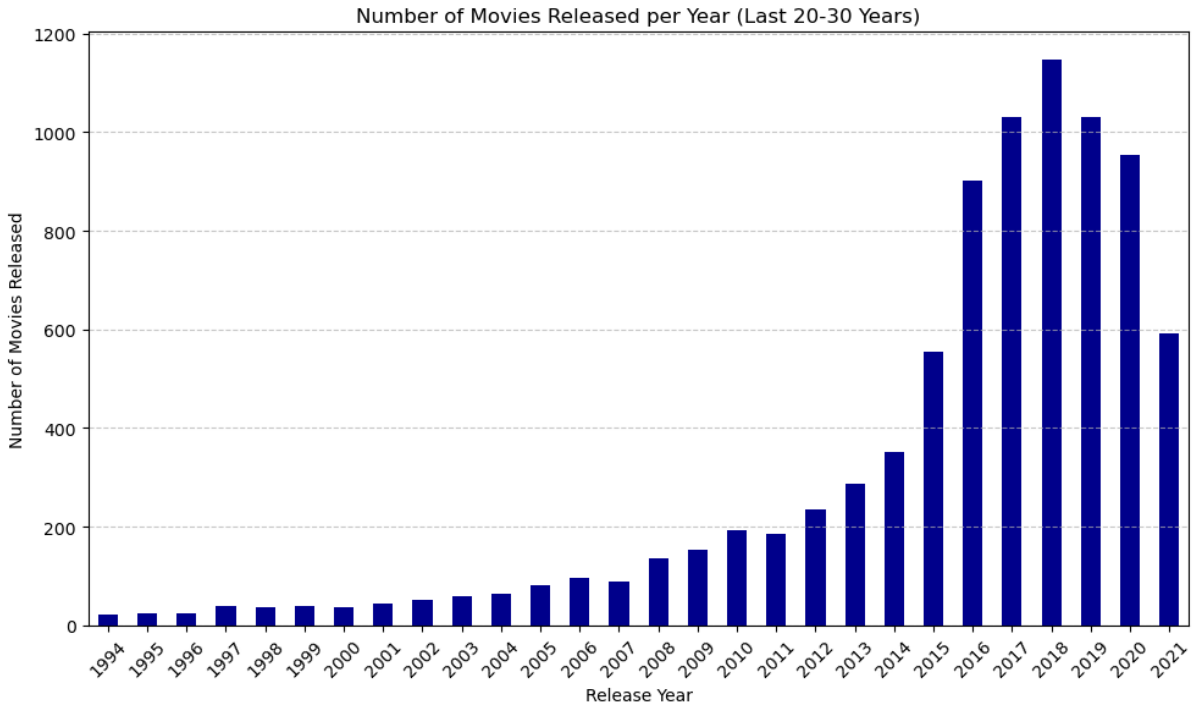
SCALER CASE STUDY- NETFLIX DATA

HIMKANT NIGAM

1. How has the number of movies released per year changed over the last 20-30 years?

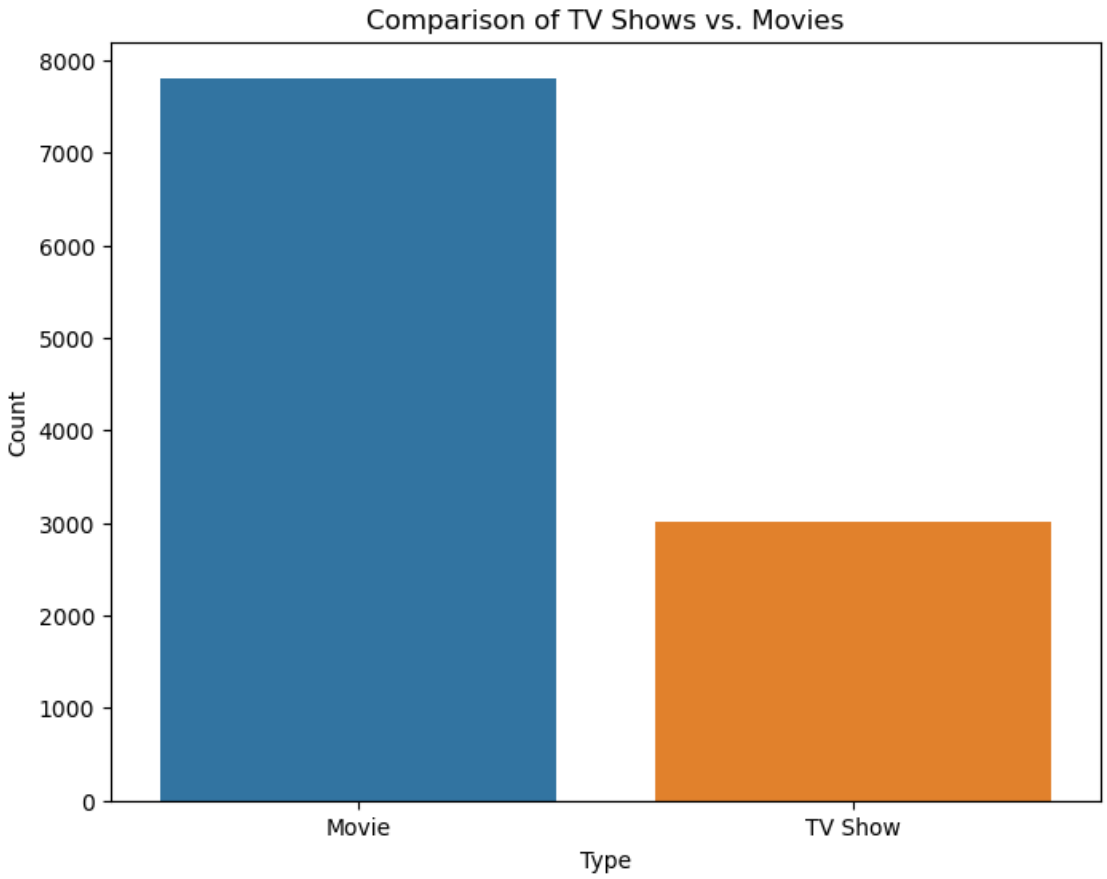
```
# Extracting release year information
df['release_year'] = df['release_year'].astype(str)
df['Release_year'] = df['release_year'].str[-4:].astype(int)

# Filtering data for the last 20-30 years
current_year = pd.Timestamp.now().year
start_year = current_year - 30
end_year = current_year
movies_last_20_30_years = df[(df['Release_year'] >= start_year) & (df['Release_year'] <= end_year)]

# Counting the number of movies released each year
movies_per_year = movies_last_20_30_years.groupby('Release_year').size()

# Visualizing the trend over time
plt.figure(figsize=(10, 6))
movies_per_year.plot(kind='bar', color='darkblue')
sns.kdeplot(movies_per_year.index, fill=True, color='orange', linestyle='--', linewidth=10)
plt.title('Number of Movies Released per Year (Last 20-30 Years)')
plt.xlabel('Release Year')
plt.ylabel('Number of Movies Released')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```

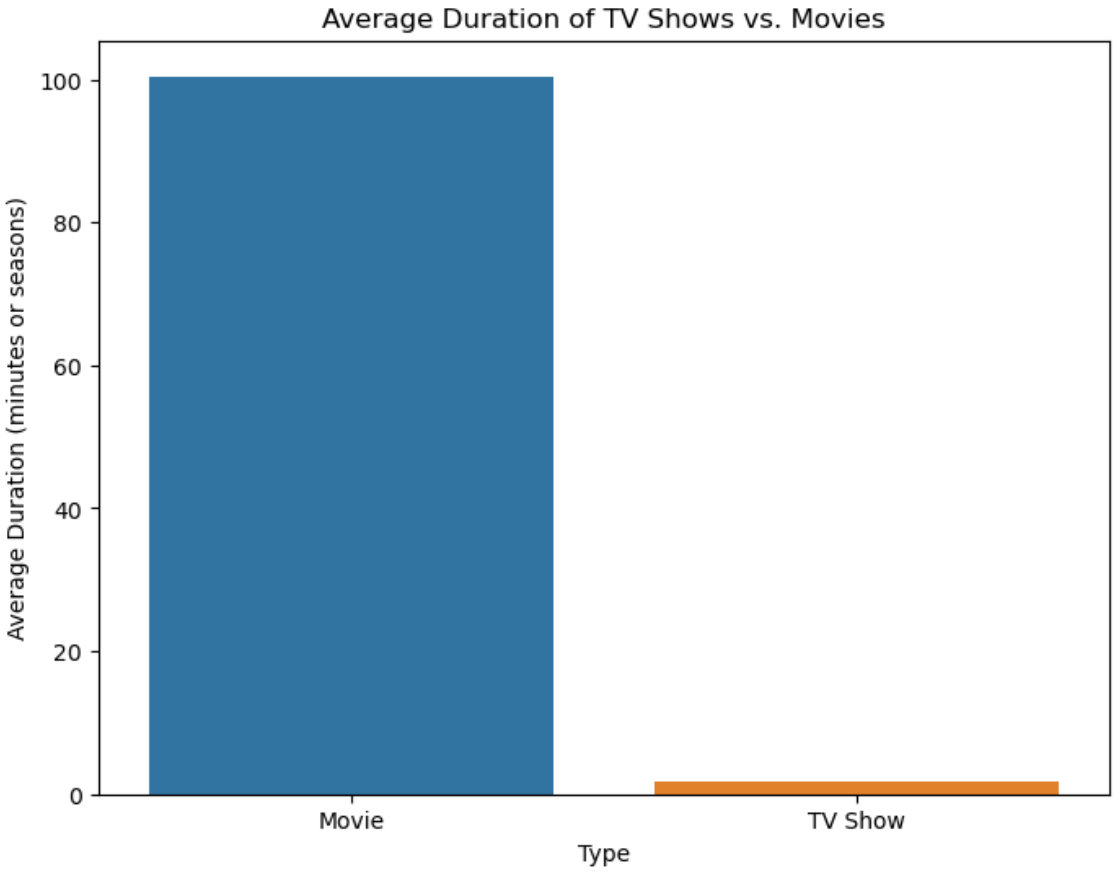


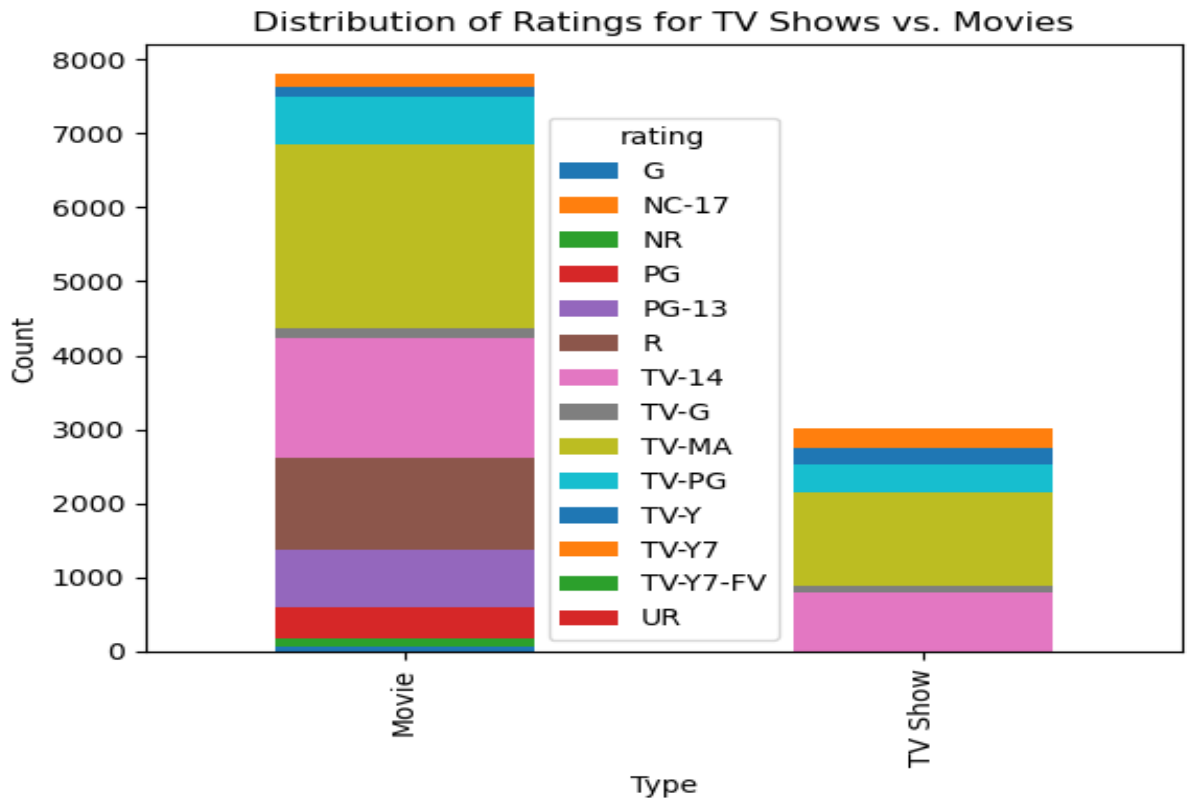
Finding :
- we can see our data is highly left skewed.
- there seems to be a positive relation with number of movies and years, as years are increasing, number of movies are also increasing though it seems that towards the end from 2019, umber of movies started to decrease gradually because of some reason.

2. Comparison of tv shows vs. movies.

