NETFLIX BUSINESS CASESTUDY

In [1]:

```python
# IMPORTING THE IMPORTANT LIBRARIES

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```python
# importing the dataset

df = pd.read_csv('netflix.csv')
```

In [3]:

```python
# first five rows of data

df.head()
```

Out[3]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to train I...

Understanding the dataset using following operations

In [5]:
```
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
```

In [6]:
```
df.describe()
```

Out[6]:

	release_year
count	8807.000000
mean	2014.180198
std	8.819312
min	1925.000000
25%	2013.000000
50%	2017.000000
75%	2019.000000
max	2021.000000

In [7]:
```
df.shape
```

Out[7]:
```
(8807, 12)
```

We see that there are total 12 columns and 8807 rows in the dataset. The dataset is about the information related to movies and tvshows.

Data Cleaning

First of all we need to check for the null values present in our dataset and we need to perform some operations on them to remove those null values or replace with another values.

In [8]:

```
#Checking columns with null values

df.isnull().any()
```

Out[8]:

```
show_id         False
type            False
title           False
director         True
cast             True
country          True
date_added       True
release_year    False
rating           True
duration         True
listed_in       False
description     False
dtype: bool
```

As we can observe that following Columns contains null values: director, cast, date_added, rating, duration

In [9]:

```
# total number null values in each column

df.T.apply(lambda x: x.isnull().sum(), axis = 1)
```

Out[9]:

```
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
```

Most number of null values are in director column and least in duration column

We need to handle these null values.

In [19]:
```
# Replacing null values with NA and the columns having very less null values are directly droped

df.director.fillna("NA", inplace=True)
df.cast.fillna("NA", inplace=True)
df.country.fillna("NA", inplace=True)
df.dropna(subset=["date_added", "rating","duration"], inplace=True)
```

In [21]:
```
# Checking for the null values are removed or not

df.isnull().any()
```

Out[21]:
```
show_id         False
type            False
title           False
director        False
cast            False
country         False
date_added      False
release_year    False
rating          False
duration        False
listed_in       False
description     False
dtype: bool
```

There are no null values present in our dataset

In [22]:
```
df.describe()
```

Out[22]:

	release_year
count	8790.000000
mean	2014.183163
std	8.825466
min	1925.000000
25%	2013.000000
50%	2017.000000
75%	2019.000000
max	2021.000000

In [23]:
```
# Looking first five rows of data
df.head()
```

Out[23]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NA	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NA	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NA	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	NA	NA	NA	September 24, 2021	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	NA	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to train I...

Unnesting of the data

We need to unnest the data where multiple values separated by comma are present. As we can see these kind of data is present in few columns such as director, cast, country, listed_in

In [28]:

```
# Unnesting values of director column

split_director = df.assign(director = df.director.str.split(', ')).explode('director').reset_index()
director_new = split_director[['title','director']]
director_new.head()
```

Out[28]:

	title	director
0	Dick Johnson Is Dead	Kirsten Johnson
1	Blood & Water	NA
2	Ganglands	Julien Leclercq
3	Jailbirds New Orleans	NA
4	Kota Factory	NA

In [29]:

```
# Unnesting values of cast column

split_cast = df.assign(cast= df.cast.str.split(', ')).explode('cast').reset_index()
cast_new = split_cast[['title','cast']]
cast_new.head()
```

Out[29]:

	title	cast
0	Dick Johnson Is Dead	NA
1	Blood & Water	Ama Qamata
2	Blood & Water	Khosi Ngema
3	Blood & Water	Gail Mabalane
4	Blood & Water	Thabang Molaba

In [30]:

```
# Unnesting values of country column

split_country = df.assign(country = df.country.str.split(', ')).explode('country').reset_index()
country_new = split_country[['title','country']]
country_new.head()
```

Out[30]:

	title	country
0	Dick Johnson Is Dead	United States
1	Blood & Water	South Africa
2	Ganglands	NA
3	Jailbirds New Orleans	NA
4	Kota Factory	India

In [31]:

```
# Unnesting values of listed_in column

split_listedin = df.assign(listed_in = df.listed_in.str.split(', ')).explode('listed_in').reset_index()
listed_in_new = split_listedin[['title','listed_in']]
listed_in_new.head()
```

Out[31]:

	title	listed_in
0	Dick Johnson Is Dead	Documentaries
1	Blood & Water	International TV Shows
2	Blood & Water	TV Dramas
3	Blood & Water	TV Mysteries
4	Ganglands	Crime TV Shows

In [40]:

```
df_merge1 = cast_new.merge(director_new,on=['title'],how='inner')
df_merge2 = df_merge1.merge(country_new,on=['title'],how='inner')
df_merge3 = df_merge2.merge(listed_in_new,on=['title'],how='inner')
df_merge3.head()
```

Out[40]:

	title	cast	director	country	listed_in
0	Dick Johnson Is Dead	NA	Kirsten Johnson	United States	Documentaries
1	Blood & Water	Ama Qamata	NA	South Africa	International TV Shows
2	Blood & Water	Ama Qamata	NA	South Africa	TV Dramas
3	Blood & Water	Ama Qamata	NA	South Africa	TV Mysteries
4	Blood & Water	Khosi Ngema	NA	South Africa	International TV Shows

In [160]:

```
# Now we need to merge unnested data with the original data
df_final=df_merge3.merge(df[['show_id', 'type', 'title', 'date_added',
       'release_year', 'rating', 'duration']],on=['title'],how='left')
df_final.head()
```

Out[160]:

	title	cast	director	country	listed_in	show_id	type	date_added	release_year	rating	duration
0	Dick Johnson Is Dead	NA	Kirsten Johnson	United States	Documentaries	s1	Movie	September 25, 2021	2020	PG-13	90 min
1	Blood & Water	Ama Qamata	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	Blood & Water	Ama Qamata	NA	South Africa	TV Dramas	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	Blood & Water	Ama Qamata	NA	South Africa	TV Mysteries	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	Blood & Water	Khosi Ngema	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons

In [43]:
```
df_final
```

Out[43]:

	title	cast	director	country	listed_in	show_id	type	date_added	release_year	rating	duration
0	Dick Johnson Is Dead	NA	Kirsten Johnson	United States	Documentaries	s1	Movie	September 25, 2021	2020	PG-13	90 min
1	Blood & Water	Ama Qamata	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	Blood & Water	Ama Qamata	NA	South Africa	TV Dramas	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	Blood & Water	Ama Qamata	NA	South Africa	TV Mysteries	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	Blood & Water	Khosi Ngema	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
...	...	...	...	...	...	...	...	...	...	...	...
201758	Zubaan	Anita Shabdish	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201759	Zubaan	Anita Shabdish	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201760	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Dramas	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201761	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201762	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min

201763 rows × 11 columns

In [45]:

```
# df_final is our final dataset after the unnesting and cleaning, now we are checking the
# duplicate values present in dataset

df_final[df_final.duplicated()]
```

Out[45]:

	title	cast	director	country	listed_in	show_id	type	date_added	release_year	rating	duration
39336	Rust Creek	Micah Hauptman	Jen McGowan	United States	Thrillers	s1632	Movie	November 30, 2020	2018	R	108 min
88474	Blood Will Tell	Oscar Martínez	Miguel Cohan	Argentina	Dramas	s3719	Movie	June 21, 2019	2019	TV-MA	113 min
88475	Blood Will Tell	Oscar Martínez	Miguel Cohan	Argentina	Independent Movies	s3719	Movie	June 21, 2019	2019	TV-MA	113 min
88476	Blood Will Tell	Oscar Martínez	Miguel Cohan	Argentina	International Movies	s3719	Movie	June 21, 2019	2019	TV-MA	113 min
88477	Blood Will Tell	Oscar Martínez	Miguel Cohan	United States	Dramas	s3719	Movie	June 21, 2019	2019	TV-MA	113 min
88478	Blood Will Tell	Oscar Martínez	Miguel Cohan	United States	Independent Movies	s3719	Movie	June 21, 2019	2019	TV-MA	113 min
88479	Blood Will Tell	Oscar Martínez	Miguel Cohan	United States	International Movies	s3719	Movie	June 21, 2019	2019	TV-MA	113 min
88486	Blood Will Tell	Dolores Fonzi	Miguel	Argentina	Dramas	s3719	Movie	June 21, 2019	2019	TV-	113 min

All of the above values are duplicates, so we are going to remove those from our dataset

In [46]:
```
df_final = df_final.drop_duplicates().reset_index()
df_final
```

Out[46]:

	index	title	cast	director	country	listed_in	show_id	type	date_added	release_year	rating	duration
0	0	Dick Johnson Is Dead	NA	Kirsten Johnson	United States	Documentaries	s1	Movie	September 25, 2021	2020	PG-13	90 min
1	1	Blood & Water	Ama Qamata	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	2	Blood & Water	Ama Qamata	NA	South Africa	TV Dramas	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	3	Blood & Water	Ama Qamata	NA	South Africa	TV Mysteries	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	4	Blood & Water	Khosi Ngema	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
...	...	...	...	...	...	...	...	...	...	...	...	...
201703	201758	Zubaan	Anita Shabdish	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201704	201759	Zubaan	Anita Shabdish	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201705	201760	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Dramas	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201706	201761	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201707	201762	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min

201708 rows × 12 columns

In [47]:
```
df_final.T.apply(lambda x: x.isnull().sum(), axis = 1)
```

Out[47]:
```
index           0
title           0
cast            0
director        0
country         0
listed_in       0
show_id         0
type            0
date_added      0
release_year    0
rating          0
duration        0
dtype: int64
```

In [48]:
```
df_final.shape
```

Out[48]:
```
(201708, 12)
```

1.Find the counts of each categorical variable both using graphical and non-graphical analysis.

In [53]:
```
# We are going to get the count for each categorical variable
# First is type Using non-graphical analysis

df_final.groupby(['type']).agg({"title":"nunique"})
```

Out[53]:

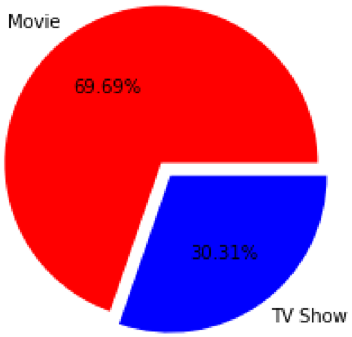
	title
type	
Movie	6126
TV Show	2664

As we can see there are more number of movies listed on netflix than TV show

In [71]:

```
# Graphical analysis

df_type=df_final.groupby(['type']).agg({"title":"nunique"}).reset_index()
plt.pie(df_type['title'], labels=df_type['type'],colors=['red','blue'],autopct='%.2f%%', explode = (0.05,0.05))
plt.show()
```



In [90]:

```
# Analysis of listed_in column.
# It is nothing but the genre of movies and tv shows
# Non-graphical analysis

df_final.groupby(['listed_in']).agg({"title":"nunique"}).sort_values(by=['title'], ascending=False)
```

Out[90]:

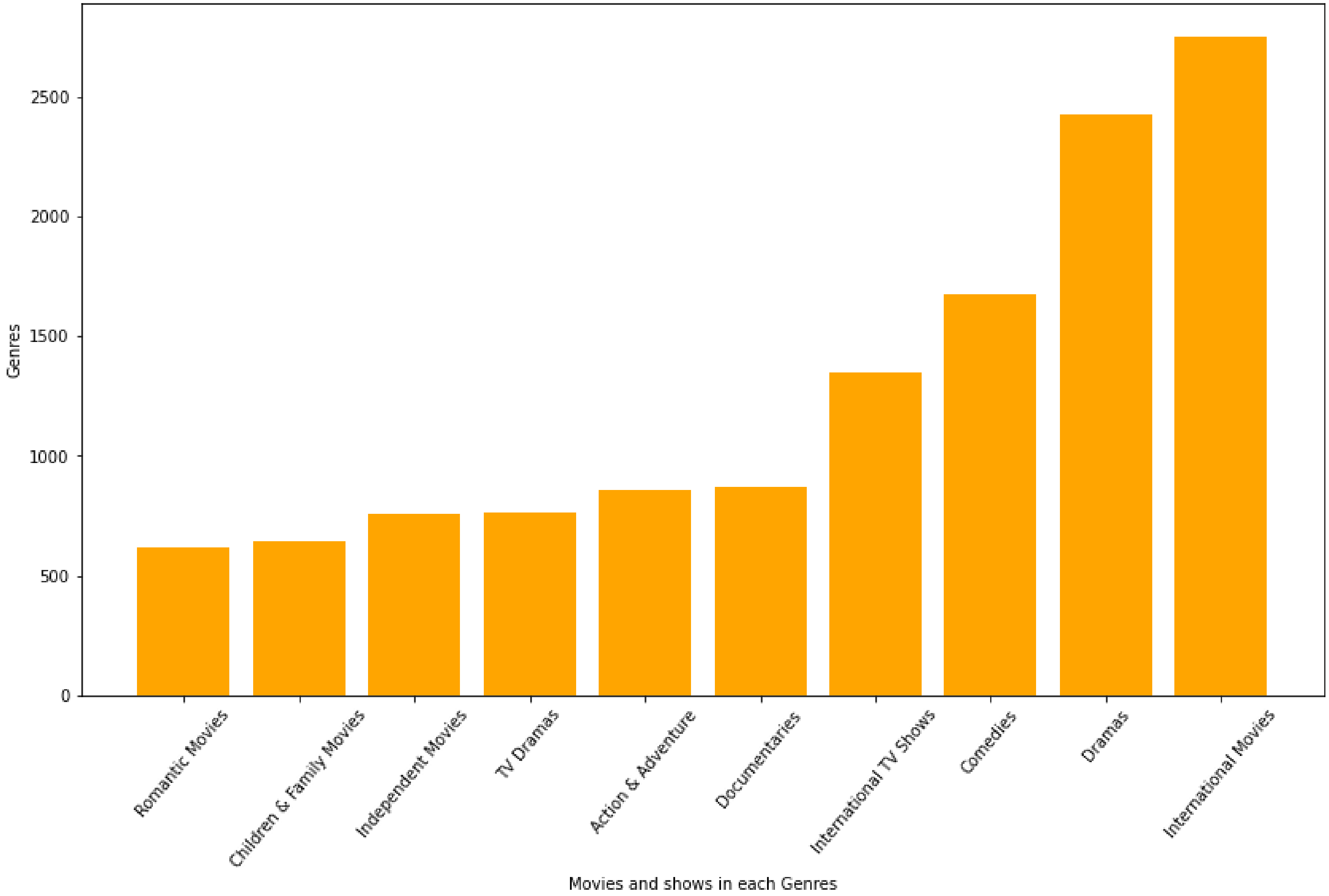
	title
listed_in	
International Movies	2752
Dramas	2426
Comedies	1674
International TV Shows	1349
Documentaries	869
Action & Adventure	859
TV Dramas	762
Independent Movies	756
Children & Family Movies	641
Romantic Movies	616
Thrillers	577
TV Comedies	573
Crime TV Shows	469
Kids' TV	448
Docuseries	394
Music & Musicals	375
Romantic TV Shows	370
Horror Movies	357
Stand-Up Comedy	343
Reality TV	255
British TV Shows	252
Sci-Fi & Fantasy	243
Sports Movies	219
Anime Series	174
Spanish-Language TV Shows	173
TV Action & Adventure	167
Korean TV Shows	151
Classic Movies	116
LGBTQ Movies	102

	title
listed_in	
TV Mysteries	98
Science & Nature TV	92
TV Sci-Fi & Fantasy	83
TV Horror	75
Anime Features	71
Cult Movies	71
Teen TV Shows	69
Faith & Spirituality	65
TV Thrillers	57
Stand-Up Comedy & Talk Shows	56
Movies	53
Classic & Cult TV	26
TV Shows	16

As per the result most of the genre is of international movies category, followed by Dramas, Comedies, International TV Shows, Documentaries, etc.

In [99]:
```
# Graphical Analysis

df_listedin = df_final.groupby(['listed_in']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'], ascending=Fa
plt.figure(figsize=(14,8))
plt.bar(df_listedin[::-1]['listed_in'], df_listedin[::-1]['title'],color=['orange'])
plt.xticks(rotation=50)
plt.xlabel('Movies and shows in each Genres')
plt.ylabel('Genres')
plt.show()
```



```
In [106]: # Non-graphical analysis of country column

          df_final.groupby(['country']).agg({"title":"nunique"})
```

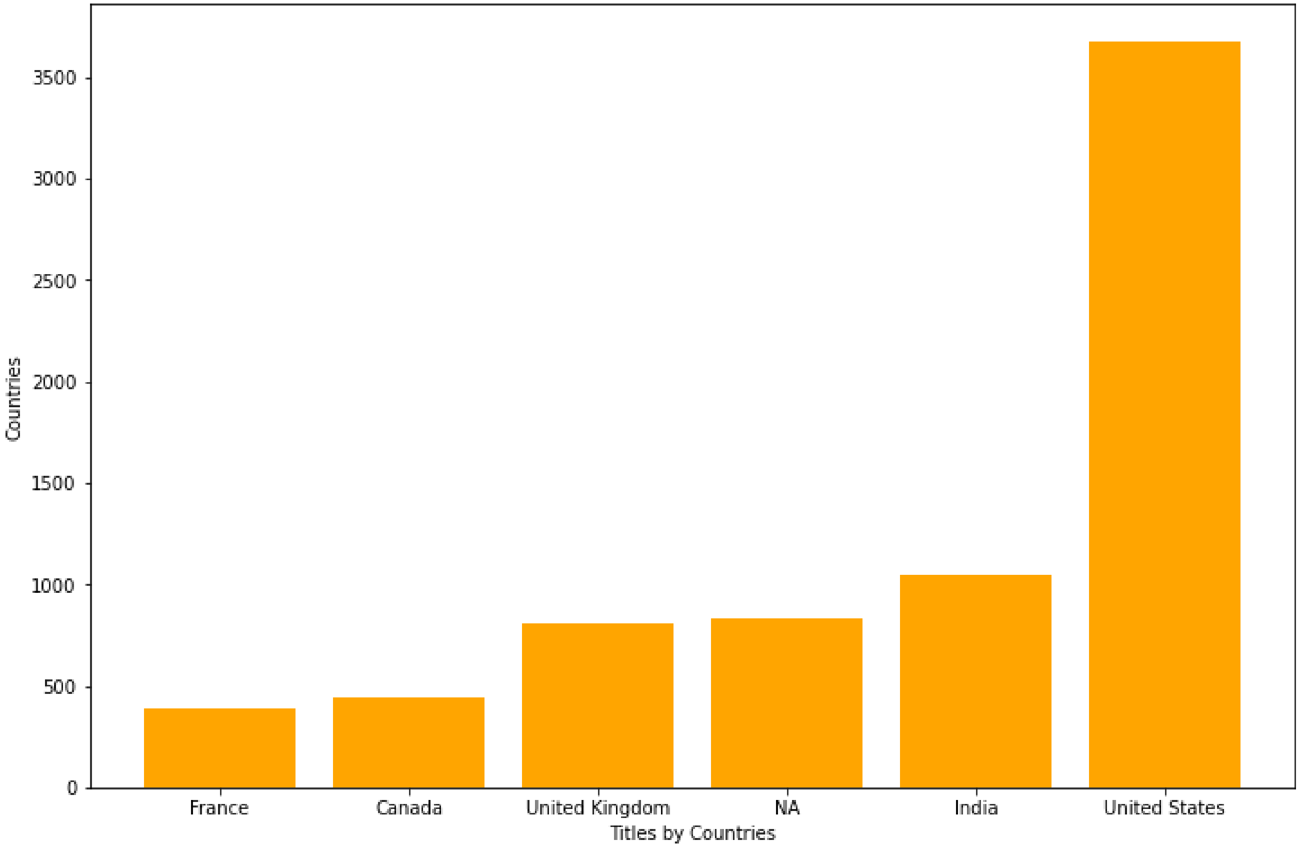
```
Out[106]:
                     title
              country
                         2
          Afghanistan    1
              Albania    1
              Algeria    3
               Angola    1
                  ...  ...
         Vatican City    1
            Venezuela    4
              Vietnam    7
         West Germany    5
             Zimbabwe    3

128 rows × 1 columns
```

In [128]:

```
# Graphical Analysis for country column

df_country=df_final.groupby(['country']).agg({"title":"nunique"}).reset_index().sort_values(by=['title'],ascending=False)[:
plt.figure(figsize=(12,8))
plt.bar(df_country[::-1]['country'], df_country[::-1]['title'],color=['orange'])
plt.xlabel('Titles by Countries')
plt.ylabel('Countries')
plt.show()
```



United States produced the most number of movies and tv shows as per the data, followed by India Canada, and France

In [111]:
```
df_final
```

Out[111]:

	index	title	cast	director	country	listed_in	show_id	type	date_added	release_year	rating	duration
0	0	Dick Johnson Is Dead	NA	Kirsten Johnson	United States	Documentaries	s1	Movie	September 25, 2021	2020	PG-13	90 min
1	1	Blood & Water	Ama Qamata	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	2	Blood & Water	Ama Qamata	NA	South Africa	TV Dramas	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	3	Blood & Water	Ama Qamata	NA	South Africa	TV Mysteries	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	4	Blood & Water	Khosi Ngema	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
...	...	...	...	...	...	...	...	...	...	...	...	...
201703	201758	Zubaan	Anita Shabdish	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201704	201759	Zubaan	Anita Shabdish	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201705	201760	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Dramas	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201706	201761	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201707	201762	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min

201708 rows × 12 columns

In [134]:

```
# Non-graphical analysis for the date of released of movies and tv shows

df_final.groupby(['release_year']).agg({"title" : "nunique"}).sort_values(by = ['title'], ascending = False)[:10]
```

Out[134]:

```
              title
release_year
        2018   1146
        2019   1030
        2017   1030
        2020    953
        2016    901
        2021    592
        2015    555
        2014    352
        2013    286
        2012    236
```

In [114]:
```
df_final
```

Out[114]:

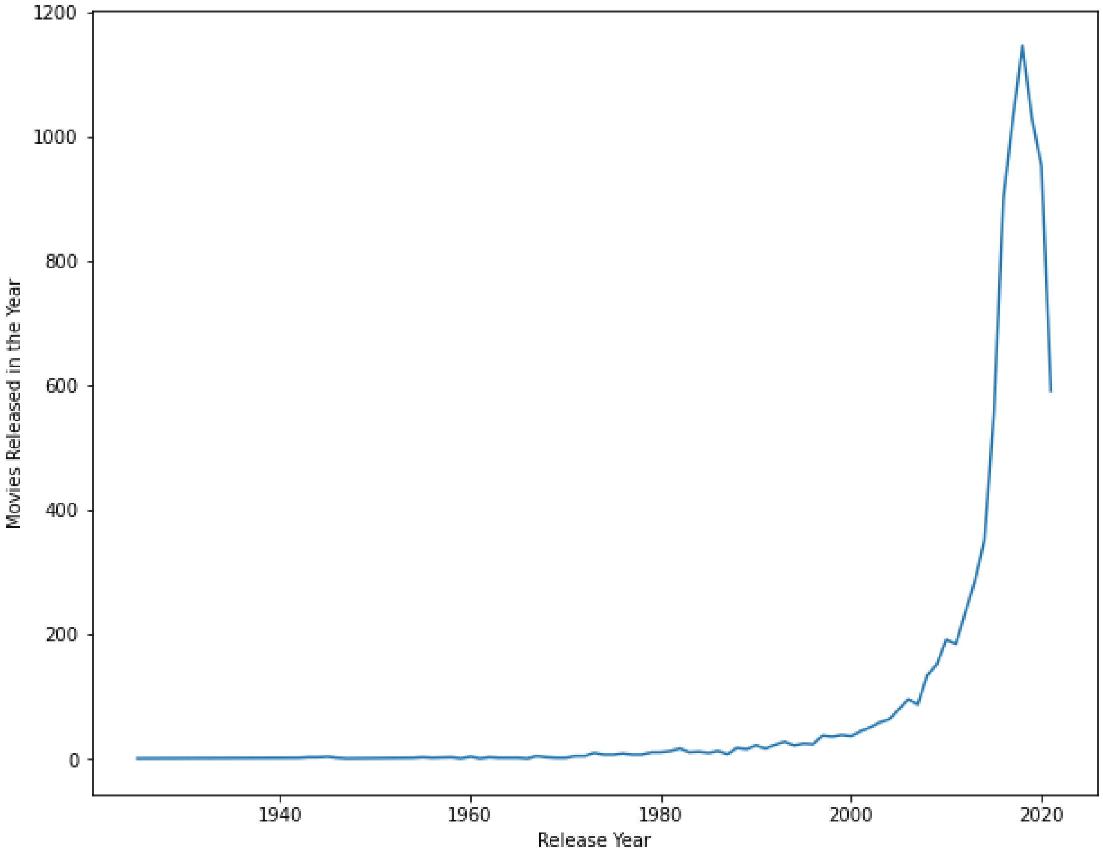
	index	title	cast	director	country	listed_in	show_id	type	date_added	release_year	rating	duration
0	0	Dick Johnson Is Dead	NA	Kirsten Johnson	United States	Documentaries	s1	Movie	September 25, 2021	2020	PG-13	90 min
1	1	Blood & Water	Ama Qamata	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
2	2	Blood & Water	Ama Qamata	NA	South Africa	TV Dramas	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
3	3	Blood & Water	Ama Qamata	NA	South Africa	TV Mysteries	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
4	4	Blood & Water	Khosi Ngema	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons
...	...	...	...	...	...	...	...	...	...	...	...	...
201703	201758	Zubaan	Anita Shabdish	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201704	201759	Zubaan	Anita Shabdish	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201705	201760	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Dramas	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201706	201761	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	International Movies	s8807	Movie	March 2, 2019	2015	TV-14	111 min
201707	201762	Zubaan	Chittaranjan Tripathy	Mozez Singh	India	Music & Musicals	s8807	Movie	March 2, 2019	2015	TV-14	111 min

201708 rows × 12 columns

Most of the movies and tv shows which were added to netflix are released in the year 2018 followed by 2019, 2017, 2020, 2016, etc.

```
In [162]: # Graphical analysis of release_year column

          df_release_year = df_final.groupby(['release_year']).agg({"title" : "nunique"}).reset_index()
          plt.figure(figsize=(10,8))
          sns.lineplot(data=df_release_year, x='release_year', y='title')
          plt.ylabel("Movies Released in the Year")
          plt.xlabel("Release Year")
          plt.show()
```



As the above graph shows that the movies and tv shows present on Netflix platform are mostly release in the year 2010 to 2021

In [163]:

```
# Non-graphical analysis for the date_added column
# This column contains the dates of the movies and tv shows which were uploaded on Netflix
# platform for viewers

# Now before doing the analysis on this, we need to convert the date values to standard
# format to do our analysis

df_final["new_formatted_date"] = pd.to_datetime(df_final["date_added"])
df_final['month_added']=df_final['new_formatted_date'].dt.month
df_final['week_Added']=df_final['new_formatted_date'].dt.week
df_final['year']=df_final['new_formatted_date'].dt.year
df_final.head()
```

```
<ipython-input-163-6bd15a189600>:10: FutureWarning: Series.dt.weekofyear and Series.dt.week have been deprecated.  Please
use Series.dt.isocalendar().week instead.
  df_final['week_Added']=df_final['new_formatted_date'].dt.week
```

Out[163]:

	title	cast	director	country	listed_in	show_id	type	date_added	release_year	rating	duration	new_formatted_date	month_added	we
0	Dick Johnson Is Dead	NA	Kirsten Johnson	United States	Documentaries	s1	Movie	September 25, 2021	2020	PG-13	90 min	2021-09-25	9	
1	Blood & Water	Ama Qamata	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
2	Blood & Water	Ama Qamata	NA	South Africa	TV Dramas	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
3	Blood & Water	Ama Qamata	NA	South Africa	TV Mysteries	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	
4	Blood & Water	Khosi Ngema	NA	South Africa	International TV Shows	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	2021-09-24	9	

In [165]:

```
df_final.groupby(['year']).agg({'title':'nunique'}).sort_values(by=['title'],ascending = False)
```

Out[165]:

