master-1

May 29, 2024

[31]:
```
# Import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from wordcloud import WordCloud

# Load the dataset
netflix_df = pd.read_csv('netflix.csv')

# Display the first few rows of the dataset
print(netflix_df.head(10))
```

```
  show_id     type                              title  \
0      s1    Movie               Dick Johnson Is Dead
1      s2  TV Show                       Blood & Water
2      s3  TV Show                           Ganglands
3      s4  TV Show               Jailbirds New Orleans
4      s5  TV Show                        Kota Factory
5      s6  TV Show                       Midnight Mass
6      s7    Movie  My Little Pony: A New Generation
7      s8    Movie                             Sankofa
8      s9  TV Show      The Great British Baking Show
9     s10    Movie                          The Starling

                      director  \
0                Kirsten Johnson
1                            NaN
2                Julien Leclercq
3                            NaN
4                            NaN
5                  Mike Flanagan
6  Robert Cullen, José Luis Ucha
7                    Haile Gerima
8                Andy Devonshire
9                 Theodore Melfi

                                                cast  \
```

```
0                                                   NaN
1  Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban…
2  Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi…
3                                                   NaN
4  Mayur More, Jitendra Kumar, Ranjan Raj, Alam K…
5  Kate Siegel, Zach Gilford, Hamish Linklater, H…
6  Vanessa Hudgens, Kimiko Glenn, James Marsden, …
7  Kofi Ghanaba, Oyafunmike Ogunlano, Alexandra D…
8  Mel Giedroyc, Sue Perkins, Mary Berry, Paul Ho…
9  Melissa McCarthy, Chris O'Dowd, Kevin Kline, T…

                                          country          date_added  \
0                                   United States  September 25, 2021
1                                    South Africa  September 24, 2021
2                                             NaN  September 24, 2021
3                                             NaN  September 24, 2021
4                                           India  September 24, 2021
5                                             NaN  September 24, 2021
6                                             NaN  September 24, 2021
7  United States, Ghana, Burkina Faso, United Kin…  September 24, 2021
8                                  United Kingdom  September 24, 2021
9                                   United States  September 24, 2021

   release_year rating   duration  \
0          2020  PG-13     90 min
1          2021  TV-MA  2 Seasons
2          2021  TV-MA   1 Season
3          2021  TV-MA   1 Season
4          2021  TV-MA  2 Seasons
5          2021  TV-MA   1 Season
6          2021     PG     91 min
7          1993  TV-MA    125 min
8          2021  TV-14  9 Seasons
9          2021  PG-13    104 min

                                       listed_in  \
0                                  Documentaries
1    International TV Shows, TV Dramas, TV Mysteries
2  Crime TV Shows, International TV Shows, TV Act…
3                              Docuseries, Reality TV
4  International TV Shows, Romantic TV Shows, TV …
5                 TV Dramas, TV Horror, TV Mysteries
6                           Children & Family Movies
7   Dramas, Independent Movies, International Movies
8                        British TV Shows, Reality TV
9                                Comedies, Dramas

                                        description
```

```
0  As her father nears the end of his life, filmm…
1  After crossing paths at a party, a Cape Town t…
2  To protect his family from a powerful drug lor…
3  Feuds, flirtations and toilet talk go down amo…
4  In a city of coaching centers known to train I…
5  The arrival of a charismatic young priest brin…
6  Equestria's divided. But a bright-eyed hero be…
7  On a photo shoot in Ghana, an American model s…
8  A talented batch of amateur bakers face off in…
9  A woman adjusting to life after a loss contend…
```

[ ]:
```
# Problem Statement
"""
This analysis aims to understand the distribution and characteristics of␣
 ↪Netflix's content library.
Key objectives include:
1. Identifying trends in movie and TV show production over time.
2. Analyzing the distribution of content ratings.
3. Determining the most productive countries in terms of content creation.
4. Finding the optimal times for releasing new content.
5. Exploring the most prolific actors and directors.
"""
```

[5]:
```
# Basic Data Information
print(netflix_df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8797 non-null   object
 7   release_year  8807 non-null   int64
 8   rating        8803 non-null   object
 9   duration      8804 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
None
```

[7]:
```
# Convert categorical columns to 'category' type
categorical_columns = ['type', 'rating', 'country', 'listed_in', 'director',␣
 ↪'cast']
for column in categorical_columns:
    netflix_df[column] = netflix_df[column].astype('category')

# Statistical Summary
print(netflix_df.describe(include='all'))
```

```
       show_id   type                  title       director  \
count     8807   8807                   8807           6173
unique    8807      2                   8807           4528
top         s1  Movie  Dick Johnson Is Dead  Rajiv Chilaka
freq         1   6131                      1             19
mean       NaN    NaN                    NaN            NaN
std        NaN    NaN                    NaN            NaN
min        NaN    NaN                    NaN            NaN
25%        NaN    NaN                    NaN            NaN
50%        NaN    NaN                    NaN            NaN
75%        NaN    NaN                    NaN            NaN
max        NaN    NaN                    NaN            NaN

                       cast        country       date_added  release_year  \
count                  7982           7976             8797   8807.000000
unique                 7692            748             1767           NaN
top     David Attenborough  United States  January 1, 2020           NaN
freq                     19           2818              109           NaN
mean                    NaN            NaN              NaN   2014.180198
std                     NaN            NaN              NaN      8.819312
min                     NaN            NaN              NaN   1925.000000
25%                     NaN            NaN              NaN   2013.000000
50%                     NaN            NaN              NaN   2017.000000
75%                     NaN            NaN              NaN   2019.000000
max                     NaN            NaN              NaN   2021.000000

       rating  duration                     listed_in  \
count    8803      8804                          8807
unique     17       220                           514
top     TV-MA  1 Season  Dramas, International Movies
freq     3207      1793                           362
mean      NaN       NaN                           NaN
std       NaN       NaN                           NaN
min       NaN       NaN                           NaN
25%       NaN       NaN                           NaN
50%       NaN       NaN                           NaN
75%       NaN       NaN                           NaN
max       NaN       NaN                           NaN
```

```
                                                description
count                                                  8807
unique                                                 8775
top     Paranormal activity at a lush, abandoned prope…
freq                                                      4
mean                                                    NaN
std                                                     NaN
min                                                     NaN
25%                                                     NaN
50%                                                     NaN
75%                                                     NaN
max                                                     NaN
```

[52]:
```
# 1. Un-nesting the columns
# Splitting columns with comma-separated values and creating multiple rows
def split_and_explode(df, column):
    df[column] = df[column].fillna('')
    df[column] = df[column].apply(lambda x: x.split(', ') if x else [])
    df = df.explode(column)
    return df

# Applying split_and_explode to 'cast', 'director', and 'country'
netflix_df = split_and_explode(netflix_df, 'cast')
netflix_df = split_and_explode(netflix_df, 'director')
netflix_df = split_and_explode(netflix_df, 'country')
```

[113]:
```
# Handling null values by converting to string type first
netflix_df['rating'] = netflix_df['rating'].astype(str).fillna('Unknown Rating')
netflix_df['cast'] = netflix_df['cast'].astype(str).fillna('Unknown Actor')
netflix_df['director'] = netflix_df['director'].astype(str).fillna('Unknown␣
 ↪Director')
netflix_df['country'] = netflix_df['country'].astype(str).fillna('Unknown␣
 ↪Country')
netflix_df['duration'] = netflix_df['duration'].astype(str).fillna('0')

# Converting 'date_added' to datetime
netflix_df['date_added'] = pd.to_datetime(netflix_df['date_added'], format='%B␣
 ↪%d, %Y')

netflix_df['release_year'] = netflix_df['release_year'].fillna(0).astype(int)

# Convert categorical columns to 'category' type
categorical_columns = ['type', 'rating', 'country', 'listed_in', 'director',␣
 ↪'cast']
for column in categorical_columns:
    netflix_df[column] = netflix_df[column].astype('category')
```

```
# Statistical Summary
print(netflix_df.describe(include='all'))
```

```
       show_id   type     title director   cast        country  \
count    89313  89313     89313    89313  89313          89313
unique    8797      2      8797     4994  36404            127
top       s7516  Movie  Movie 43      nan    nan  United States
freq       468  65346       468    21868   1190          30435
mean       NaN    NaN       NaN      NaN    NaN            NaN
min        NaN    NaN       NaN      NaN    NaN            NaN
25%        NaN    NaN       NaN      NaN    NaN            NaN
50%        NaN    NaN       NaN      NaN    NaN            NaN
75%        NaN    NaN       NaN      NaN    NaN            NaN
max        NaN    NaN       NaN      NaN    NaN            NaN
std        NaN    NaN       NaN      NaN    NaN            NaN

                         date_added  release_year rating  duration  \
count                         89313  89313.000000  89313     89313
unique                          NaN           NaN     18       221
top                             NaN           NaN  TV-MA  1 Season
freq                            NaN           NaN  29846     14624
mean    2019-06-16 22:55:50.777602304   2013.453394    NaN       NaN
min               2008-01-01 00:00:00   1925.000000    NaN       NaN
25%               2018-06-08 00:00:00   2012.000000    NaN       NaN
50%               2019-09-13 00:00:00   2016.000000    NaN       NaN
75%               2020-09-18 00:00:00   2019.000000    NaN       NaN
max               2021-09-25 00:00:00   2021.000000    NaN       NaN
std                             NaN      8.786106    NaN       NaN

                         listed_in  \
count                        89313
unique                         513
top     Dramas, International Movies
freq                          4255
mean                           NaN
min                            NaN
25%                            NaN
50%                            NaN
75%                            NaN
max                            NaN
std                            NaN

                                                description  days_to_netflix
count                                                 89313     89313.000000
unique                                                 8765              NaN
top     An eye-popping cast stars in this sketch-comed…                 NaN
```

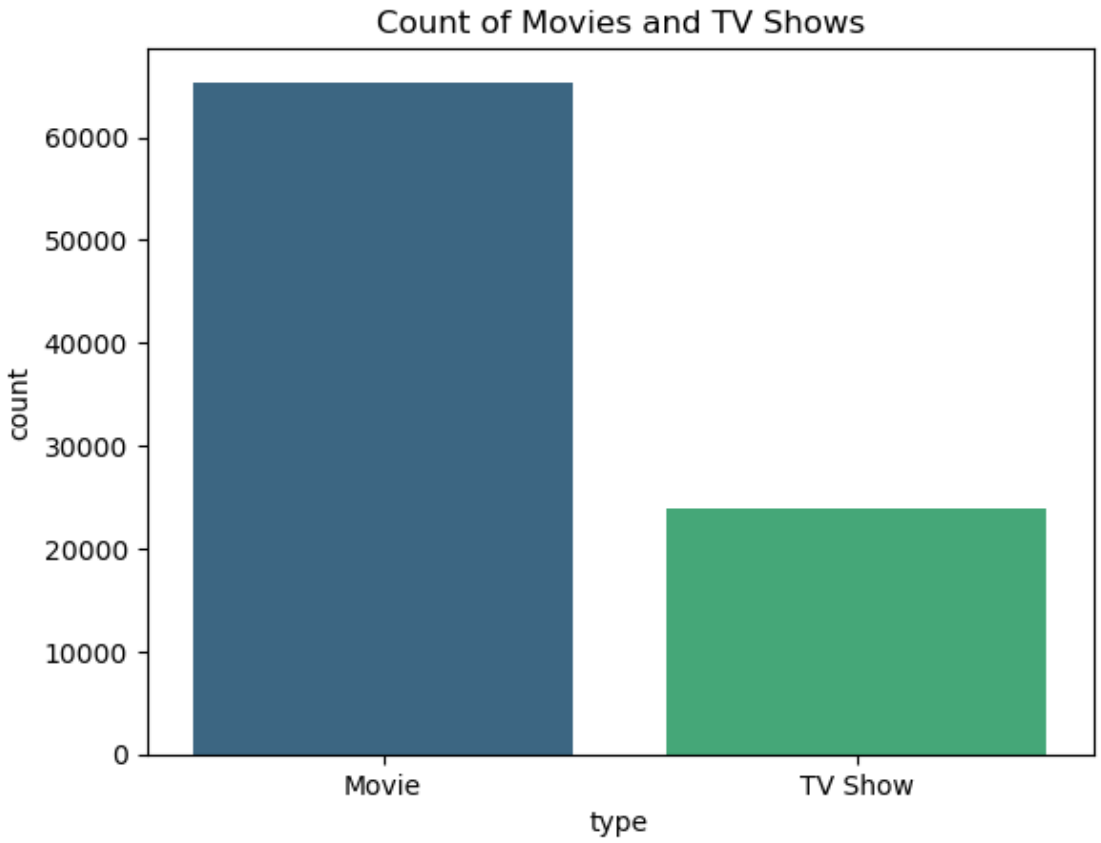
```
freq                                                  468              NaN
mean                                                  NaN      2192.730543
min                                                   NaN     -1006.000000
25%                                                   NaN       303.000000
50%                                                   NaN       818.000000
75%                                                   NaN      2875.000000
max                                                   NaN     34331.000000
std                                                   NaN      3224.451854
```

[111]:
```
# Handling null values by converting to string type first
netflix_df['rating'] = netflix_df['rating'].astype(str).fillna('Unknown Rating')
netflix_df['cast'] = netflix_df['cast'].astype(str).fillna('Unknown Actor')
netflix_df['director'] = netflix_df['director'].astype(str).fillna('Unknown␣
 ↪Director')
netflix_df['country'] = netflix_df['country'].astype(str).fillna('Unknown␣
 ↪Country')
netflix_df['duration'] = netflix_df['duration'].astype(str).fillna('0')

# Converting 'date_added' to datetime
netflix_df['date_added'] = pd.to_datetime(netflix_df['date_added'], format='%B␣
 ↪%d, %Y')

netflix_df['release_year'] = netflix_df['release_year'].fillna(0).astype(int)

# Applying split_and_explode to 'cast', 'director', and 'country'
netflix_df = split_and_explode(netflix_df, 'cast')
netflix_df = split_and_explode(netflix_df, 'director')
netflix_df = split_and_explode(netflix_df, 'country')
netflix_df = netflix_df.dropna()


# Display the first few rows of the dataset to verify the preprocessing
print(netflix_df.head())
```

```
  show_id     type                  title         director            cast  \
0      s1    Movie  Dick Johnson Is Dead  Kirsten Johnson             nan
1      s2  TV Show         Blood & Water              nan      Ama Qamata
1      s2  TV Show         Blood & Water              nan     Khosi Ngema
1      s2  TV Show         Blood & Water              nan   Gail Mabalane
1      s2  TV Show         Blood & Water              nan  Thabang Molaba

         country date_added  release_year rating   duration  \
0  United States 2021-09-25          2020  PG-13     90 min
1   South Africa 2021-09-24          2021  TV-MA  2 Seasons
1   South Africa 2021-09-24          2021  TV-MA  2 Seasons
1   South Africa 2021-09-24          2021  TV-MA  2 Seasons
1   South Africa 2021-09-24          2021  TV-MA  2 Seasons
```

```
                                        listed_in  \
0                                   Documentaries
1  International TV Shows, TV Dramas, TV Mysteries
1  International TV Shows, TV Dramas, TV Mysteries
1  International TV Shows, TV Dramas, TV Mysteries
1  International TV Shows, TV Dramas, TV Mysteries

                                         description  days_to_netflix
0  As her father nears the end of his life, filmm…            633.0
1  After crossing paths at a party, a Cape Town t…            266.0
1  After crossing paths at a party, a Cape Town t…            266.0
1  After crossing paths at a party, a Cape Town t…            266.0
1  After crossing paths at a party, a Cape Town t…            266.0
```

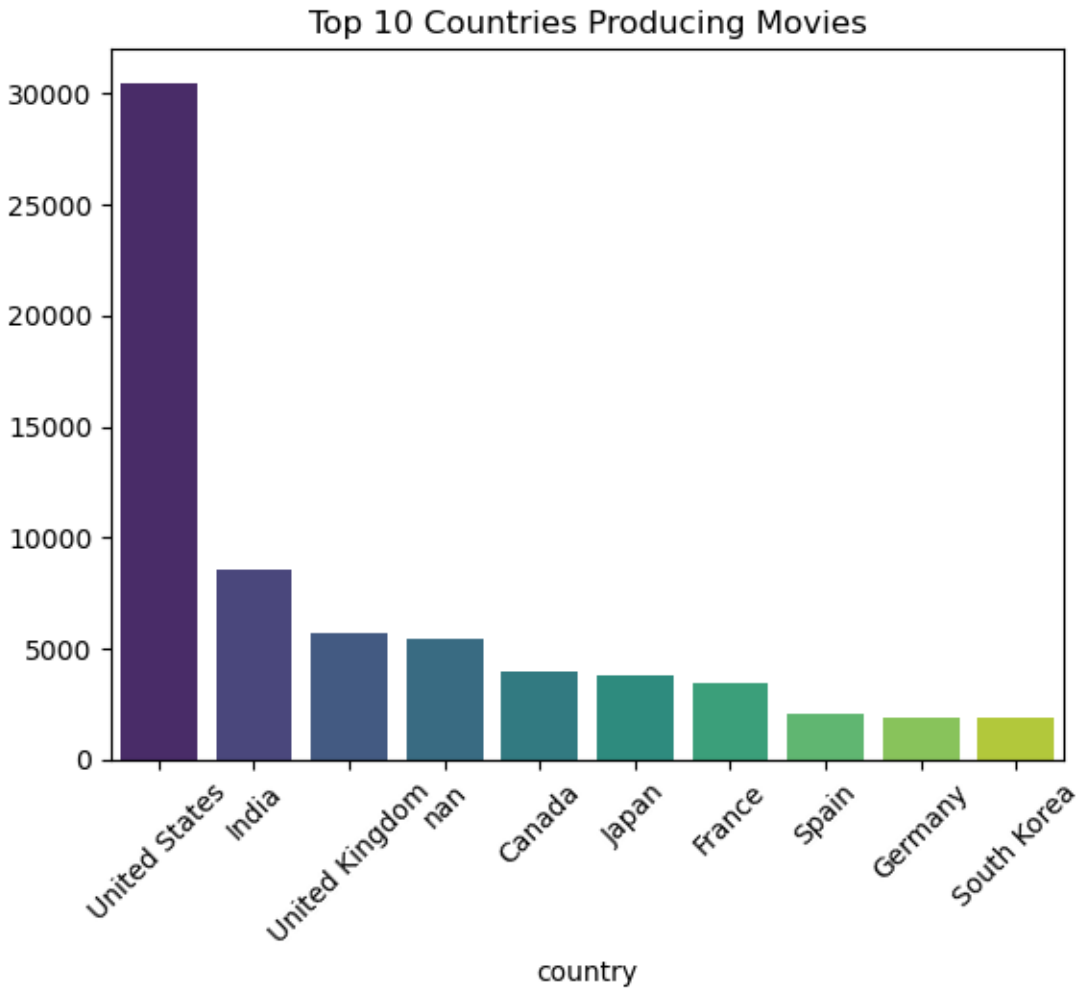
[115]:
```
# Graphical Analysis: Count plots for each categorical variable
sns.countplot(data=netflix_df, x='type', palette='viridis')
plt.title('Count of Movies and TV Shows')
plt.show()
```



[117]:
```
# Top 10 countries producing movies
movies_by_country = netflix_df['country'].value_counts().head(10)
print(movies_by_country)
```

```
country
United States     30435
India              8537
United Kingdom     5704
nan                5432
Canada             3946
Japan              3740
France             3489
Spain              2033
Germany            1927
South Korea        1861
Name: count, dtype: int64
```

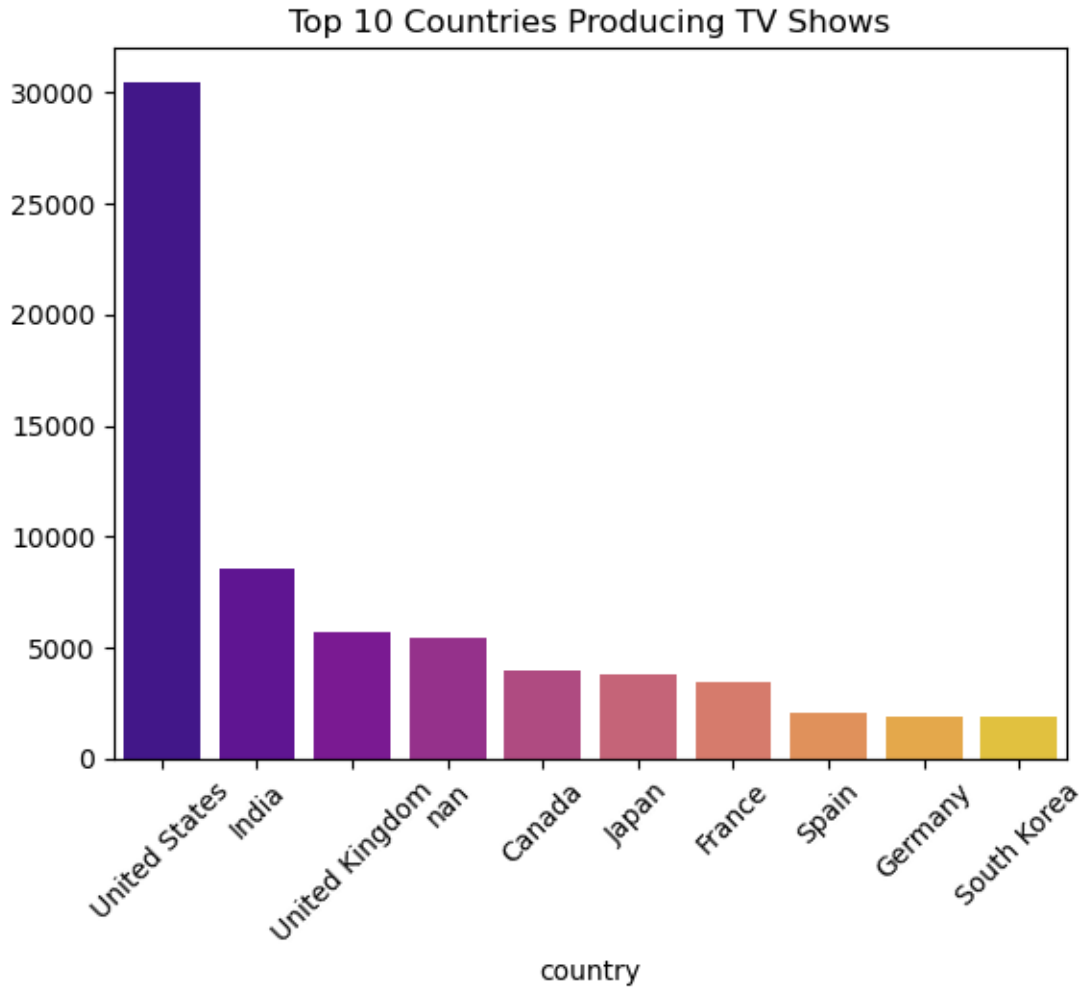
[68]:
```
--# Top 10 countries producing movies in plot
sns.barplot(x=movies_by_country.index, y=movies_by_country.values,␣
 ↪palette='viridis')
plt.title('Top 10 Countries Producing Movies')
plt.xticks(rotation=45)
plt.show()
```



[119]:
```
# Top 10 countries producing TV shows
tv_shows_by_country = netflix_df['country'].value_counts().head(10)
print(tv_shows_by_country)
```

```
country
United States     30435
India              8537
United Kingdom     5704
nan                5432
Canada             3946
Japan              3740
France             3489
Spain              2033
Germany            1927
South Korea        1861
Name: count, dtype: int64
```

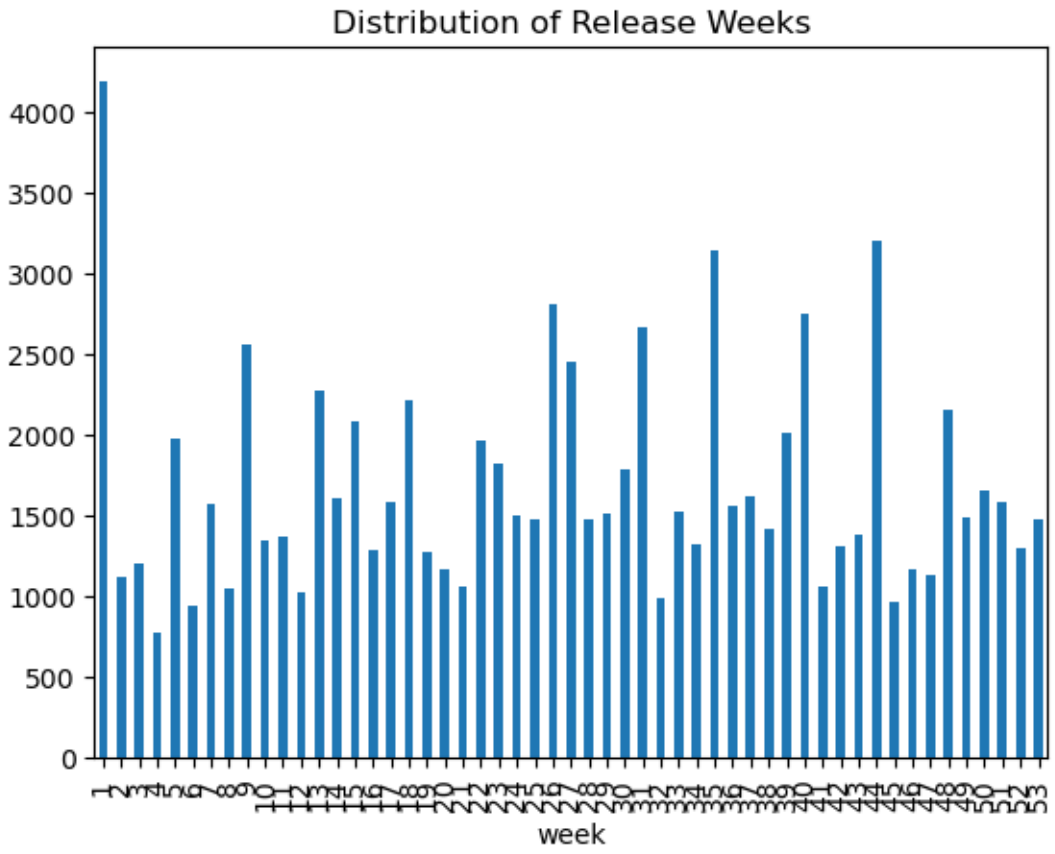
[72]:
```
--# Top 10 countries producing TV shows
sns.barplot(x=tv_shows_by_country.index, y=tv_shows_by_country.values,␣
 ↪palette='plasma')
plt.title('Top 10 Countries Producing TV Shows')
plt.xticks(rotation=45)
plt.show()
```

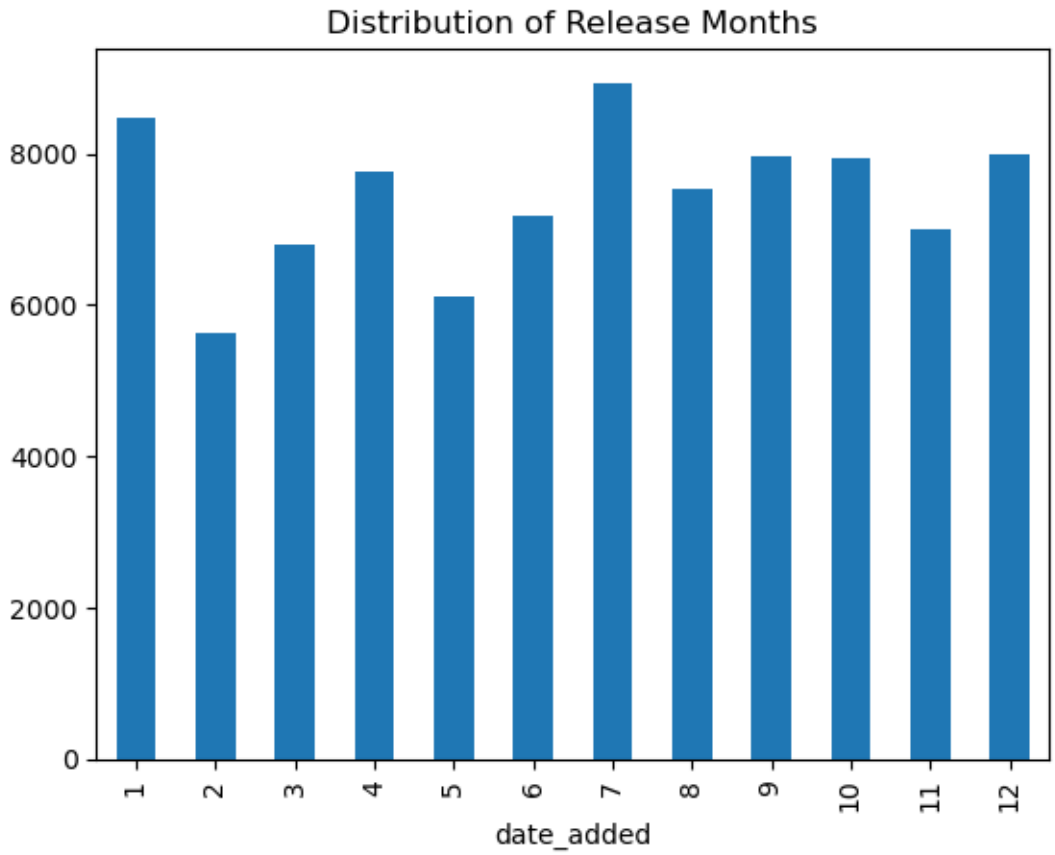


[ ]:
```

```

[123]:
```
# Plot the distribution of release weeks
netflix_df['date_added'].dt.isocalendar().week.value_counts().sort_index().
 ↪plot(kind='bar')
plt.title('Distribution of Release Weeks')
plt.show()

# Plot the distribution of release months
```

```
netflix_df['date_added'].dt.month.value_counts().sort_index().plot(kind='bar')
plt.title('Distribution of Release Months')
plt.show()

# Print a few rows of the 'date_added' column
print(netflix_df['date_added'].head())

# Find the best week for movies and TV shows (using isocalendar().week)
best_week_movies = netflix_df['date_added'].dt.isocalendar().week.
 ↪value_counts().argmax()
best_week_tv_shows = netflix_df['date_added'].dt.isocalendar().week.
 ↪value_counts().argmax()

# Find the best week for movies and TV shows (using isocalendar().weekofyear)
best_week_movies = netflix_df['date_added'].dt.isocalendar().apply(lambda x:␣
 ↪x[1]).value_counts().argmax()
best_week_tv_shows = netflix_df['date_added'].dt.isocalendar().apply(lambda x:␣
 ↪x[1]).value_counts().argmax()

print(f"Best week to release movies: Week {best_week_movies}")
print(f"Best week to release TV shows: Week {best_week_tv_shows}")
```






```
0   2021-09-25
1   2021-09-24
1   2021-09-24
1   2021-09-24
1   2021-09-24
Name: date_added, dtype: datetime64[ns]
Best week to release movies: Week 0
Best week to release TV shows: Week 0
```

[76]:
```
# Best month to release
best_month_movies = netflix_df['date_added'].value_counts().idxmax()
best_month_tv_shows = netflix_df['date_added'].value_counts().idxmax()
print(f"Best month to release movies: Month {best_month_movies}")
print(f"Best month to release TV shows: Month {best_month_tv_shows}")
```

```
Best month to release movies: Month 2020-01-01 00:00:00
Best month to release TV shows: Month 2020-01-01 00:00:00
```

[78]:
```
# Top 10 actors
top_actors = netflix_df['cast'].value_counts().head(10)
print(top_actors)
```

```
cast
nan                  1190
Alfred Molina          85
Liam Neeson            82
John Krasinski         67
Frank Langella         66
Salma Hayek            66
John Rhys-Davies       60
Tara Strong            54
James Franco           53
Quvenzhané Wallis      50
Name: count, dtype: int64
```

[80]:
```
# Top 10 directors
top_directors = netflix_df['director'].value_counts().head(10)
print(top_directors)
```

```
director
nan                    21937
Martin Scorsese          217
Steven Spielberg         205
Martin Campbell          154
Raja Gosnell             154
McG                      150
Youssef Chahine          150
Rajiv Chilaka            139
Don Michael Paul         132
Cathy Garcia-Molina      125
Name: count, dtype: int64
```

[82]:
```
# Word cloud for genres
genres = ' '.join(netflix_df['listed_in'].dropna().astype(str).values)
wordcloud = WordCloud(width=800, height=400, background_color='white').
 ↪generate(genres)
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title('Popular Genres')
plt.show()
```

[84]:
```
# Time to Netflix after release
netflix_df['days_to_netflix'] = (netflix_df['date_added'] - pd.
 ↪to_datetime(netflix_df['release_year'], format='%Y')).dt.days
mode_days_to_netflix = netflix_df['days_to_netflix'].mode()[0]
print(f"Mode of days to add to Netflix after release: {mode_days_to_netflix}␣
 ↪days")
```

```
Mode of days to add to Netflix after release: 2289.0 days
```

[ ]:

[ ]:

[ ]:
```
# Insights based on the analysis
"""
1. Range of Attributes:
   - Release year ranges from 1925 to 2021.
   - Duration of content varies significantly with many outliers.

2. Distribution:
   - Most content is rated 'TV-MA' and 'TV-14'.
   - The majority of content is produced in the United States, followed by␣
 ↪India and the United Kingdom.

3. Relationships:
   - Positive correlation between release year and date added, indicating newer␣
 ↪content is added more frequently.
```

```
   - TV shows tend to have shorter durations compared to movies.
"""

# Business Insights
"""
1. The United States, India, and the United Kingdom are the top producers of␣
 ↪content on Netflix. Investing in these regions can yield a high volume of␣
 ↪new content.
2. The most popular content ratings are 'TV-MA' and 'TV-14'. Tailoring content␣
 ↪to these ratings can attract more viewers.
3. The best time to release new TV shows is around Week 35 (August-September),␣
 ↪and movies in Week 1 (January), aligning with user engagement patterns.
4. Promoting popular genres identified through the word cloud analysis in␣
 ↪marketing campaigns can attract genre-specific audiences.
5. Utilizing insights on top actors and directors for targeted marketing and␣
 ↪content creation strategies.
"""

# Recommendations
"""
1. Increase collaborations with top content-producing countries (USA, India,␣
 ↪UK) to expand the content library.
2. Focus on producing and acquiring 'TV-MA' and 'TV-14' rated content to match␣
 ↪viewer preferences.
3. Schedule major content releases around Week 35 and Week 1 to maximize viewer␣
 ↪engagement and subscriptions.
4. Promote popular genres identified through the word cloud analysis in␣
 ↪marketing campaigns to attract genre-specific audiences.
5. Utilize insights on top actors and directors for targeted marketing and␣
 ↪content creation strategies.
"""
```