```
# Group by 'Type' to count the number of TV shows and movies
type_counts = df['type'].value_counts()

# Visualize the comparison of TV shows and movies
plt.figure(figsize=(8, 6))
sns.barplot(x=type_counts.index, y=type_counts.values)
plt.title('Comparison of TV Shows vs. Movies')
plt.xlabel('Type')
plt.ylabel('Count')
plt.show()

# Average duration for TV shows vs. movies
df['Duration Value'] = df['duration'].str.extract(r'(\d+)').astype(int)
df['Duration Type'] = df['duration'].str.extract(r'(\w+)')
average_duration = df.groupby('type')['Duration Value'].mean()

# Visualize the average duration for TV shows and movies
plt.figure(figsize=(8, 6))
sns.barplot(x=average_duration.index, y=average_duration.values)
plt.title('Average Duration of TV Shows vs. Movies')
plt.xlabel('Type')
plt.ylabel('Average Duration (minutes or seasons)')
plt.show()

# Analysis of ratings distribution for TV shows vs. movies
rating_counts = df.groupby('type')['rating'].value_counts().unstack()
plt.figure(figsize=(15, 8))
rating_counts.plot(kind='bar', stacked=True)
plt.title('Distribution of Ratings for TV Shows vs. Movies')
plt.xlabel('Type')
plt.ylabel('Count')
plt.show()
```


Comparison of TV Shows vs. Movies
Count
8000
7000
6000
5000
4000
3000
2000
1000
0
Movie
TV Show
Type


Average Duration of TV Shows vs. Movies
Average Duration (minutes or seasons)
100
80
60
40
20
0
Movie
TV Show
Type

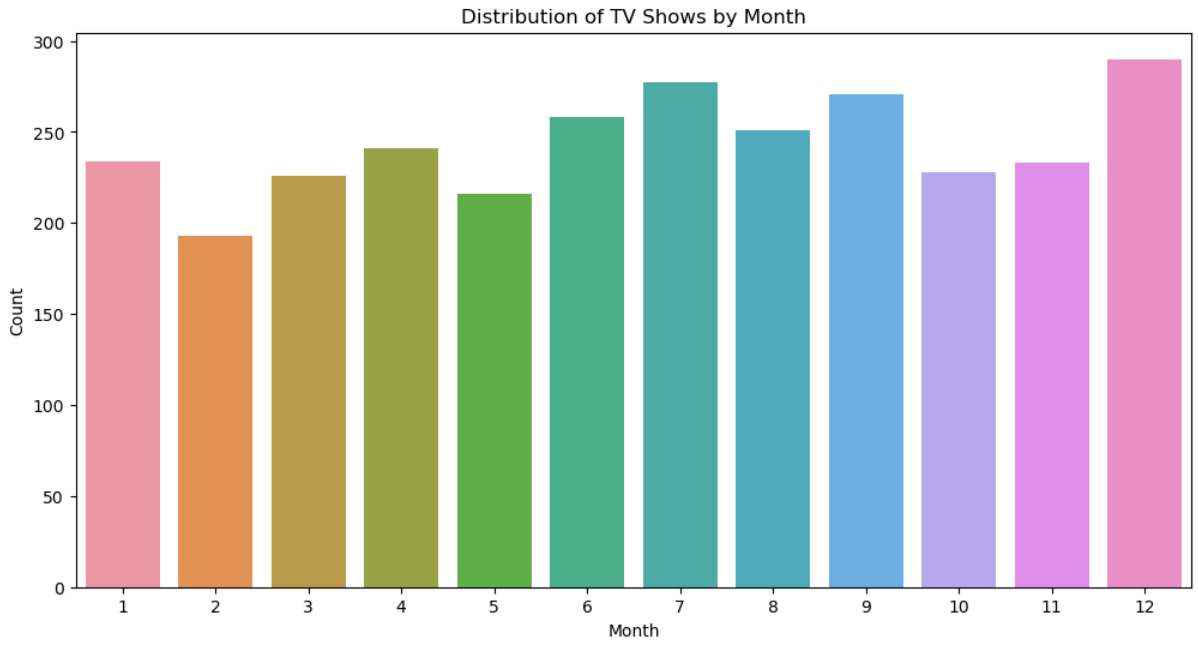


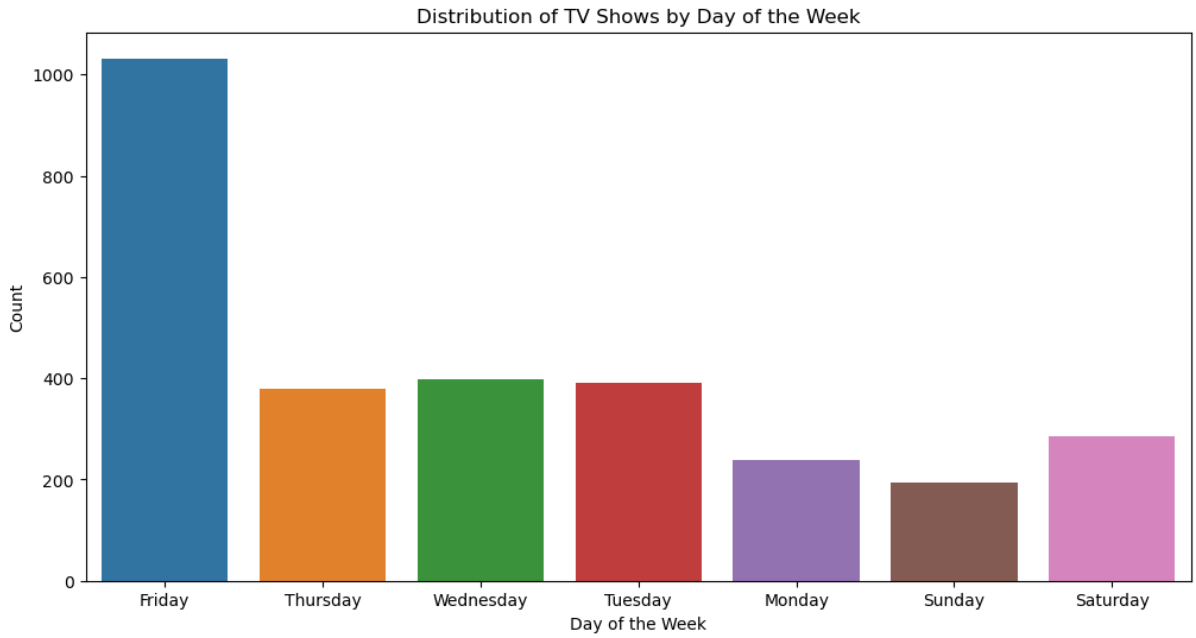
FINDING :

- TV show is less than 50% of movies in count
- Considering a person watches one episode or one movie per day, movie duration is much higher than of television.

3. What is the best time to launch a TV show?

```
# Convert 'date_added' to datetime format
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

# Drop rows with invalid or missing dates
df.dropna(subset=['date_added'], inplace=True)

# Extract the month and day of the week from 'date_added'
df['month'] = df['date_added'].dt.month
df['day_of_week'] = df['date_added'].dt.day_name()

# Determine the distribution of TV shows across months
plt.figure(figsize=(12, 6))
sns.countplot(x='month', data=df[df['type'] == 'TV Show'])
plt.title('Distribution of TV Shows by Month')
plt.xlabel('Month')
plt.ylabel('Count')
plt.show()

# Determine the distribution of TV shows across days of the week
plt.figure(figsize=(12, 6))
sns.countplot(x='day_of_week', data=df[df['type'] == 'TV Show'])
plt.title('Distribution of TV Shows by Day of the Week')
plt.xlabel('Day of the Week')
plt.ylabel('Count')
plt.show()
```



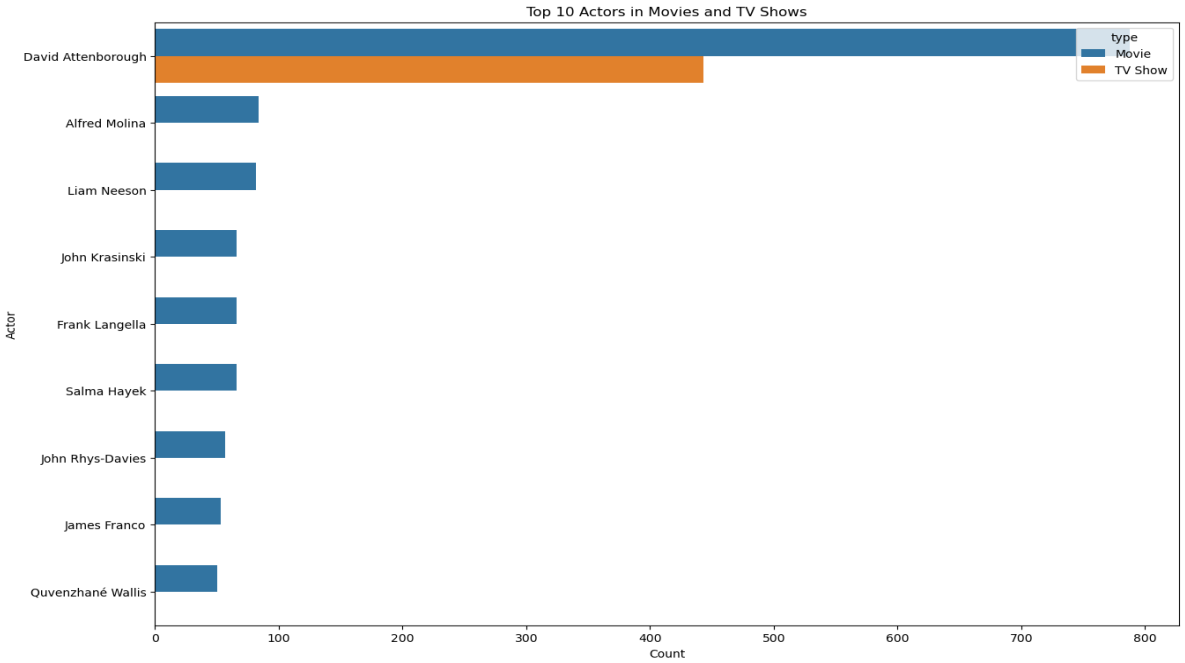


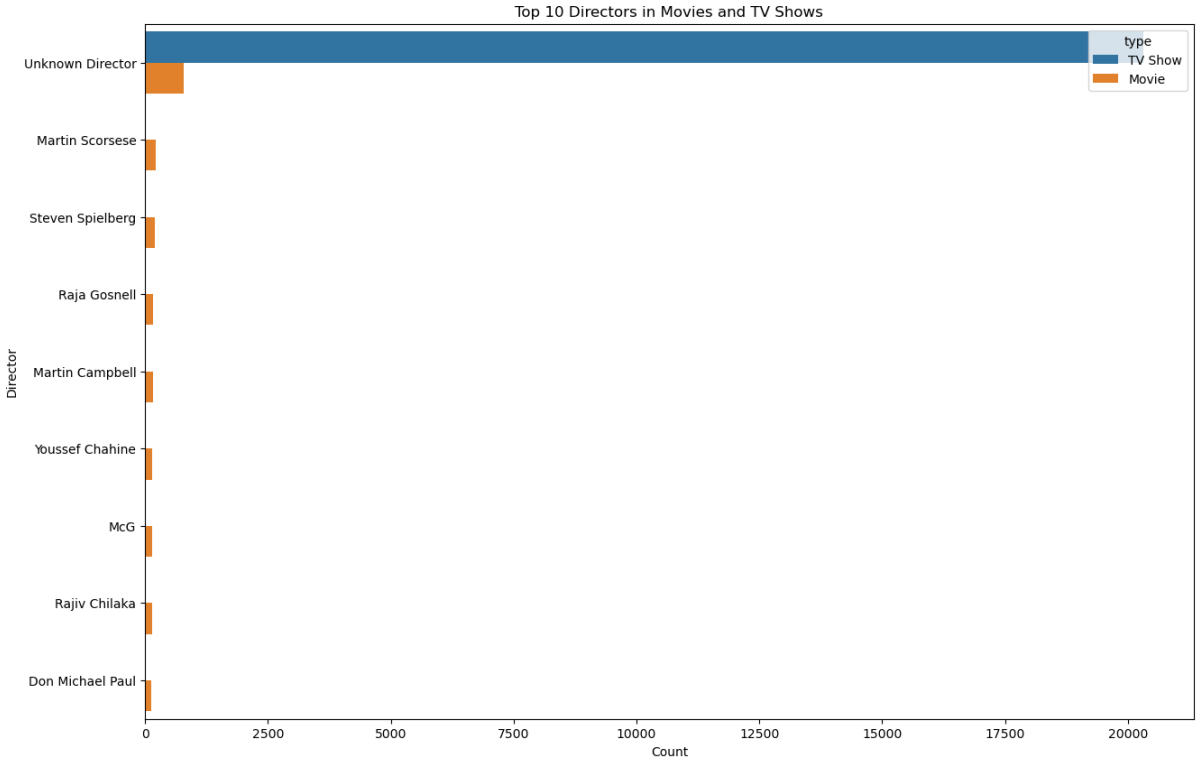
FINDING :

- Most of the TV shows are watched on Friday considering it is a weekend night.
- Most of the new TV shows are either launched or replayed on last month of the year considering people utilising holidays and new year starting.

4. Analysis of actors/directors of different types of shows/movies.

```
# Split the 'Cast' and 'Director' columns to handle multiple names
df['cast'] = df['cast'].str.split(', ')
df['director'] = df['director'].str.split(', ')

# Explode the 'Cast' and 'Director' columns to get individual actors and directors
df = df.explode('cast')
df = df.explode('director')

# Group by 'Type' to get counts of unique actors and directors for Movies and TV Shows
actor_type_counts = df.groupby(['type', 'cast']).size().reset_index(name='Count')
director_type_counts = df.groupby(['type', 'director']).size().reset_index(name='Count')

# Visualize the most common actors for Movies and TV Shows
plt.figure(figsize=(15, 10))
sns.barplot(x='Count', y='cast', data=actor_type_counts.sort_values('Count',
ascending=False)[:10], hue='type')
plt.title('Top 10 Actors in Movies and TV Shows')
plt.xlabel('Count')
plt.ylabel('Actor')
plt.show()

# Visualize the most common directors for Movies and TV Shows
plt.figure(figsize=(15, 10))
sns.barplot(x='Count', y='director', data=director_type_counts.sort_values('Count',
ascending=False)[:10], hue='type')
plt.title('Top 10 Directors in Movies and TV Shows')
plt.xlabel('Count')
plt.ylabel('Director')
plt.show()
```



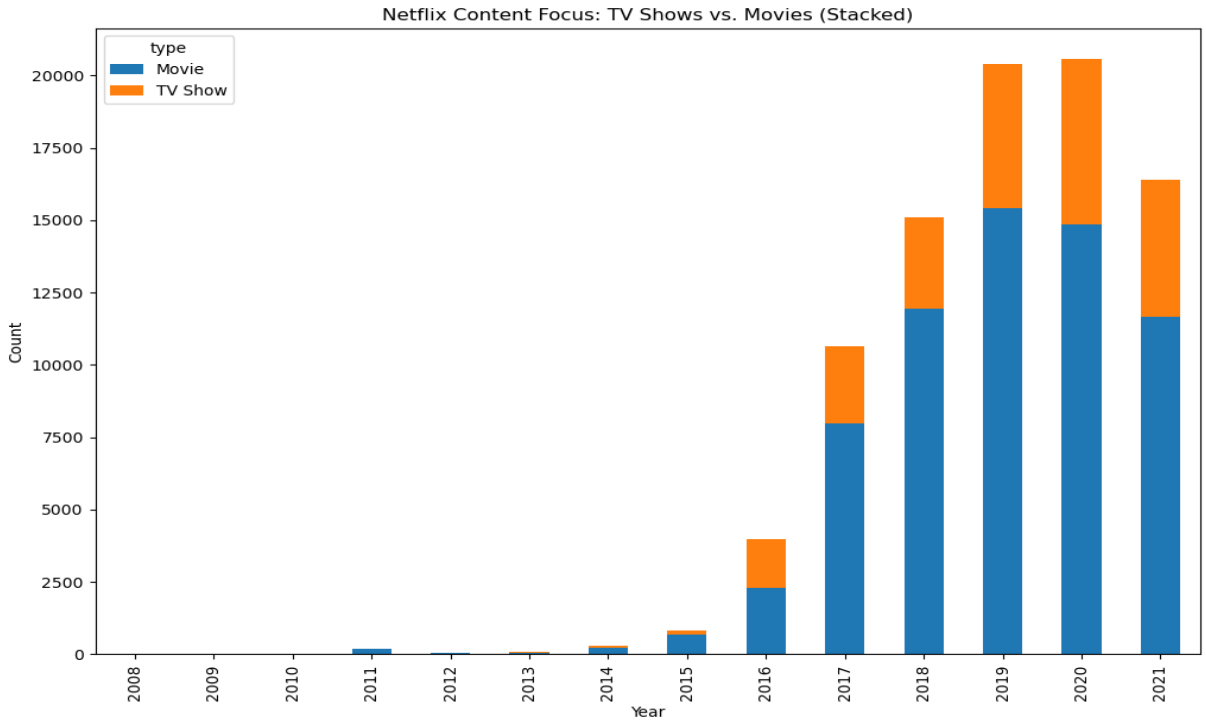


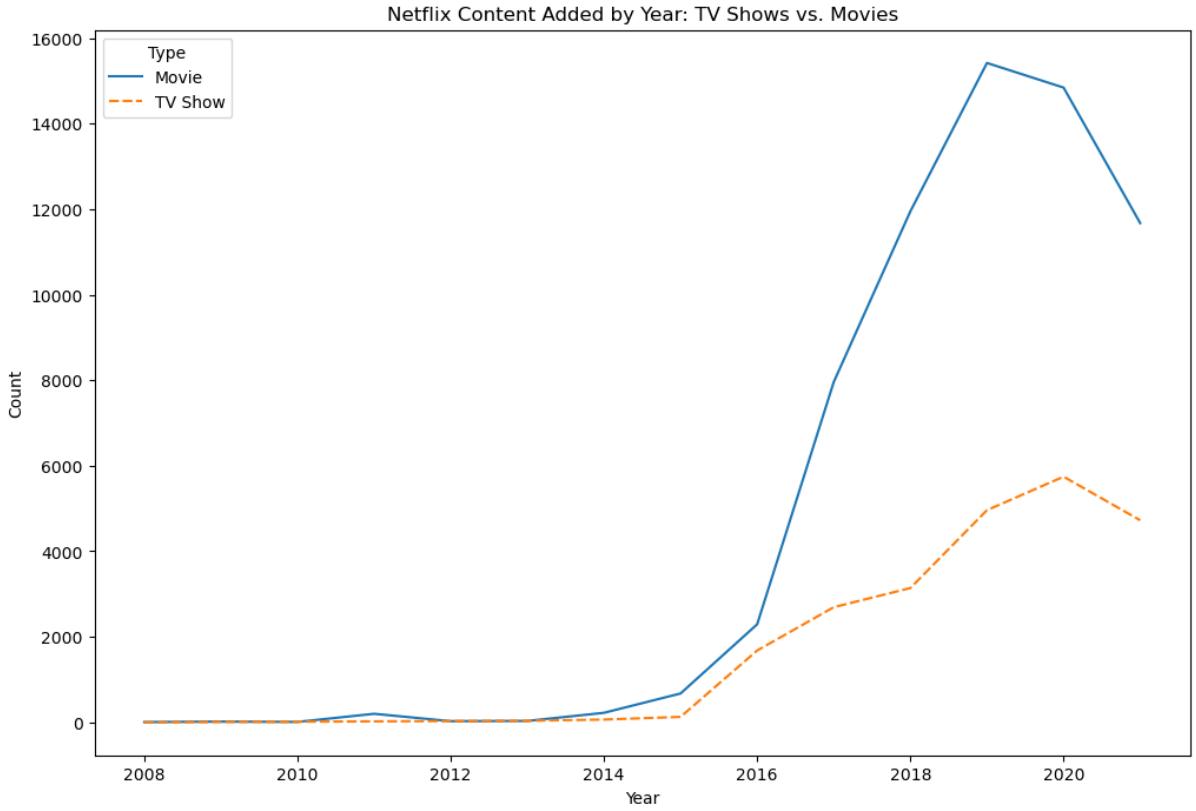
FINDING :

- Most of the TV shows have unknown directors, either they are new or not known.
- David Attenborough is the only director that makes both movies and television shows.

5. Does Netflix has more focus on TV Shows than movies in recent years

```
# Convert 'date_added' to datetime format
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')

# Extract the year from 'date_added'
df['year_added'] = df['date_added'].dt.year

# Group by 'year_added' and 'type' to count TV shows and movies added each year
year_type_counts = df.groupby(['year_added', 'type']).size().reset_index(name='count')

# Pivot the data for easier plotting
year_type_pivot = year_type_counts.pivot(index='year_added', columns='type',
values='count')

# Line plot to visualize trends over the years for TV shows and movies
plt.figure(figsize=(12, 8))
sns.lineplot(data=year_type_pivot)
plt.title('Netflix Content Added by Year: TV Shows vs. Movies')
plt.xlabel('Year')
plt.ylabel('Count')
plt.legend(title='Type')
plt.show()

# Stacked bar plot to show the focus on TV shows vs. movies over the years
plt.figure(figsize=(12, 8))
year_type_pivot.plot(kind='bar', stacked=True, ax=plt.gca())
plt.title('Netflix Content Focus: TV Shows vs. Movies (Stacked)')
plt.xlabel('Year')
plt.ylabel('Count')
plt.show()
```



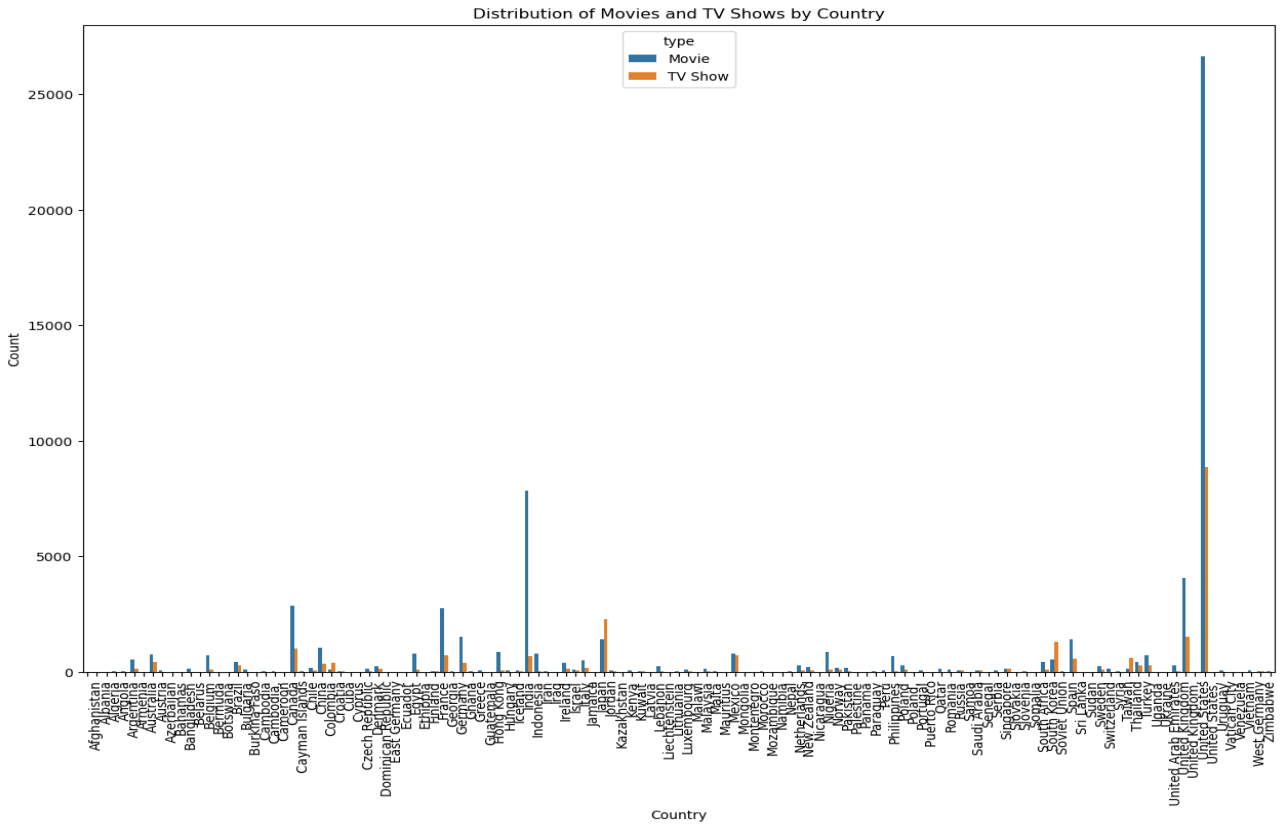


FINDING :

- Maybe since television series take a long time to finish and movies get over within 2 hours on an average, number of movies are comparatively much higher than tv shows.

6. Understanding what content is available in different countries

```
# Group by 'Country' to count the number of movies and TV shows
country_type_counts = df.groupby(['country', 'type']).size().reset_index(name='Count')

# Group by 'Country' and 'Genre' to count different types of content
country_genre_counts = df.groupby(['country',
'listed_in']).size().reset_index(name='Count')

# Visualize the distribution of content across countries (bar plot)
plt.figure(figsize=(15, 10))
sns.barplot(x='country', y='Count', hue='type', data=country_type_counts)
plt.title('Distribution of Movies and TV Shows by Country')
plt.xlabel('Country')
plt.ylabel('Count')
plt.xticks(rotation=90)
plt.show()

# Stacked bar plot to show the breakdown of content types by country
plt.figure(figsize=(15, 10))
sns.barplot(x='country', y='Count', data=country_genre_counts, hue='listed_in')
plt.title('Genres Available by Country')
plt.xlabel('Country')
plt.ylabel('Count')
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```



7. What type of content is available in different countries?

```
df['country'] = df['country'].str.split(', ')
df = df.explode('country')

# Group the data by 'Country' and 'Type' to get counts
country_type_counts = df.groupby(['country', 'type']).size().reset_index(name='count')

# Visualize the data: Stacked bar plot to show counts of Movies and TV Shows by country
plt.figure(figsize=(15, 10))
sns.barplot(x='country', y='count', hue='type', data=country_type_counts, dodge=True)
plt.title('Movies and TV Shows by Country')
plt.xlabel('Country')
plt.ylabel('Count')
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```