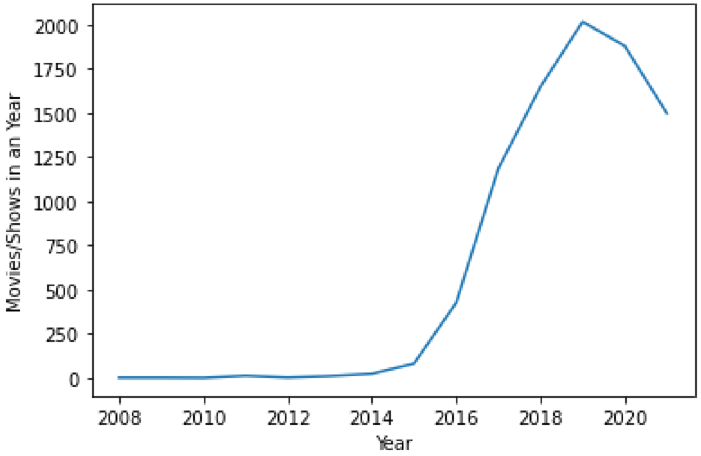
```
      title
year
2019   2016
2020   1879
2018   1648
2021   1498
2017   1185
2016    426
2015     82
2014     24
2011     13
2013     11
2012      3
2008      2
2009      2
2010      1
```

In [166]:

```
# Non-graphical analysis of dated_added column

df_year = df_final.groupby(['year']).agg({"title":"nunique"}).reset_index()

sns.lineplot(data=df_year, x='year', y='title')
plt.ylabel("Movies/Shows in an Year")
plt.xlabel("Year")
plt.show()
```



We can see the trend after year 2016 that number of movies and tv show uploads in increased. Most number of uploads were seen in the year 2019

2. Comparison of tv show vs. movie.

a. Finding the number of movies produced in each country and pick the top 10 countries.

In [168]:

```
# First of all we need to group values base on movie and tv show.

df_movie = df_final[df_final['type'] == 'Movie']
df_tv_show = df_final[df_final['type'] == 'TV Show']
```

In [175]:

```
df_movie.groupby(['country']).agg({'title':'nunique'}).sort_values(by = ['title'], ascending = False)[:11]
```

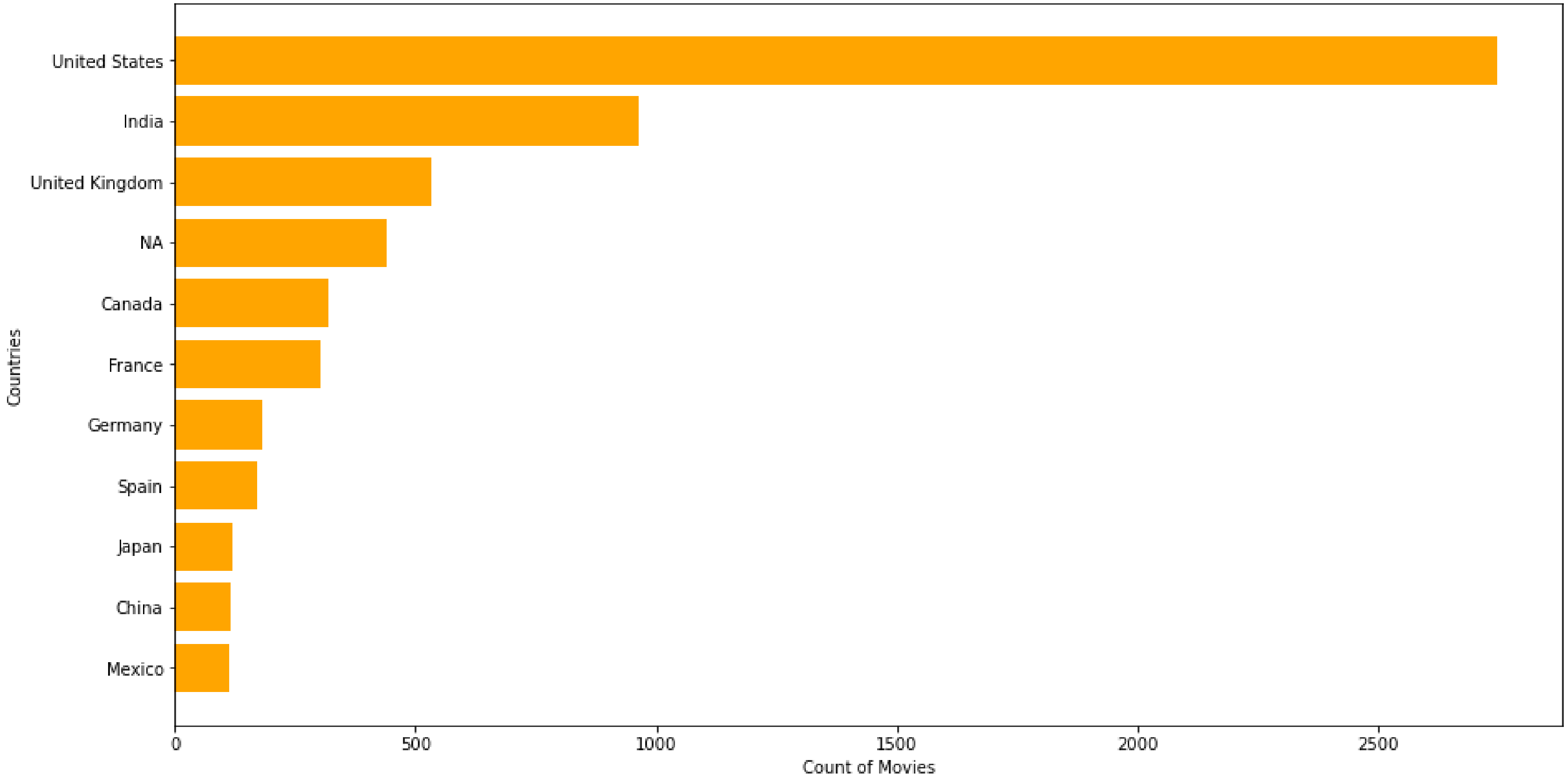
Out[175]:

	title
country	
United States	2748
India	962
United Kingdom	532
NA	439
Canada	319
France	303
Germany	182
Spain	171
Japan	119
China	114
Mexico	111

In [180]:

```
df_country = df_movie.groupby(['country']).agg({'title':'nunique'}).reset_index().sort_values(by = ['title'], ascending = F

plt.figure(figsize=(15,8))
plt.barh(df_country[::-1]['country'], df_country[::-1]['title'],color=['orange'])
plt.xlabel('Count of Movies')
plt.ylabel('Countries')
plt.show()
```



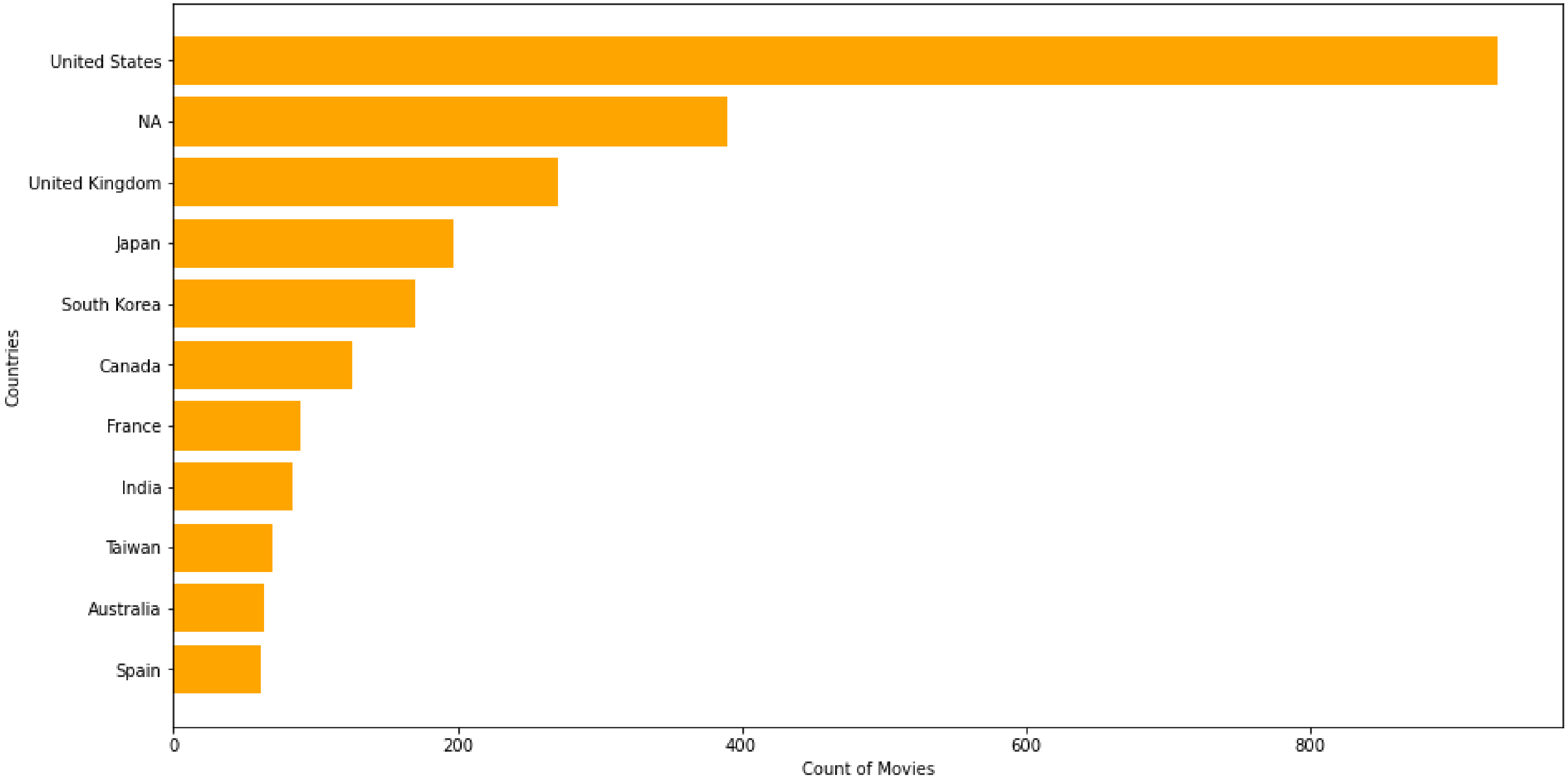
The most number of movies are made in United States, followed by India, UK, Canada.

```
b. Finding the number of Tv-Shows produced in each country and pick the top 10
countries.
```

```
In [181]: df_tv_show.groupby(['country']).agg({'title':'nunique'}).sort_values(by = ['title'], ascending = False)[:11]
```

```
Out[181]:
                      title
             country
      United States    932
                 NA    390
     United Kingdom    271
              Japan    197
        South Korea    170
             Canada    126
             France     90
              India     84
             Taiwan     70
          Australia     64
              Spain     61
```

```
In [188]: df_country = df_tv_show.groupby(['country']).agg({'title':'nunique'}).reset_index().sort_values(by = ['title'], ascending =
          plt.figure(figsize=(15,8))
          plt.barh(df_country[::-1]['country'], df_country[::-1]['title'],color=['orange'])
          plt.xlabel('Count of Movies')
          plt.ylabel('Countries')
          plt.show()
```



In tv shows also USA tops with 932 shows. Followed by UK, Japan, South Korea, Canada

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

a. Finding which is the best week to release the Tv-show or the movie. Do the analysis separately for Tv-shows and Movies

In [194]:
```
# Let's first analyse for movies, in which week most number of movies are released

df_movie.groupby(['week_Added']).agg({'title':'nunique'})
```

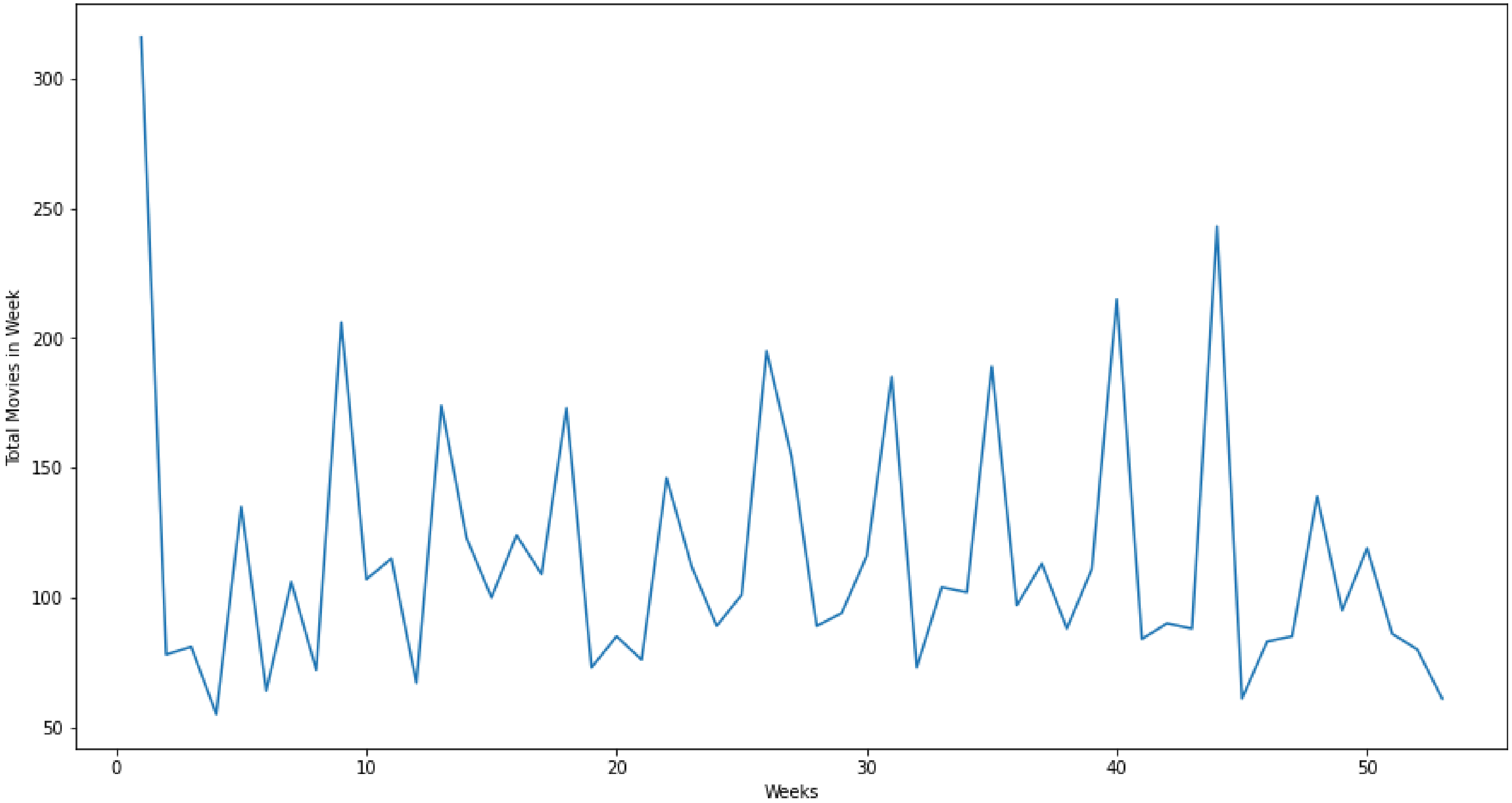
Out[194]:

	title
week_Added	
1	316
2	78
3	81
4	55
5	135
6	64
7	106
8	72
9	206
10	107
11	115
12	67
13	174
14	123
15	100
16	124
17	109
18	173
19	73
20	85
21	76
22	146
23	112
24	89
25	101
26	195
27	154
28	89
29	94

	title
week_Added	
30	116
31	185
32	73
33	104
34	102
35	189
36	97
37	113
38	88
39	111
40	215
41	84
42	90
43	88
44	243
45	61
46	83
47	85
48	139
49	95
50	119
51	86
52	80
53	61

In [195]:

```
df_week=df_movie.groupby(['week_Added']).agg({"title":"nunique"}).reset_index()
plt.figure(figsize=(15,8))
sns.lineplot(data=df_week, x='week_Added', y='title')
plt.ylabel("Total Movies in Week")
plt.xlabel("Weeks")
plt.show()
```



Most of the movies are uploaded on Netflix in the first week after that there are many ups and down in the uploads. And also week number 44 has more number of uploads

In [200]:

```
# Analysis of tv show

df_tv_show.groupby(['week_Added']).agg({'title':'nunique'})
```

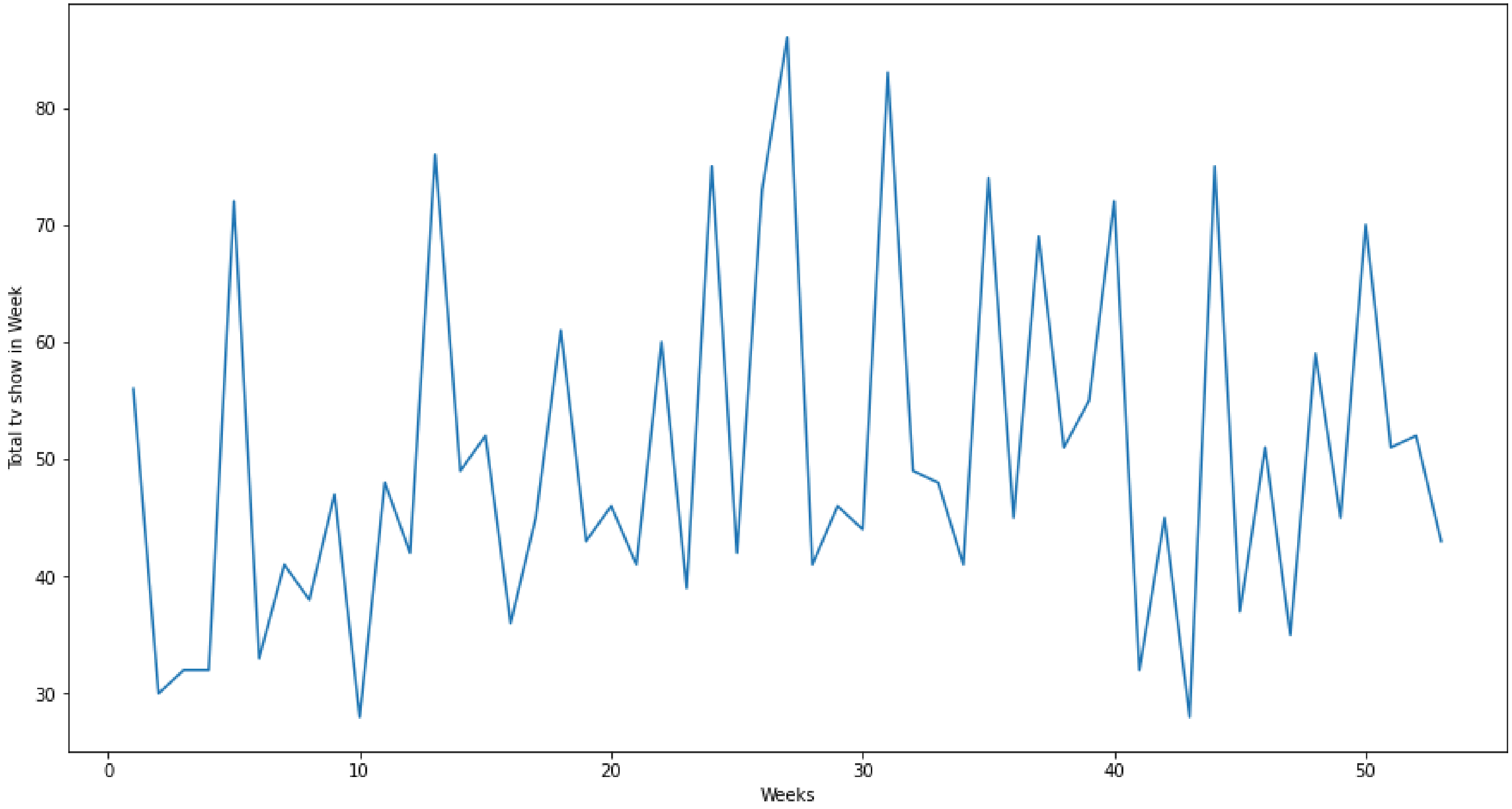
Out[200]:

	title
week_Added	
1	56
2	30
3	32
4	32
5	72
6	33
7	41
8	38
9	47
10	28
11	48
12	42
13	76
14	49
15	52
16	36
17	45
18	61
19	43
20	46
21	41
22	60
23	39
24	75
25	42
26	73
27	86
28	41
29	46

	title
week_Added	
30	44
31	83
32	49
33	48
34	41
35	74
36	45
37	69
38	51
39	55
40	72
41	32
42	45
43	28
44	75
45	37
46	51
47	35
48	59
49	45
50	70
51	51
52	52
53	43

In [198]:

```
df_week=df_tv_show.groupby(['week_Added']).agg({"title":"nunique"}).reset_index()
plt.figure(figsize=(15,8))
sns.lineplot(data=df_week, x='week_Added', y='title')
plt.ylabel("Total tv show in Week")
plt.xlabel("Weeks")
plt.show()
```



In case of tv show the middle weeks of the year has more uploads than starting and end of the year. More number of uploads are in week number 27, 13, 31

b. Finding which is the best month to release the Tv-show or the movie. Do the analysis separately for Tv-shows and Movies

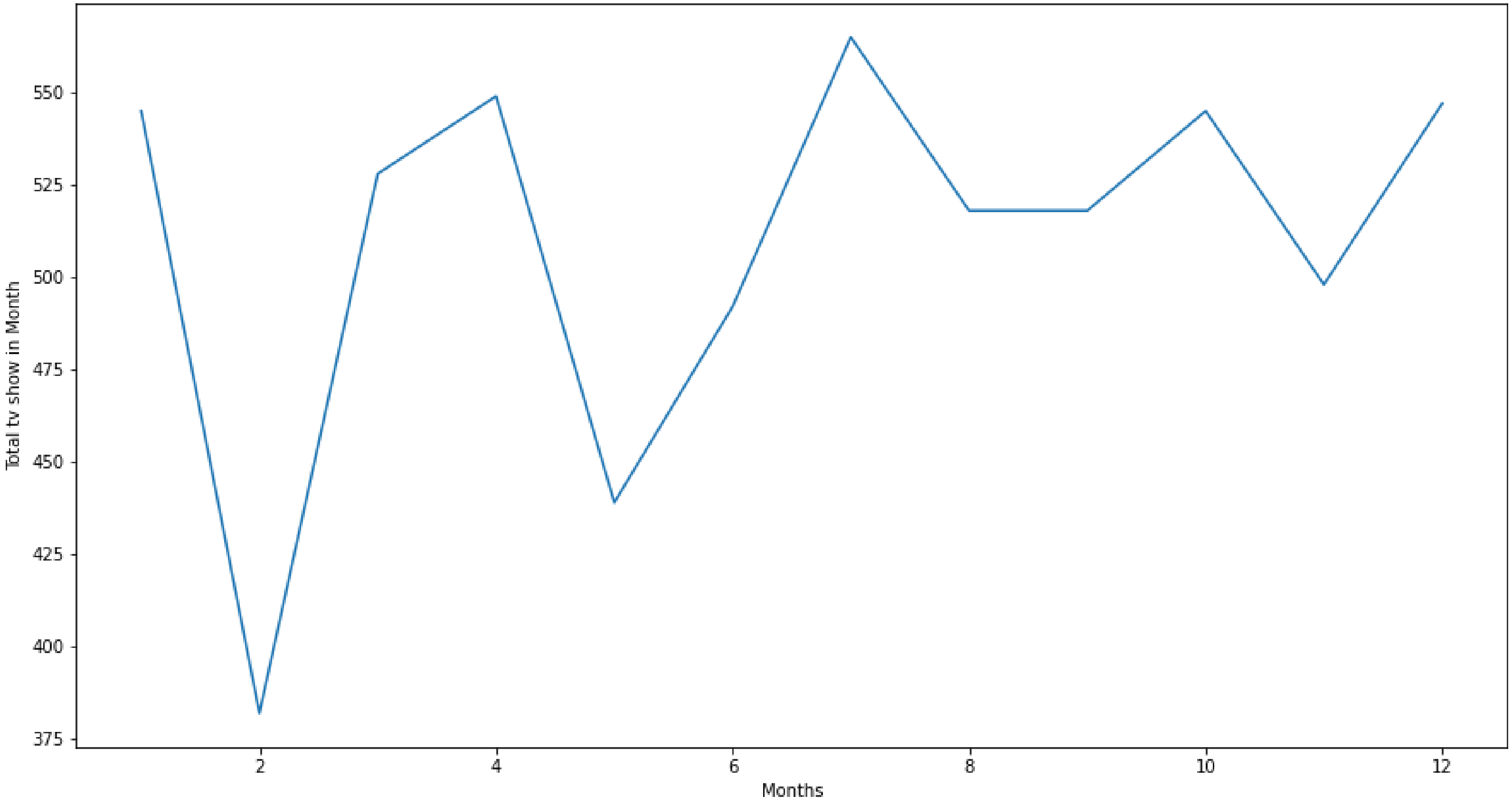
```
In [202]: # Analysis of number of movies uploaded on Netflix by months

          df_movie.groupby(['month_added']).agg({'title':'nunique'})
```

```
Out[202]:
                  title
     month_added
               1    545
               2    382
               3    528
               4    549
               5    439
               6    492
               7    565
               8    518
               9    518
              10    545
              11    498
              12    547
```

In [204]:
```
df_month=df_movie.groupby(['month_added']).agg({"title":"nunique"}).reset_index()

plt.figure(figsize=(15,8))
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("Total tv show in Month")
plt.xlabel("Months")
plt.show()
```



All months have nearly equal number of movie uploads except month number 2 and 5 which have low number of uploads compare to other months. So that will not be movie releasing month.

In [205]:

```
# Analysis of number of tv shows uploaded on Netflix by months

df_tv_show.groupby(['month_added']).agg({'title':'nunique'})
```

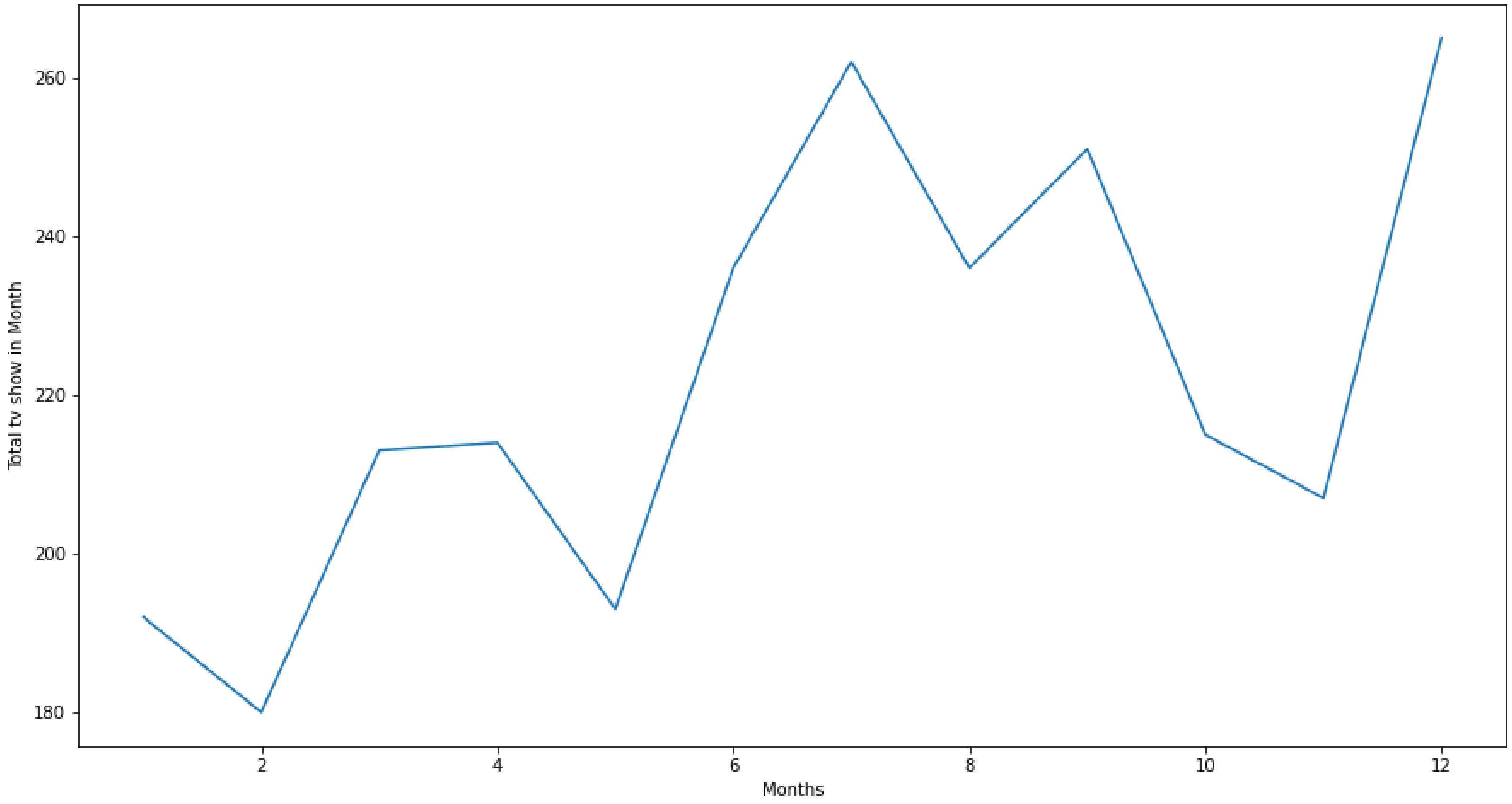
Out[205]:

	title
month_added	
1	192
2	180
3	213
4	214
5	193
6	236
7	262
8	236
9	251
10	215
11	207
12	265

In [206]:

```
df_month = df_tv_show.groupby(['month_added']).agg({"title":"nunique"}).reset_index()

plt.figure(figsize=(15,8))
sns.lineplot(data=df_month, x='month_added', y='title')
plt.ylabel("Total tv show in Month")
plt.xlabel("Months")
plt.show()
```



In case of tv shows the more released are in the month 12th and least released in the month 2nd.

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

In [213]:

```
# Finding the actors which have appeared most in movies and tv shows, which were uploaded on Netflix
# Non-graphical analysis
df_final[df_final['cast']!='NA'].groupby('cast').agg({'title':'nunique'}).sort_values(by = ['title'], ascending = False)[:2
```

Out[213]:

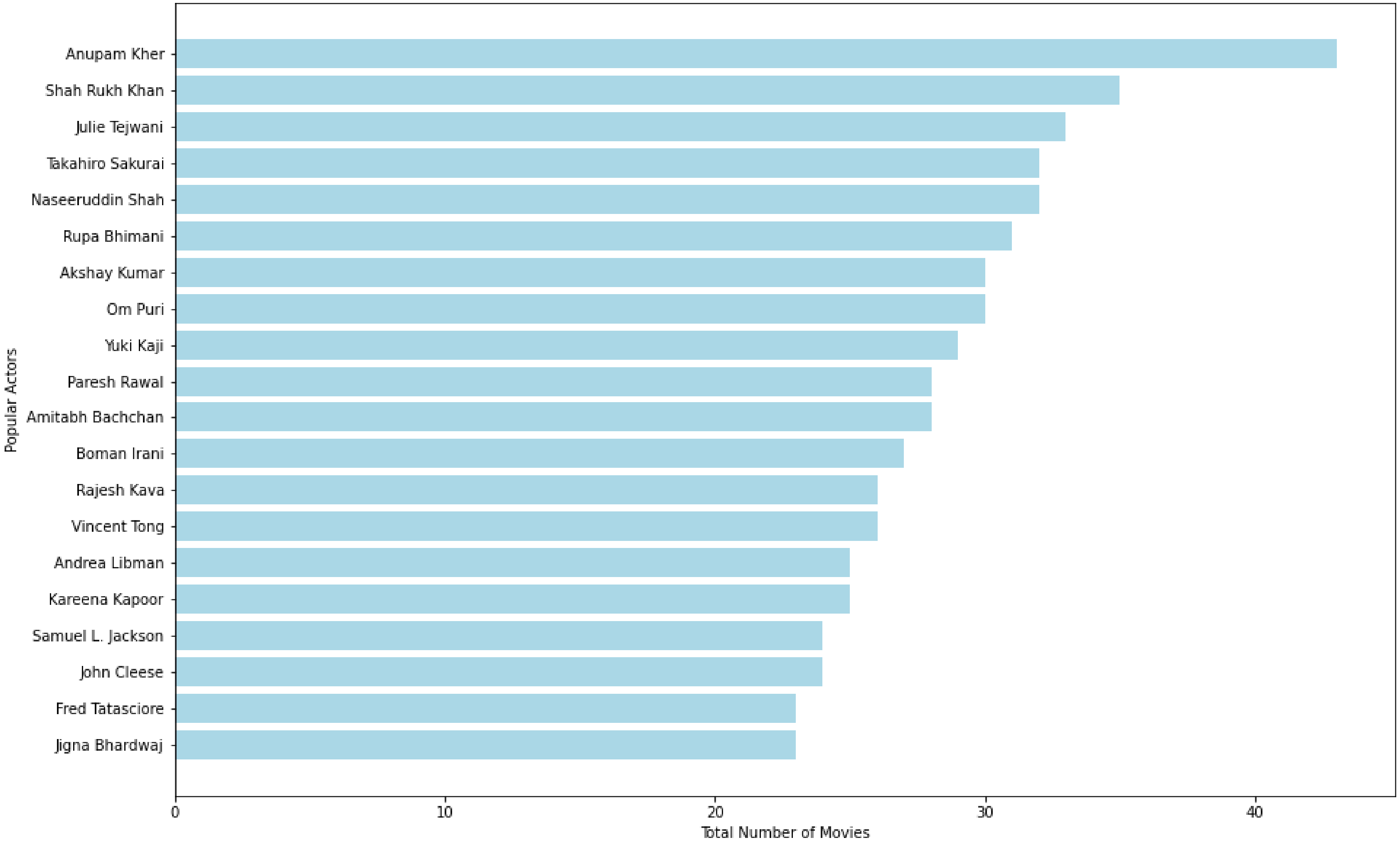
	title
cast	
Anupam Kher	43
Shah Rukh Khan	35
Julie Tejwani	33
Takahiro Sakurai	32
Naseeruddin Shah	32
Rupa Bhimani	31
Akshay Kumar	30
Om Puri	30
Yuki Kaji	29
Paresh Rawal	28
Amitabh Bachchan	28
Boman Irani	27
Rajesh Kava	26
Vincent Tong	26
Andrea Libman	25
Kareena Kapoor	25
Samuel L. Jackson	24
John Cleese	24
Fred Tatasciore	23
Jigna Bhardwaj	23

In [219]:

```
# graphical analysis

df_actor = df_final[df_final['cast']!='NA'].groupby('cast').agg({'title':'nunique'}).reset_index().sort_values(by = ['title

plt.figure(figsize=(15,10))
plt.barh(df_actor[::-1]['cast'], df_actor[::-1]['title'],color=['lightblue'])
plt.xlabel('Total Number of Movies')
plt.ylabel('Popular Actors')
plt.show()
```



Anupam Kher has the most number of movies and tv shows on Netflix platform And in this list most of the actors are from India

In [220]:

```
# Finding the directors which had directed most in movies and tv shows, which were uploaded on Netflix
# Non-graphical analysis

df_final[df_final['director']!='NA'].groupby('director').agg({'title':'nunique'}).sort_values(by = ['title'], ascending = F
```

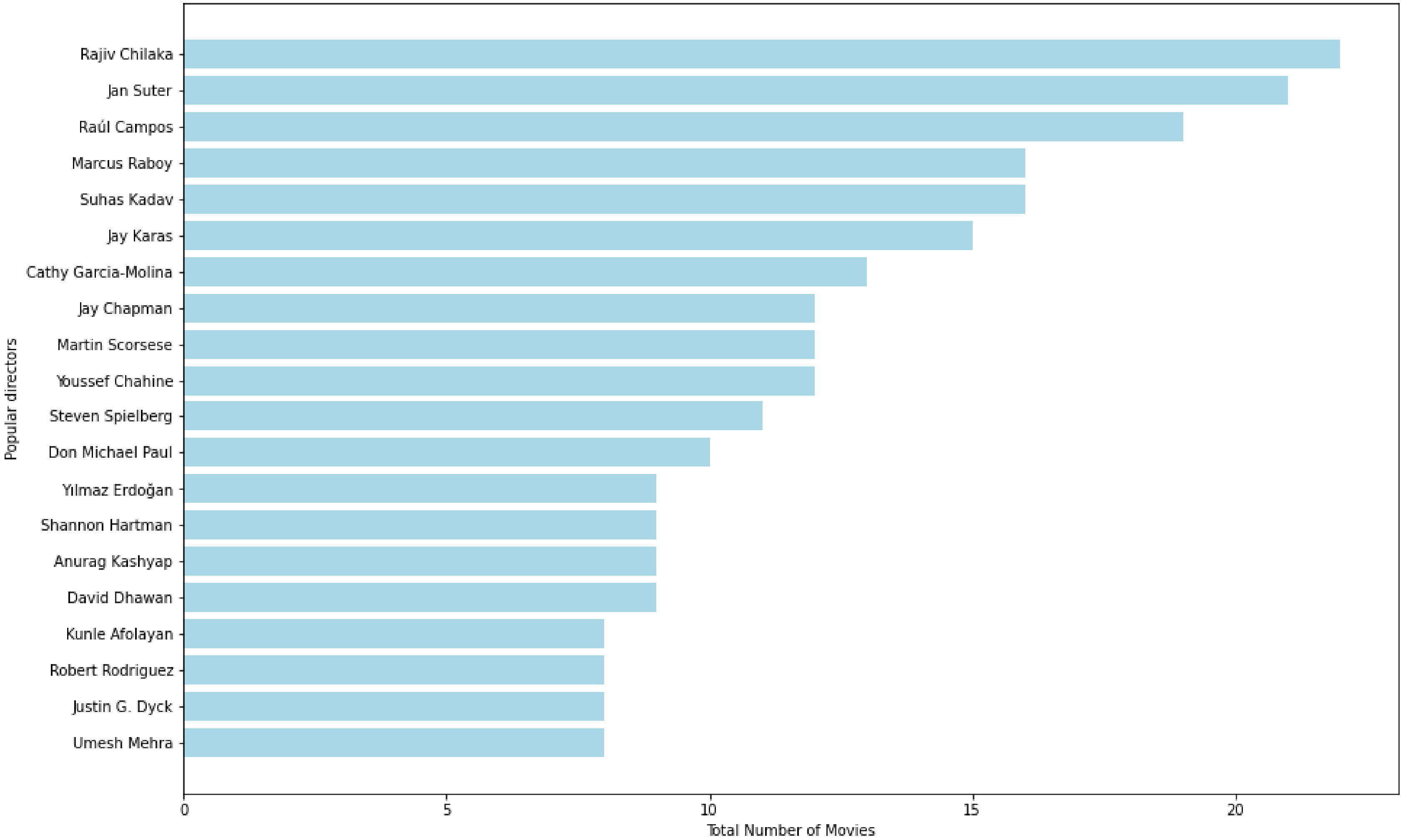
Out[220]:

