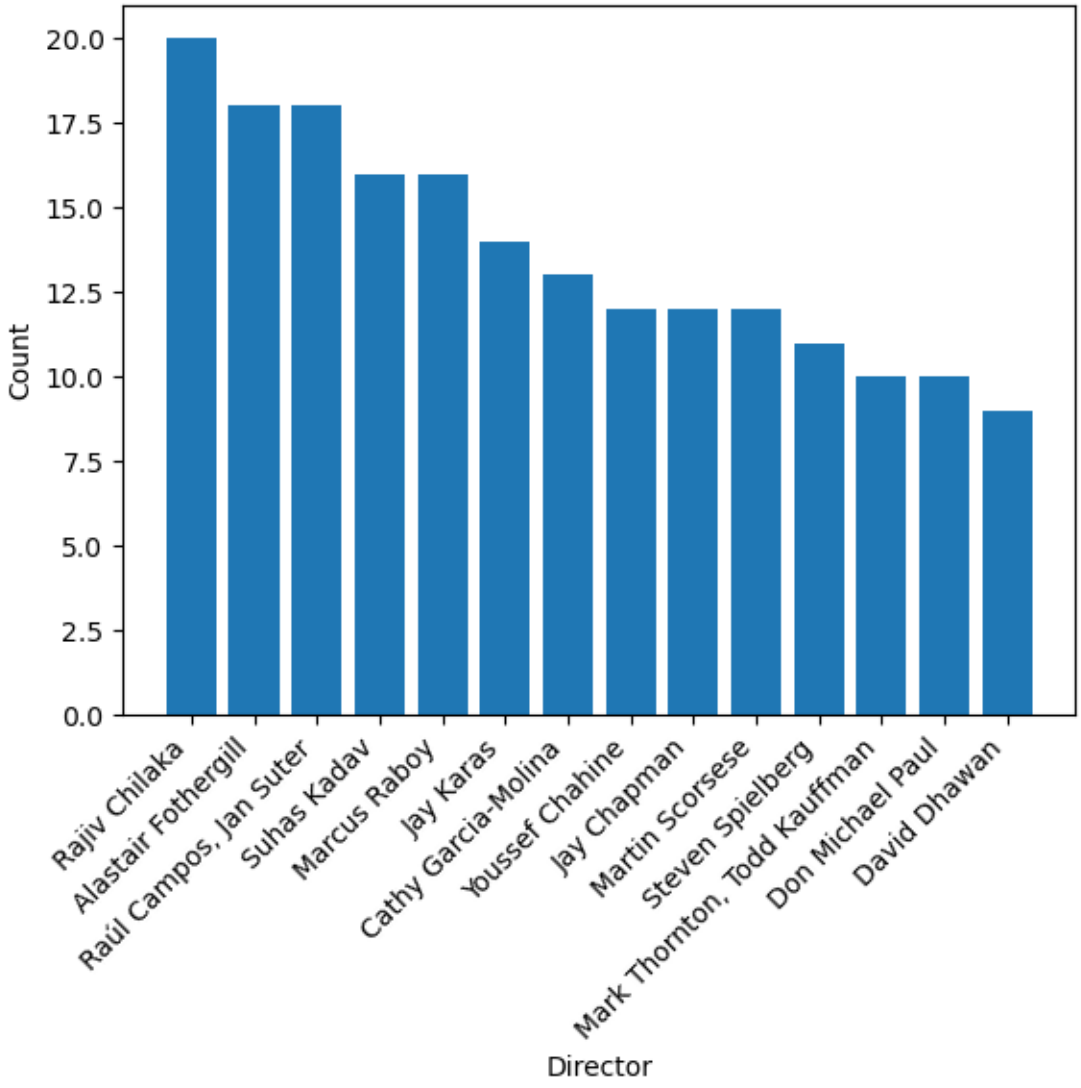


[44]:
```
# Top 10 TV Show Genre
plt.bar(popular_series_genre.index,popular_series_genre.values)
```

```
plt.xticks(rotation=45,ha='right')
plt.xlabel('Genre')
plt.ylabel('TV Show Frequency')
plt.suptitle('Top 10 popular genre for TV Show on Netflix')
plt.show()
```



[46]:
```
# Top 15 Directors across Netflix with high frequency of movies and show
director = data['director'].value_counts().reset_index().
 ↪sort_values(by='count',ascending=False)[1:15]
plt.bar(director['director'],director['count'])
plt.xlabel('Director')
plt.ylabel('Count')
plt.xticks(rotation=45,ha='right')
plt.show()
```



[48]:
```
data.info
```

[48]:
```
<bound method DataFrame.info of      show_id     type
title         director  \
```

```
0         s1    Movie              Dick Johnson Is Dead  Kirsten Johnson
1         s3  TV Show                         Ganglands  Julien Leclercq
2         s6  TV Show                     Midnight Mass    Mike Flanagan
3        s14    Movie  Confessions of an Invisible Girl    Bruno Garotti
4         s8    Movie                           Sankofa     Haile Gerima
…       …       …                                 …               …
8785   s8797  TV Show                       Yunus Emre        Not Given
8786   s8798  TV Show                        Zak Storm        Not Given
8787   s8801  TV Show              Zindagi Gulzar Hai        Not Given
8788   s8784  TV Show                            Yoko        Not Given
8789   s8786  TV Show                             YOM        Not Given

            country date_added  release_year rating   duration  \
0     United States 2021-09-25          2020  PG-13     90 min
1            France 2021-09-24          2021  TV-MA   1 Season
2     United States 2021-09-24          2021  TV-MA   1 Season
3            Brazil 2021-09-22          2021  TV-PG     91 min
4     United States 2021-09-24          1993  TV-MA    125 min
…              …        …           …     …        …
8785         Turkey 2017-01-17          2016  TV-PG  2 Seasons
8786  United States 2018-09-13          2016  TV-Y7  3 Seasons
8787       Pakistan 2016-12-15          2012  TV-PG   1 Season
8788       Pakistan 2018-06-23          2016   TV-Y   1 Season
8789       Pakistan 2018-06-07          2016  TV-Y7   1 Season

                                              listed_in  year_added  \
0                                         Documentaries        2021
1     Crime TV Shows, International TV Shows, TV Act…        2021
2                   TV Dramas, TV Horror, TV Mysteries        2021
3                   Children & Family Movies, Comedies        2021
4      Dramas, Independent Movies, International Movies        2021
…                                                   …         …
8785                 International TV Shows, TV Dramas        2017
8786                                         Kids' TV        2018
8787  International TV Shows, Romantic TV Shows, TV …        2016
8788                                         Kids' TV        2018
8789                                         Kids' TV        2018

      month_added  year  month  day
0               9  2021      9   25
1               9  2021      9   24
2               9  2021      9   24
3               9  2021      9   22
4               9  2021      9   24
…           …    …     …  …
8785            1  2017      1   17
8786            9  2018      9   13
```

```
8787           12  2016     12   15
8788            6  2018      6   23
8789            6  2018      6    7

[8790 rows x 15 columns]>
```

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: