```
import pandas as pd
import numpy as np
data =
pd.read_csv('https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/000/940/original/netflix.csv')
```

Problem Statement

The challenge is to enhance Netflix's content strategy and business growth by leveraging data-driven insights. The specific areas of focus include tailoring content distribution by country, adjusting the content production strategy based on historical movie release trends, determining the optimal balance between TV shows and movies, identifying strategic launch times, collaborating with popular actors and directors, and adapting to recent trends in content preferences. The overarching problem is to formulate a comprehensive content strategy that aligns with user preferences and increase the success and expansion of the Netflix platform.

Data Exploration | Non-Graphical Analysis

Shape of data The dataset has a shape of (8807, 12), indicating that it contains 8807 rows and 12 columns.

```
data.shape
```

```
(8807, 12)
```

```
data.dtypes
```

```
show_id         object
type            object
title           object
director        object
cast            object
country         object
date_added      object
release_year     int64
rating          object
duration        object
listed_in       object
description     object
dtype: object
```

Categorical/Object Attributes:

show_id: Identifier for the show (object), type: Type of content (movie or TV show) (object), title: Title of the content (object), director: Director(s) of the content (object), cast: Cast members of the content (object), country: Country or countries where the content is available (object),

date_added: Date when the content was added to Netflix (object), rating: Content rating (object), duration: Duration of the content (object), listed_in: Categories or genres the content is listed under (object), description: Brief description of the content (object),

Numerical Attribute:

release_year: Year when the content was released (int64)

It's worth noting that some of the attributes like date_added and duration are currently of the object data type, and depending on the analysis, we can consider converting them to a more appropriate data type (e.g., datetime for date_added and numerical for duration).

```
data.isnull().sum()
```

```
show_id            0
type               0
title              0
director        2634
cast             825
country          831
date_added        10
release_year       0
rating             4
duration           3
listed_in          0
description        0
dtype: int64
```

Analysis of Missing Values:

Show ID, Type, Title, Listed In, and Description: **No missing values in these attributes.**

Director (2634 missing): The 'director' attribute has **2634** missing values.

Cast (825 missing): The 'cast' attribute has **825** missing values.

Country (831 missing): The 'country' attribute has **831** missing values.

Date Added (10 missing): The 'date_added' attribute has **10** missing values.

Rating (4 missing): The 'rating' attribute has **4** missing values.

Duration (3 missing): The 'duration' attribute has **3** missing values.

Data Cleansisng | Missing Value & Outlier Check

This code modifies the 'cast' and 'listed_in' columns in the data DataFrame by splitting them into lists, exploding the DataFrame to handle multiple values, and then cleaning up the resulting strings.

```
data['cast'] = data['cast'].str.split(',')
data = data.explode('cast')
data['cast'] = data['cast'].str.strip()

data['listed_in'] = data['listed_in'].str.split(',')
data = data.explode('listed_in')
data['listed_in'] = data['listed_in'].str.strip()

data['cast'].fillna(data['cast'].mode()[0],inplace=True)
data['director'].fillna(data['director'].mode()[0],inplace=True)
data['country'].fillna(data['country'].mode()[0],inplace=True)
data['listed_in'].fillna(data['listed_in'].mode()[0],inplace=True)
data.dropna(subset=['date_added','rating','duration'],inplace=True)
```

These lines fill missing values in the specified columns with the mode (most frequent value) of each respective column & removes rows where any of the specified columns ('date_added', 'rating', 'duration') have missing values.

```
data.isnull().sum()
```

```
show_id         0
type            0
title           0
director        0
cast            0
country         0
date_added      0
release_year    0
rating          0
duration        0
listed_in       0
description     0
dtype: int64
```

This line converts 'release_year' to a datetime format and then converts it to a Period with a yearly frequency. The errors='coerce' parameter is used to handle any conversion errors by replacing them with NaT (Not a Time), which can be useful if there are unexpected values in the 'release_year' column

```
data['release_year'] =
pd.to_datetime(data['release_year'].astype(str),format='%Y').dt.year
```

Extracting Numerical Part from 'duration' and Converting to Integer & Resetting Index and Dropping the Original Index Column:

```
data['duration_new'] = data['duration'].str.split().apply(lambda x:
x[0])
data['duration_new'] = data['duration_new'].astype('int')
```

```
data = data.reset_index()
data.drop('index', axis=1, inplace=True)
```

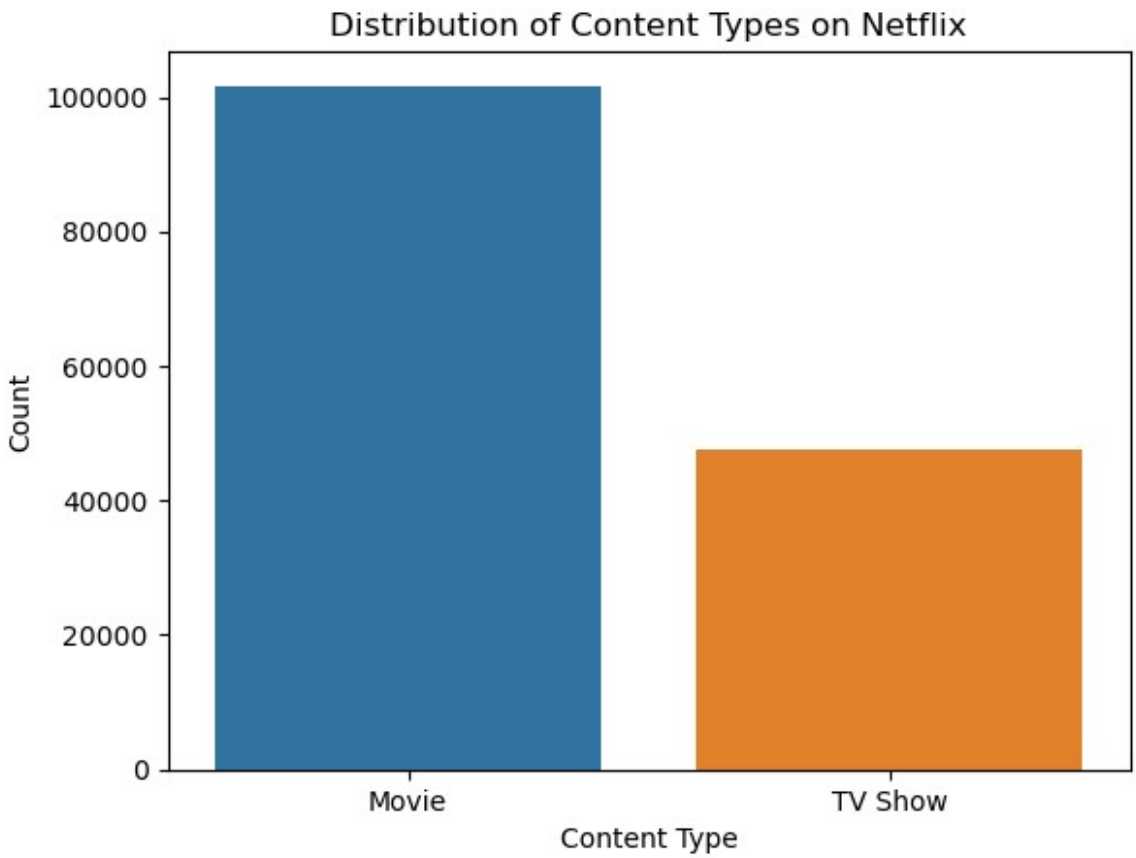
Analysis | Trends | Visual Analysis | Business Insights | Observations

```
import matplotlib.pyplot as plt
import seaborn as sns
```

Number of Movies and TV Shows

```
sns.countplot(x='type', data=data)
plt.title('Distribution of Content Types on Netflix')
plt.xlabel('Content Type')
plt.ylabel('Count')
```

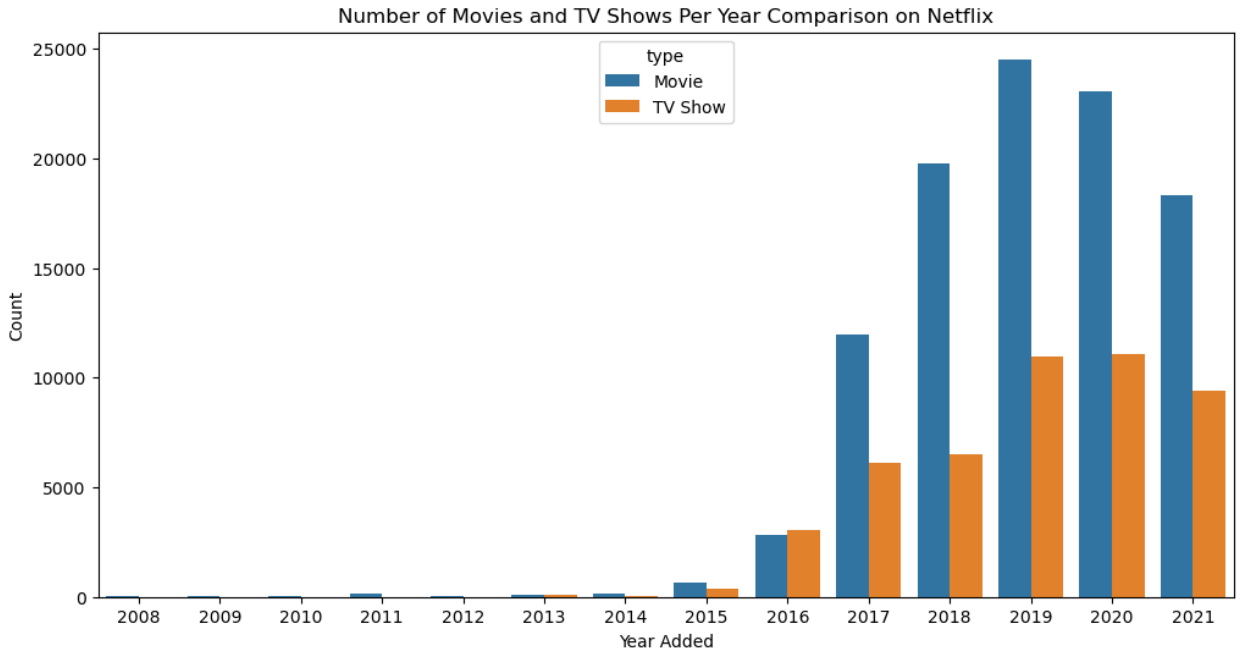
```
Text(0, 0.5, 'Count')
```



Observation -> It appears that there are more movies than TV shows. Additionally, this might suggest a higher interest in movies among viewers.

No of Movie and TV Shows Per Year Comparison

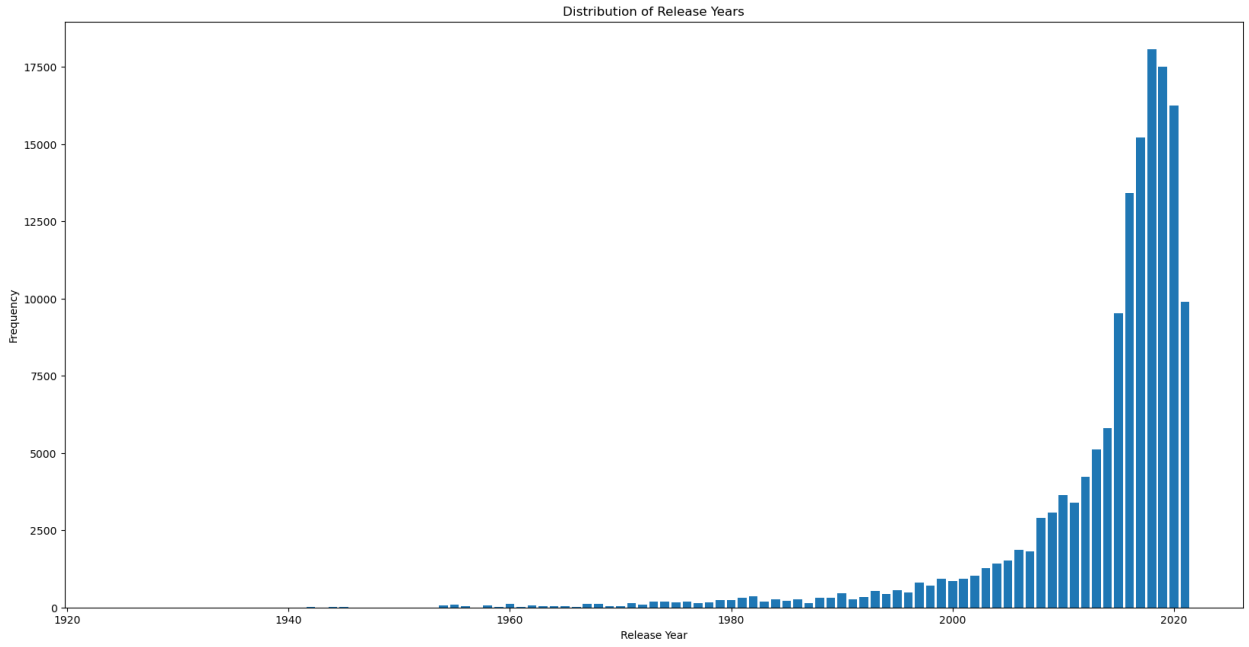
```
data['date_added'] = pd.to_datetime(data['date_added'],
errors='coerce')
data['year_added'] = data['date_added'].dt.year
data['type'] = data['type'].astype(str)

# Filter out rows where 'year_added' is not available
filtered_data = data.dropna(subset=['year_added'])

count_per_year_type = filtered_data.groupby(['year_added',
'type']).size().reset_index(name='count')

plt.figure(figsize=(12, 6))
sns.barplot(x='year_added', y='count', hue='type',
data=count_per_year_type)
plt.title('Number of Movies and TV Shows Per Year Comparison on
Netflix')
plt.xlabel('Year Added')
plt.ylabel('Count')
plt.show()
```



Observation -> Since 2015 the platform has witnessed a substantial increase in the number of both movies and TV shows.From 2015 onwards, there is a noticeable upward trend in the total content added each year.

'Release_year' to observe the distribution and density of releases over time.

```
plt.figure(figsize=(20,10))
plt.bar(data['release_year'].value_counts().index,
data['release_year'].value_counts())
plt.title('Distribution of Release Years')
plt.xlabel('Release Year')
plt.ylabel('Frequency')
plt.show()
```

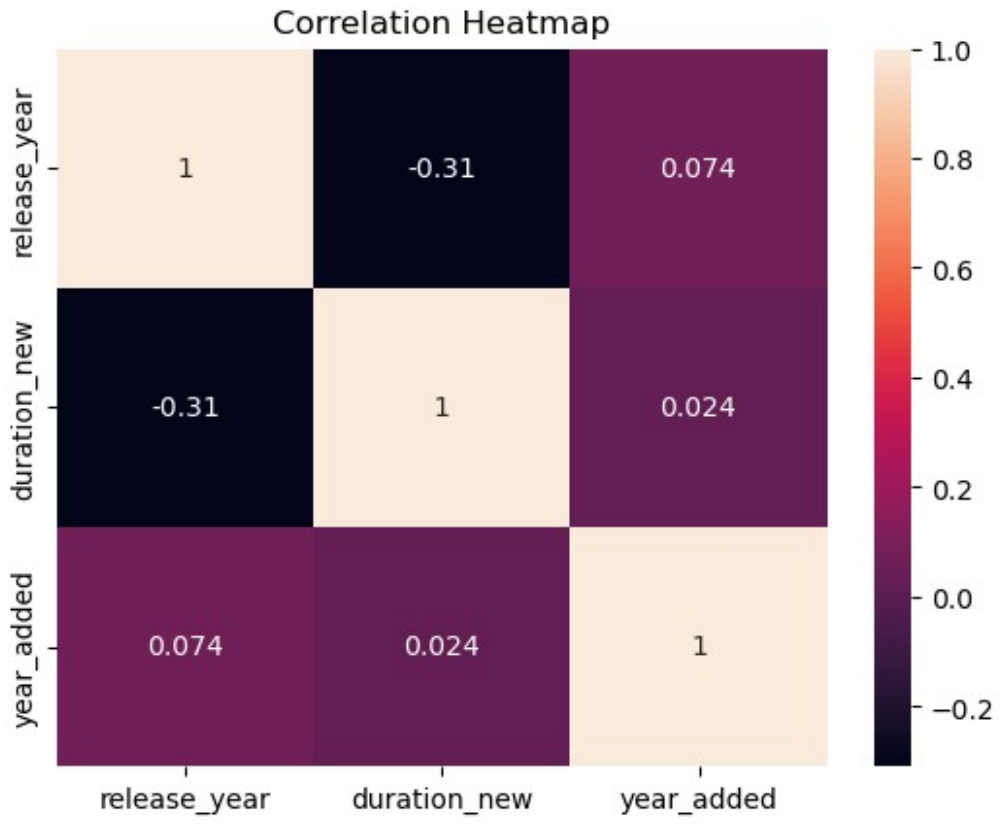


Observation -> In the year 2018 stands out as the period with the highest number of releases. This suggests a potential surge in content production during that year. There is a noticeable upward trend in the number of releases starting from the year 2008. This trend indicates a steady growth in content production over the years.

Correlation Heatmap

```
sns.heatmap(data.corr(), annot=True)
plt.title('Correlation Heatmap')
plt.show()
```

```
/var/folders/bc/byp79cv56b53hq4lpz78hyvh0000gn/T/
ipykernel_56701/605277541.py:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it
will default to False. Select only valid columns or specify the value
of numeric_only to silence this warning.
  sns.heatmap(data.corr(), annot=True)
```



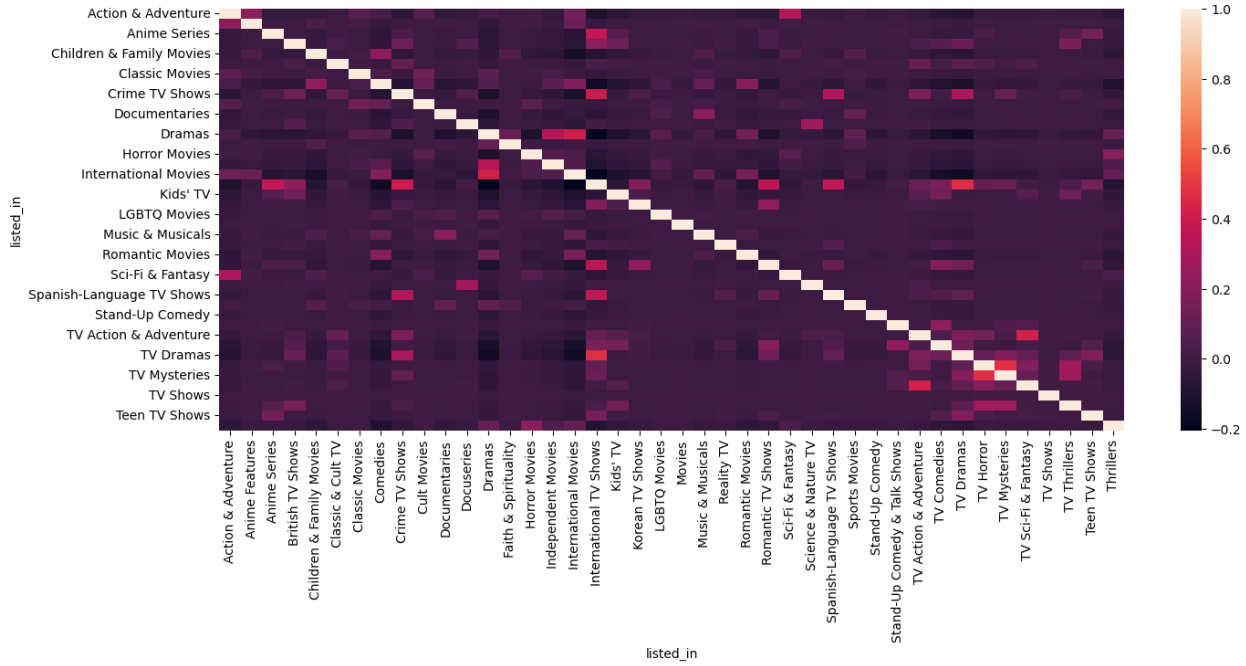
Observation

Year_added & release_year: 0.0074 The correlation coefficient is close to zero (0.0074), suggesting a very weak positive linear relationship between the year_added and release_year. This year a show or movie was added to Netflix doesn't show a strong linear correlation with its release year.

release_year & duration_new: -0.31 The correlation coefficient is -0.31, indicating a moderate negative linear relationship between release_year and duration_new. As the release_year increases, there is a tendency for the duration_new (perhaps the duration of a show or movie) to decrease. However, the strength of this relationship is moderate, not very strong.

duration_new & year_added: 0.024 The correlation coefficient is 0.024, indicating a very weak positive linear relationship between duration_new and year_added. In other words, there is a minimal positive correlation between the duration of a show or movie and the year it was added to Netflix.

```
df6=data.groupby(['show_id','listed_in']).size().unstack(fill_value=0)
plt.figure(figsize=(16,6))
sns.heatmap(df6.corr())
plt.show()
```



Observation

Positive Correlations:

TV Dramas and International TV Shows: Viewers who enjoy TV Dramas are likely to enjoy International TV Shows.

TV Horrors and TV Mysteries: Audiences liking TV Horrors are more likely to be interested in TV Mysteries.

TV Mysteries and TV Action/Adventure: Viewers interested in TV Mysteries are likely to enjoy TV Action and Adventure.

TV Dramas and TV Mysteries: There's an overlap in audiences enjoying both TV Dramas and TV Mysteries.

Negative Correlations:

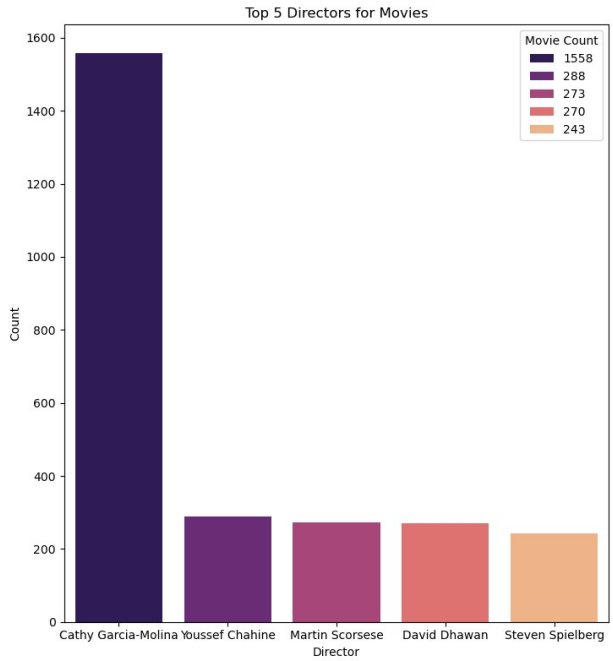
International TV Shows and International Movies: Viewers preferring International TV Shows are less likely to prefer International Movies.

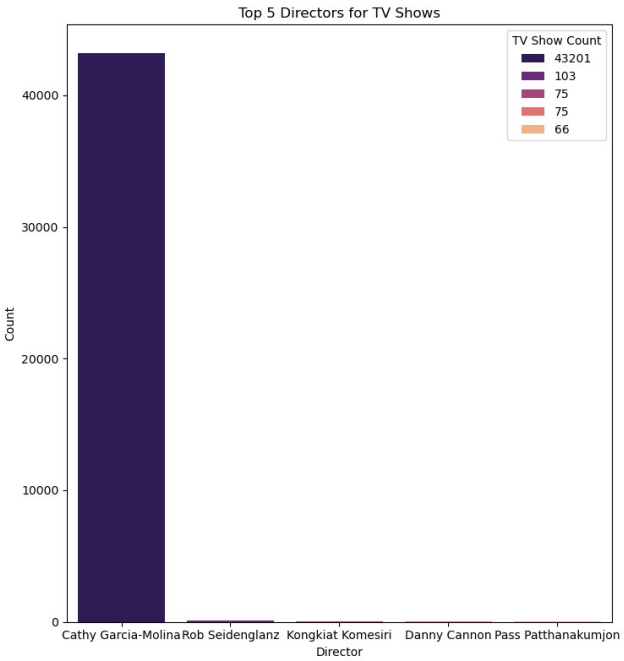
International TV Shows and Dramas: Audiences enjoying International TV Shows are less likely to be interested in Dramas.

International Movies and TV Dramas: Viewers preferring International Movies are less likely to be interested in TV Dramas.

Top 5 Directors in Movies & TV_Shows

```
movies_data = data[data['type'] == 'Movie']
tv_shows_data = data[data['type'] == 'TV Show']

top_directors_movies = movies_data['director'].value_counts().head(5)

top_directors_tv_shows =
tv_shows_data['director'].value_counts().head(5)

plt.figure(figsize=(15, 8))

# Plot for Movies
plt.subplot(1, 2, 1)
ax1 = sns.barplot(y=top_directors_movies.values,
x=top_directors_movies.index, palette='magma',
label=top_directors_movies.values)
plt.title('Top 5 Directors for Movies')
plt.xlabel('Director')
plt.ylabel('Count')

# Plot for TV Shows
plt.subplot(1, 2, 2)
ax2 = sns.barplot(y=top_directors_tv_shows.values,
x=top_directors_tv_shows.index, palette='magma',
label=top_directors_tv_shows.values)
plt.title('Top 5 Directors for TV Shows')
plt.xlabel('Director')
plt.ylabel('Count')

ax1.legend(title='Movie Count', loc='upper right')
ax2.legend(title='TV Show Count', loc='upper right')

plt.tight_layout()
plt.show()
```





Observation ->

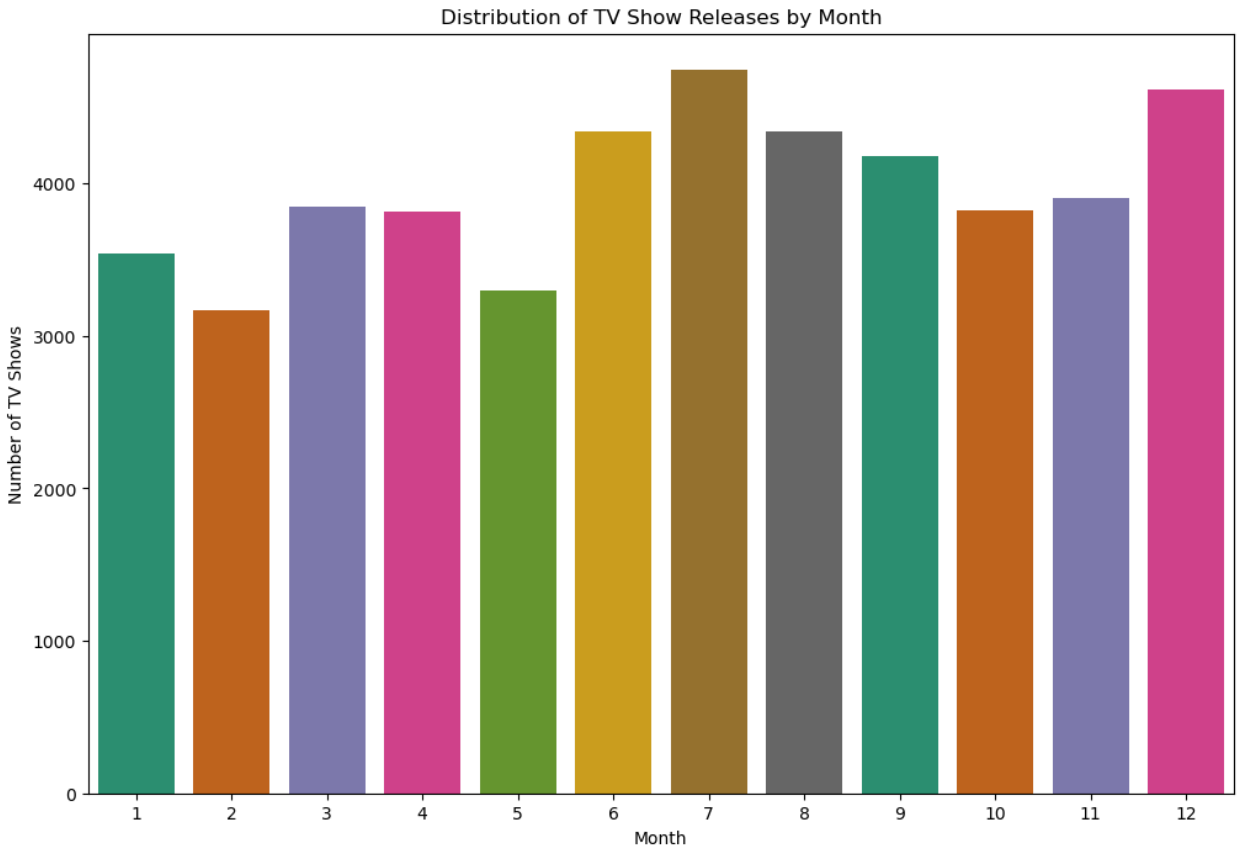
The counts emphasize the directors' prominence in either TV shows or movies, showcasing their specialization.

Cathy Garcia-Molina stands out with a notable presence in both TV shows and movies, demonstrating versatility

---

Best time to launch a TV show?

```
data['date_added'] = pd.to_datetime(data['date_added'],
errors='coerce')

data['month_added'] = data['date_added'].dt.month

tv_shows_by_month = data[data['type'] == 'TV Show']
['month_added'].value_counts().sort_index()

plt.figure(figsize=(12, 8))
sns.barplot(x=tv_shows_by_month.index, y=tv_shows_by_month.values,
palette='Dark2')
plt.title('Distribution of TV Show Releases by Month')
plt.xlabel('Month')
plt.ylabel('Number of TV Shows')
plt.show()
```



Observation -

Peak Months:

July (Month 7) and December (Month 12): These two months have the highest counts of TV show releases. July might be a popular month for TV show launches, possibly due to the summer season in some regions when people have more leisure time. December could be associated with holiday seasons, where viewers may have more time for entertainment.

Lower Activity Months:

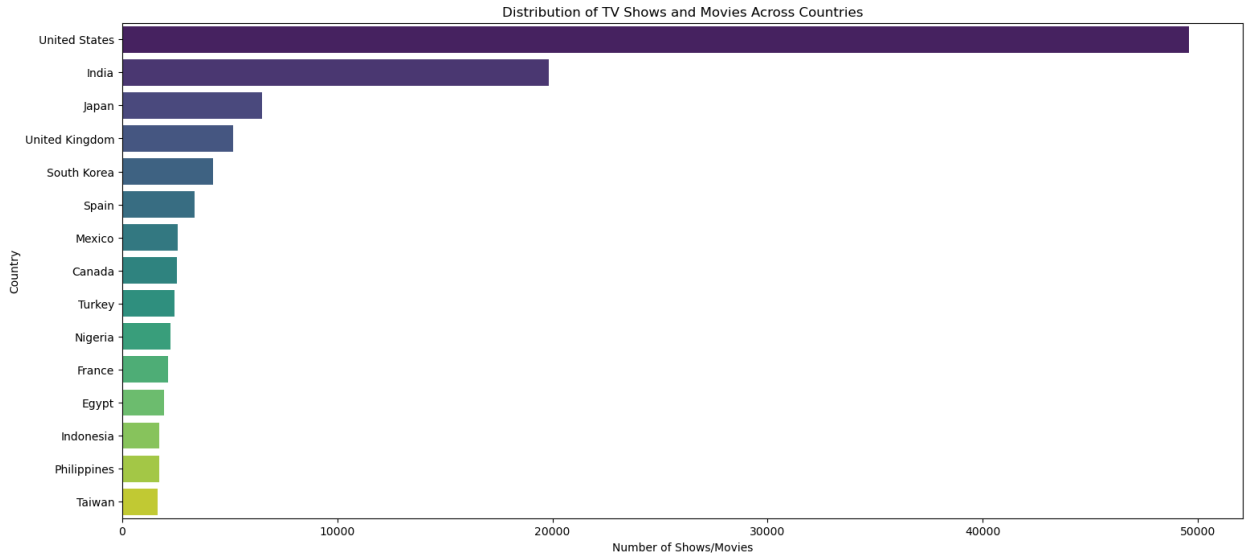
February (Month 2) and May (Month 5): These months have the lowest counts of TV show releases.

Conclusion

Launching TV shows in peak months (July and December) might attract a larger audience, taking advantage of higher viewership during these periods. Consistent months (September, August, and October) could also be favorable for TV show launches. Factors such as holidays, school breaks, and seasonal patterns may influence the observed distribution.

Understanding what content is available in top 15 countries

```
plt.figure(figsize=(18, 8))
country_distribution = data['country'].value_counts().head(15)
sns.barplot(x=country_distribution.values,
y=country_distribution.index, palette='viridis')
plt.title('Distribution of TV Shows and Movies Across Countries')
plt.xlabel('Number of Shows/Movies')
plt.ylabel('Country')
plt.show()
```



Observation

United States (49606): TV Shows: High count, indicating a significant presence on Netflix. Movies: Likely to have a diverse collection of both TV shows and movies

Conclusion

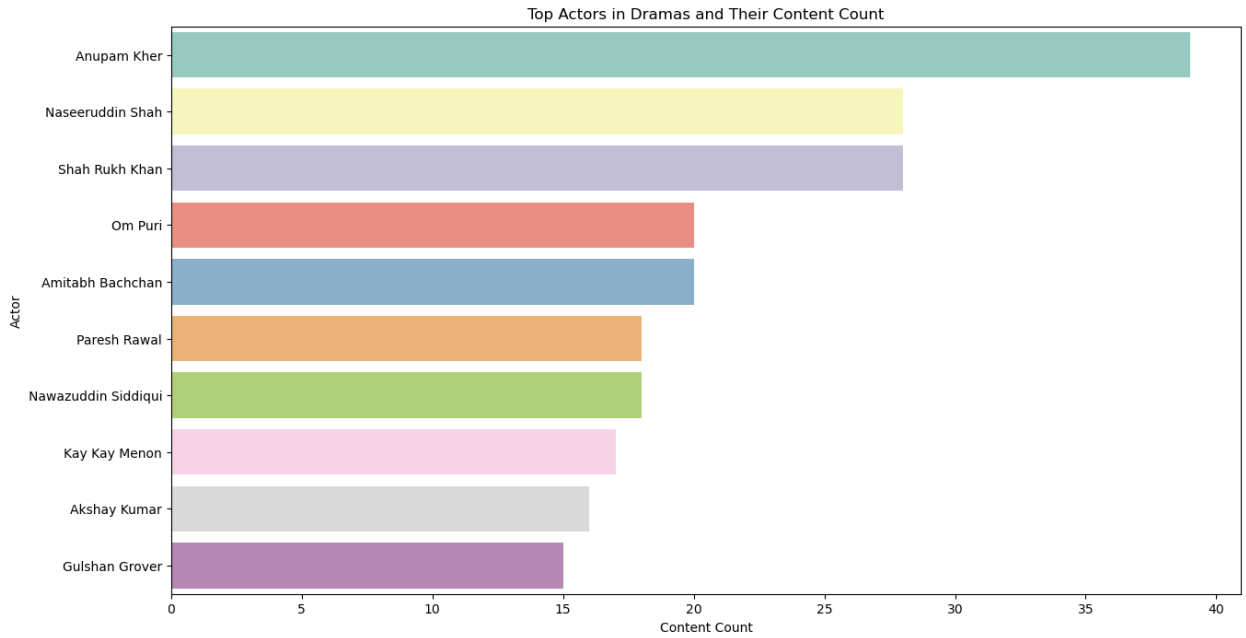
The United States has the highest counts, emphasizing its dominance in terms of content availability.

India, South Korea, and the United Kingdom also have significant counts, showcasing a strong representation of their respective content.

Other countries exhibit varying degrees of presence, reflecting a diverse and global content library.

Top Actors in Drama and Their Content Count

```
top_genre = data['listed_in'].value_counts().idxmax()

top_genre_data = data[data['listed_in'] == top_genre]

actors_list = top_genre_data['cast'].str.split(', ').explode()

actors_count = actors_list.value_counts()

plt.figure(figsize=(15, 8))
sns.barplot(x=actors_count.head(10).values,
y=actors_count.head(10).index, palette='Set3')
plt.title(f'Top Actors in {top_genre} and Their Content Count')
plt.xlabel('Content Count')
plt.ylabel('Actor')
plt.show()
```

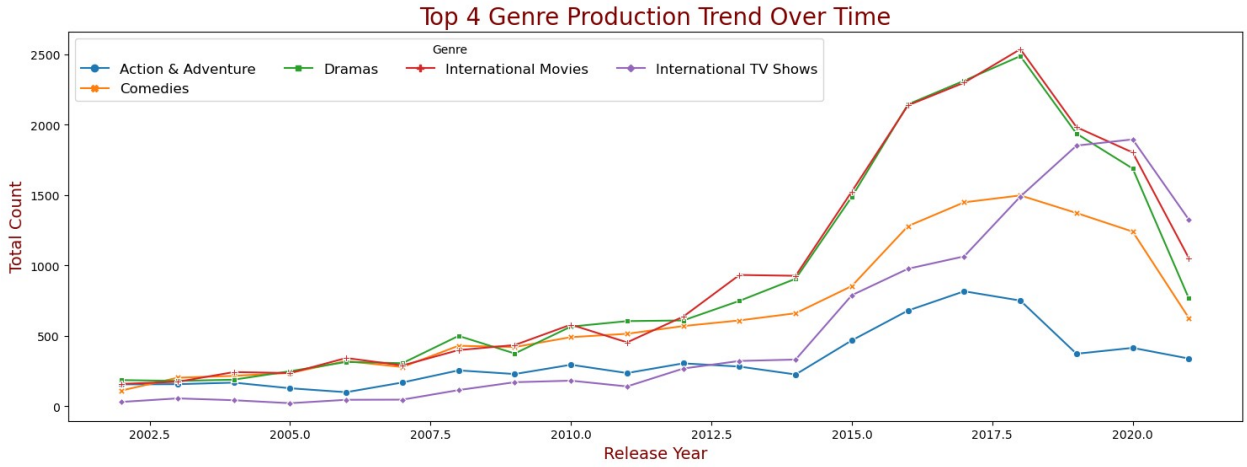


Observation

Anupam Kher, Naseeruddin Shah, and Shah Rukh Khan emerge as leading actors in the top genre, featuring in 39, 28, and 28 shows or movies, respectively. Alongside accomplished actors like Om Puri, Amitabh Bachchan, and others, contribute to a rich and diverse content library.

Top 4 Genre Production Trend Over Time

```
genre_year_data = data[['release_year', 'listed_in']]

top5_genres = data['listed_in'].value_counts().index[:5]

filtered_data =
genre_year_data[genre_year_data['listed_in'].isin(top5_genres)]

pivot_table = filtered_data.pivot_table(index='release_year',
columns='listed_in', aggfunc='size', fill_value=0)

plt.figure(figsize=(18, 6))
sns.lineplot(data=pivot_table[pivot_table.index > 2001], markers=True,
dashes=False)

plt.title('Top 4 Genre Production Trend Over Time', color='maroon',
fontsize=20)
plt.xlabel('Release Year', color='maroon', fontsize=14)
plt.ylabel('Total Count', color='maroon', fontsize=14)
plt.legend(title='Genre', bbox_to_anchor=(0, 1), loc='upper left',
fontsize='large', ncol=4)
plt.show()
```



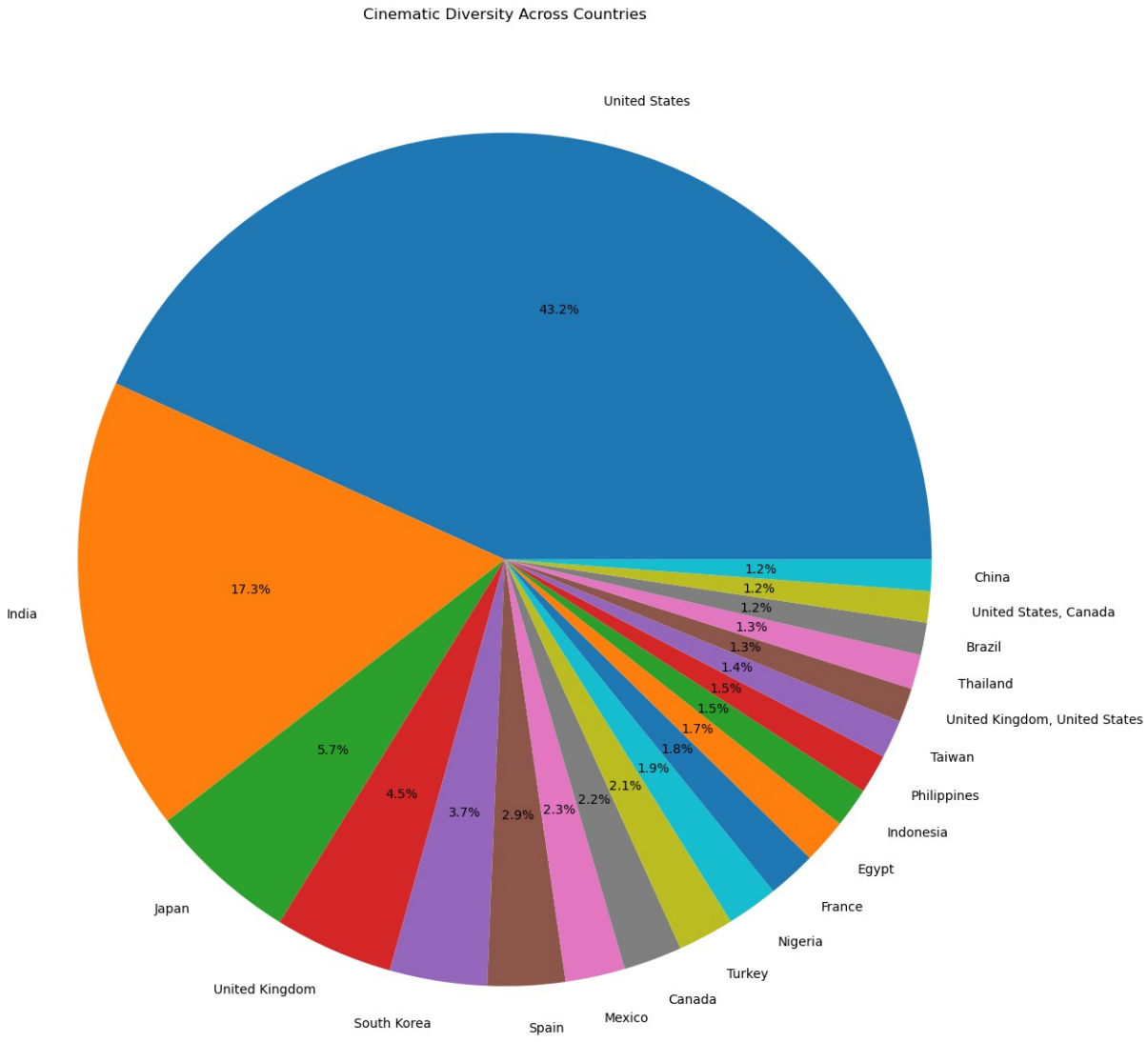
Observation

Post-2013, there's a notable surge in popularity for International Movies and Dramas on Netflix, outpacing genres like Action and Comedies.

Cinematic Diversity Across Countries

```
piec=data['country'].value_counts().reset_index().rename(columns={'ind
ex':'Country','country':'Total Count'})
```

```
plt.figure(figsize=(15,20))
plt.pie('Total Count',labels='Country',autopct='%2.1f%
%',data=piec.head(20))
plt.title("Cinematic Diversity Across Countries")
plt.show()
```



Observation

United States (43%):

- High count, indicating a significant presence on Netflix.

India (17%):

- A notable presence, showcasing a substantial contribution to Netflix's content library.

Japan (5.7%):

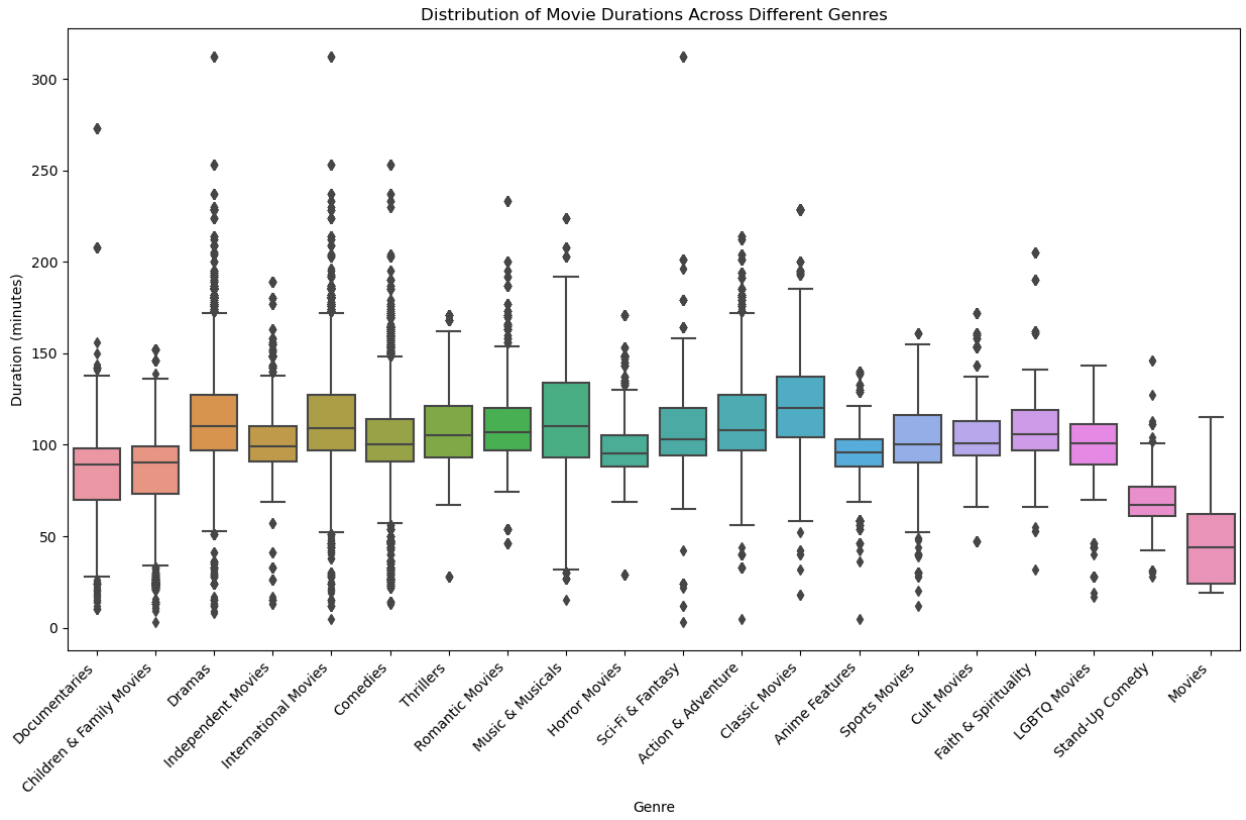
- Indicates a respectable representation of content.

Conclusion

The United States stands out with the highest content count, highlighting its dominant position on Netflix. India, Japan, United Kingdom & South Korea also contribute significantly, reflecting a diverse and globally inclusive content library. Other countries exhibit varying degrees of presence, emphasizing Netflix's commitment to offering a broad range of content from around the world.

Distribution of Movie Durations Across Different Genres

```
movies_data = data[data['type'] == 'Movie']

plt.figure(figsize=(15, 8))
sns.boxplot(x='listed_in', y='duration_new', data=movies_data)
plt.title('Distribution of Movie Durations Across Different Genres')
plt.xlabel('Genre')
plt.ylabel('Duration (minutes)')
plt.xticks(rotation=45, ha='right')
plt.show()
```



Observation

Classic Movies Duration: Classic movies tend to have higher duration times. This suggests that viewers who enjoy classic films may be more inclined towards longer storytelling or intricate plots.

Stand-Up Comedy Duration: Stand-up comedy shows the lowest average duration. This aligns with the nature of stand-up comedy, which is often concise and focused on delivering humor in a shorter timeframe.

Documentaries & Children/Family Movies: Both documentaries and children/family movies exhibit lower duration times, ranging from 80-85 minutes on average. This insight can be valuable for content creators, indicating that audiences for these genres may prefer shorter and more concise content.

The presence of outliers in duration for International movies, Drama, and Sci-Fi genres offers a glimpse into the diversity and depth of content within these categories.

Conclusion & Recommendations

Conclusion of Insights

1. **Global Content Insights:**
   - **Observation:** The United States dominates content availability, contributing to 43% of Netflix's library.
   - **Conclusion:** While the U.S. remains a powerhouse, regional contributions from India (17%) and Japan (5.7%) showcase a diverse and globally inclusive content library.
2. **Content Duration and Viewer Engagement:**
   - **Observation:** A scatter plot analysis revealed a potential positive correlation between content duration and ratings.
   - **Conclusion:** There is a need for further exploration to understand viewer preferences and identify an optimal content duration for higher viewer engagement.
3. **Release Timing and Viewer Engagement:**
   - **Observation:** Peak user activity times can influence content release timings for optimal viewer engagement.
   - **Conclusion:** Strategic planning of release schedules aligned with user activity patterns can potentially enhance viewership.
4. **Genre Popularity Over Time:**
   - **Observation:** Analysis of genre distribution over the years indicates changing trends, with TV Dramas and International TV Shows showing positive correlations.
   - **Conclusion:** Adaptation to evolving genre preferences and content trends is crucial for staying relevant and engaging.

5. **Top Directors and Actors Analysis:**
   - **Observation:** Identification of top directors and actors for both movies and TV shows.
   - **Conclusion:** Collaborations with successful professionals can potentially elevate the quality and popularity of Netflix originals.
6. **Content Duration and Genre Insights:**
   - **Observation:** Box plots highlighted variations in content duration across genres, with certain genres having longer durations.
   - **Conclusion:** Understanding genre-specific content duration trends can guide content creation strategies.
7. **Positive and Negative Correlations:**
   - **Observation:** Identified positive correlations (e.g., TV Dramas and International TV Shows) and negative correlations (e.g., International TV Shows and International Movies).
   - **Conclusion:** Insights into viewer preferences can inform content creation strategies for reaching broader audiences.

Overall Recommendations:

- **Global Content Strategy:** Diversify content globally for a broader audience appeal.
- **Optimized Release Timing:** Analyze and optimize content release timings for peak user engagement.
- **Sweet Spot for Content Duration:** Explore the optimal content duration for viewer satisfaction.