	title
director	
Rajiv Chilaka	22
Jan Suter	21
Raúl Campos	19
Marcus Raboy	16
Suhas Kadav	16
Jay Karas	15
Cathy Garcia-Molina	13
Jay Chapman	12
Martin Scorsese	12
Youssef Chahine	12
Steven Spielberg	11
Don Michael Paul	10
Yılmaz Erdoğan	9
Shannon Hartman	9
Anurag Kashyap	9
David Dhawan	9
Kunle Afolayan	8
Robert Rodriguez	8
Justin G. Dyck	8
Umesh Mehra	8

```
In [221]: # graphical analysis

          df_director = df_final[df_final['director']!='NA'].groupby('director').agg({'title':'nunique'}).reset_index().sort_values(b

          plt.figure(figsize=(15,10))
          plt.barh(df_director[::-1]['director'], df_director[::-1]['title'],color=['lightblue'])
          plt.xlabel('Total Number of Movies')
          plt.ylabel('Popular directors')
          plt.show()
```



Rajiv Chilaka is the most popular director on Netflix, who directed 22 movies/tv shows which were uploaded on Netflix. In the top 20 directors list many are from India.

5. Which genre movies are more popular or produced more

In [225]:
```
# We have listed_in column, that is nothing but the genre of the movies

df_movie.groupby(['listed_in']).agg({'title':'nunique'}).sort_values(by = ['title'], ascending = False)
```

Out[225]:

	title
listed_in	
International Movies	2752
Dramas	2426
Comedies	1674
Documentaries	869
Action & Adventure	859
Independent Movies	756
Children & Family Movies	641
Romantic Movies	616
Thrillers	577
Music & Musicals	375
Horror Movies	357
Stand-Up Comedy	343
Sci-Fi & Fantasy	243
Sports Movies	219
Classic Movies	116
LGBTQ Movies	102
Cult Movies	71
Anime Features	71
Faith & Spirituality	65
Movies	53

International Movies, Dramas, Comedies, Documentaries, Action & Adventure, these are the popular genres for movies present on Netflix.

6. Finding After how many days the movie will be added to Netflix after the release of the movie

```
In [230]: # Adding the new column which will tell that after how much time the movie or tv show was uploaded
          # on netflix after it's date of release.

          df_final['year_difference'] = df_final['year']-df_final['release_year']
```

```
In [236]: df_final.groupby('year_difference').agg({'title':'nunique'}).sort_values(by = ['title'], ascending = False)[:10]
```

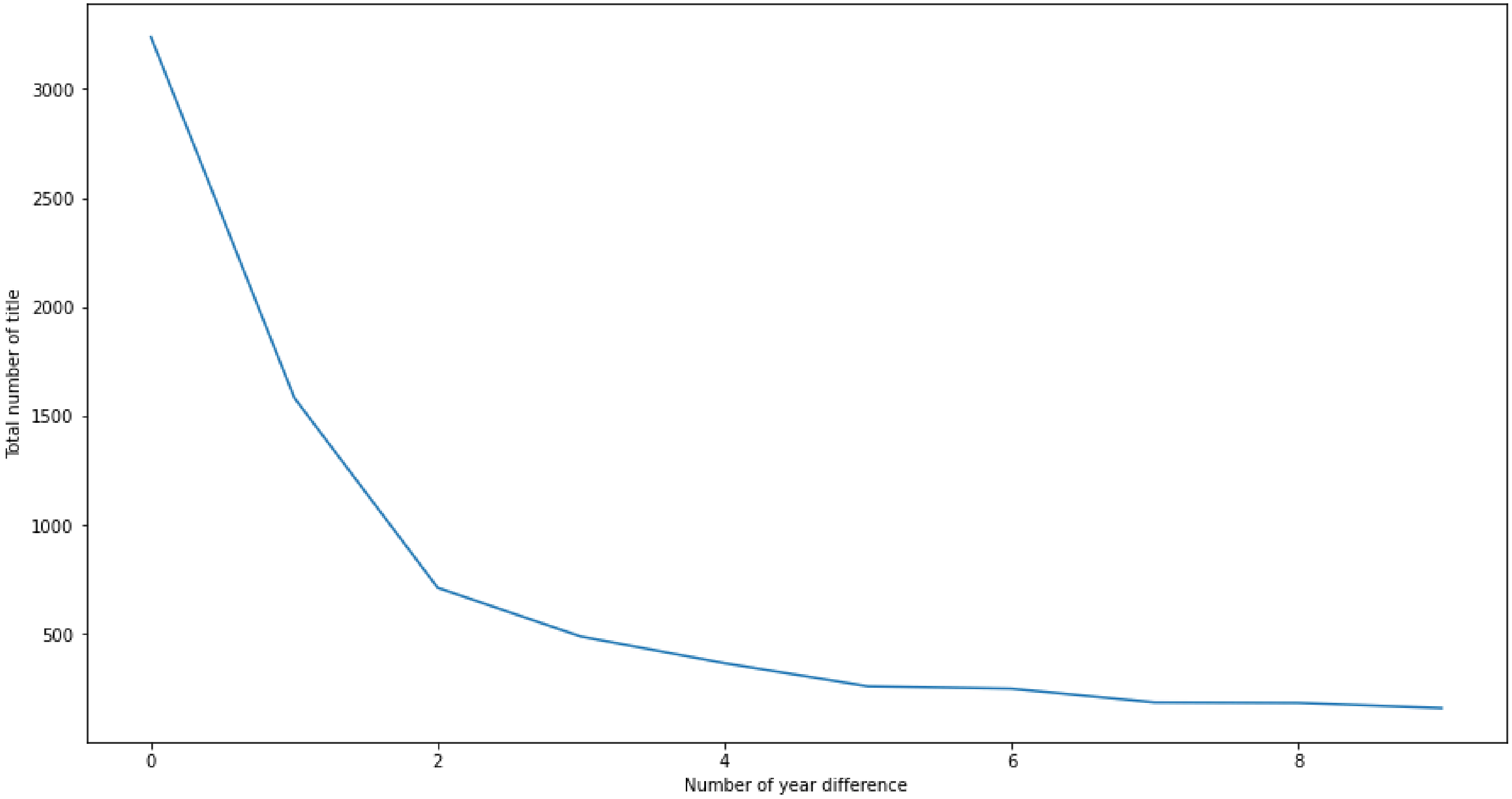
```
Out[236]:
                          title
          year_difference
                        0  3239
                        1  1584
                        2   713
                        3   489
                        4   367
                        5   261
                        6   250
                        7   187
                        8   185
                        9   161
```

In [237]:

```
# Graphical Analysis

df_year_diff =df_final.groupby('year_difference').agg({'title':'nunique'}).reset_index().sort_values(by = ['title'], ascend

plt.figure(figsize=(15,8))
sns.lineplot(data=df_year_diff, x='year_difference', y='title')
plt.ylabel("Total number of title")
plt.xlabel("Number of year difference")
plt.show()
```



```
From the resultant data, we can say that most of the movies and tv_shows which were released were uploaded on Netflix in
that year only.
So we can say that best time to upload movies are within a year of it's realease.
```

Recommendations

1. The most popular genre across Netflix are dramas, comedy, international tv show, documentaries, action and adventure, so the content which being uploaded on Netflix in future should be from these genre which is recommended.
2. The most of the content uploaded on Netflix is from USA, India, UK, Canada.
3. Most of the movies and tv shows which were added to netflix are released in the year 2018 followed by 2019, 2017, 2020, 2016. This shows that the latest movies are in more demand. So the recommendation is that latest movies should be instantly upload on Netflix after the date of released
4. The movies and tv shows present on Netflix platform are mostly released in the year 2010 to 2021. From this we can conclude that old movies are not that much in demand, therefore it is recommended to upload movies which are not that much old.
5. The best time to upload the movies on Netflix is the first week of the year, also first and last month of the year is also good to upload movies and tv shows.
6. The top most actor in Netflix are Anupam Kher, Shahrukh Khan, Akshay Kumar, etc. These are the very famous actor of India. So it is recommended to consider the famous director and actor from their country before uploading the movies or tv shows.
7. The list of uploads of movies and tv shows ranks US as top most position. While India lists 2nd in movies column but at number 7 in tv shows column, therefore it is recommended to add more number of tv shows also from India.

In [ ]: