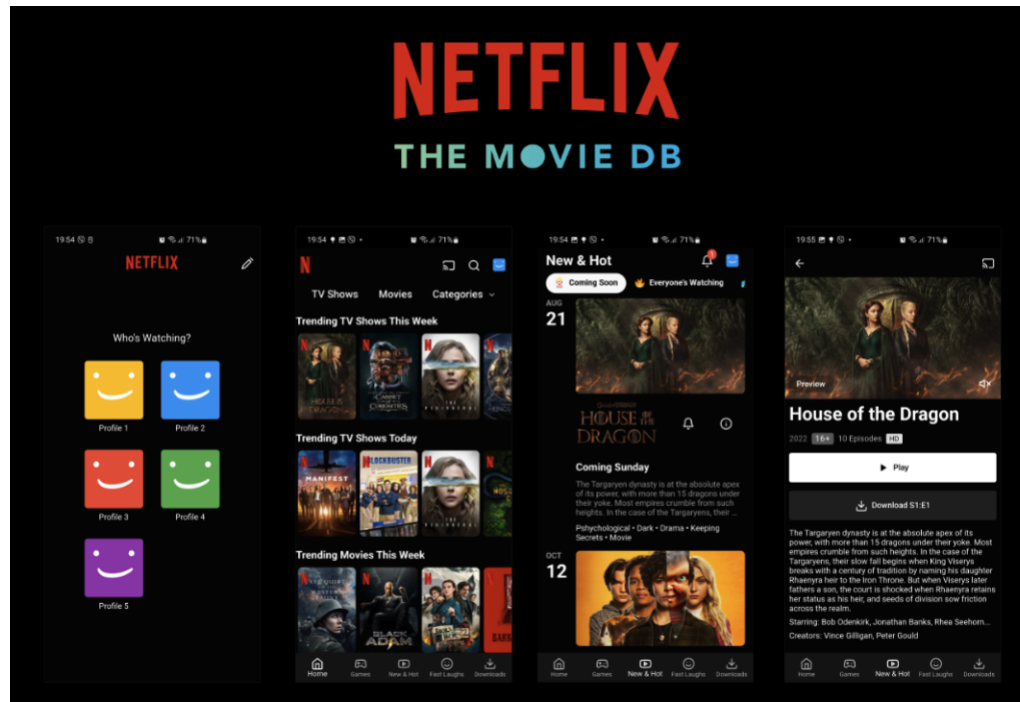


Business Case

Netflix - Data Exploration and Visualisation

Suman Debnath



Introduction

Netflix is one of the most popular media and video streaming platforms. They have over 10000 movies or tv shows available on their platform, as of mid-2021, they have over 222M Subscribers globally. This tabular dataset consists of listings of all the movies and tv shows available on Netflix, along with details such as - cast, directors, ratings, release year, duration, etc.

Business Problem

Analyze the data and generate insights that could help Netflix in deciding which type of shows/movies to produce and how they can grow the business in different countries

Dataset

Link: [Dataset_link](#)

The dataset provided to you consists of a list of all the TV shows/movies available on Netflix:

- Show_id** : Unique ID for every Movie / Tv Show
- Type** : Identifier - A Movie or TV Show
- Title** : Title of the Movie / Tv Show
- Director** : Director of the Movie
- Cast** : Actors involved in the movie/show
- Country** : Country where the movie/show was produced
- Date_added** : Date it was added on Netflix
- Release_year** : Actual Release year of the movie/show
- Rating** : TV Rating of the movie/show
- Duration** : Total Duration - in minutes or number of seasons
- Listed_in** : Genre
- Description** : The summary description

Summary

- Netflix added more movies as compare to TV shows, as the data shows almost 70% of the content are of type Movies
- Content for United States on Netflix is maximum followed by India and UK, as compare to other countries.
- Netflix content is mostly available for Mature Audience [Only for Adults]
- Very less no. of TV shows which are meant for Children
- Most popular genres in recent years are International movies, Dramas, Comedies, and International TV Shows
- In 2020, there is significant amount of drop in content added due to COVID pandemic

For **Movies**

- In United States, India and United kingdom movies are more popular as compare to other countries
- Almost same no. of movies are added on netflix every month.

- Most the movies are around the duration of 1h 30mins

For **TV Shows**

- Most of the TV shows have 1 season or 2 seasons.
- For Japan and South Korea, Netflix should focus more on TV shows as compare to movies

Recommendation

For **Movies**

- Preferred movies duration should be less than 2hrs or so.
- Netflix should add more movies for United States and India falling in category of International movies and comedies
- Netflix should add more movies for United States and India having rating of TV-MA & TV-14.
- Top three countries where movies added are United States, India & United Kingdom.
- Netflix should add Movies/TV Show on Friday than any other weekday.

For **TV Shows**

- Preferd movies duration is 1-2 seasons.
- Netflix should focus on countries like Japan, South Korea and France in TV shows , as they prefer TV shows over movies.
- Netflix shoud add TV Show on Friday than other weekday
- More content for children should be added

Detailed Analysis

Importing all the **Libs**

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Loading the **data**

```
In [2]: # data_set = 'https://d2beighkhq929f0.cloudfront.net/public_assets/assets/000/000/940/original/netflix.csv'
data_set = 'netflix.csv'

In [3]: df = pd.read_csv(data_set)

In [4]: df.shape

Out[4]: (8807, 12)

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.sample(5)
```

Out[6]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration		listed_in	description
	323	s324	TV Show	30 Rock	NaN	Tina Fey, Alec Baldwin, Tracy Morgan, Jane Kra...	United States	August 1, 2021	2012	TV-14	7 Seasons	TV Comedies	Liz Lemon juggles the demands of running an NB...
	4967	s4968	TV Show	Trump: An American Dream	NaN	Donald Trump	United Kingdom	March 30, 2018	2018	TV-14	1 Season	Docuseries	Friends, associates and critics reveal the tru...
	7979	s7980	Movie	Secrets of Westminster	Louise Wardle	Samuel West	United States, United Kingdom	April 22, 2017	2014	TV-PG	55 min	Documentaries, International Movies	Take an insider's tour of Parliament, Big Ben ...
	8486	s8487	Movie	The Ring	Gore Verbinski	Naomi Watts, Martin Henderson, David Dorfman, ...	United States	January 1, 2020	2002	PG-13	116 min	Horror Movies, Thrillers	A reporter watches a video that supposedly set...
	507	s508	Movie	Three Thieves	Udoka Oyeka	Koye Kekere Ekun, Shawn Faqua, Frank Donga, Lu...	NaN	July 7, 2021	2019	TV-14	108 min	Comedies, International Movies	Frustrated at work, three young men use a case...

Observation

In this data set we can see the follow which we need to take care before we do anything

- The following column contains the data seperated by comma (,) so we need to separate it out: **cast**, **director**, **country**, **listed_in**
- We have many **NaN** (missing value) in many fields, like **director**, **cast**, **country**
- The column **date_added** is a string, we may like to change it to date format
- The column **duration** is mixed, like in mins for Movie and no. of seasons for TV Show

Exploring the data

```
In [7]: df.describe()
```

```
Out[7]:
```

	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 [8]: 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 [9]: df.describe(include='object').T
```

```
Out[9]:
```

	count	unique	top	freq
show_id	8807	8807	s1	1
type	8807	2	Movie	6131
title	8807	8807	Dick Johnson Is Dead	1
director	6173	4528	Rajiv Chilaka	19
cast	7982	7692	David Attenborough	19
country	7976	748	United States	2818
date_added	8797	1767	January 1, 2020	109
rating	8803	17	TV-MA	3207
duration	8804	220	1 Season	1793
listed_in	8807	514	Dramas, International Movies	362
description	8807	8775	Paranormal activity at a lush, abandoned prope...	4

```
In [10]: df.isnull().sum()
```

```
Out[10]:
```

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	

```
In [11]: # Percentage of null value in each columns
(df.isnull().sum() / df.shape[0]) * 100
```

```
Out[11]:
```

show_id	0.000000
type	0.000000
title	0.000000
director	29.908028
cast	9.367549
country	9.435676
date_added	0.113546
release_year	0.000000
rating	0.045418
duration	0.034064
listed_in	0.000000
description	0.000000
dtype: float64	

```
In [12]: df.columns
```

```
Out[12]:
```

Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added', 'release_year', 'rating', 'duration', 'listed_in', 'description'], dtype='object')
--

```
In [13]: df.nunique()
```

```
Out[13]:
```

show_id	8807
type	2
title	8807
director	4528
cast	7692
country	748
date_added	1767
release_year	74
rating	17
duration	220
listed_in	514
description	8775
dtype: int64	

```
In [14]: df.shape
```

```
Out[14]: (8807, 12)
```

Unnesting the data for few of the cols (cast , director , country , listed_in)

In [15]: df.head(4)

Out[15]:

	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, Nabil...	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...

I. Unnesting Cast

In [16]:

```
df_1 = df.loc[:, 'cast'].apply(lambda x: str(x).split(' ')).tolist()
df_1 = pd.DataFrame(df_1, index=df.title).stack().reset_index()
df_1 = df_1.loc[:, ['title', 0]]
df_1 = df_1.rename(columns={0: "cast"})
```

In [17]: df_1

Out[17]:

	title	cast
0	Dick Johnson Is Dead	nan
1	Blood & Water	Ama Qamata
2	Blood & Water	Khosi Ngema
3	Blood & Water	Gail Mabalane
4	Blood & Water	Thabang Molaba
...
64946	Zubaan	Manish Chaudhary
64947	Zubaan	Meghna Malik
64948	Zubaan	Malkeet Rauni
64949	Zubaan	Anita Shabdish
64950	Zubaan	Chittaranjan Tripathy

64951 rows x 2 columns

II. Unnesting Directors

In [18]:

```
df_2 = df.loc[:, 'director'].apply(lambda x: str(x).split(' ')).tolist()
df_2 = pd.DataFrame(df_2, index=df.title).stack().reset_index()
df_2 = df_2.loc[:, ['title', 0]]
df_2 = df_2.rename(columns={0: "director"})
```

In [19]: df_2

Out[19]:

	title	director
0	Dick Johnson Is Dead	Kirsten Johnson
1	Blood & Water	nan
2	Ganglands	Julien Leclercq
3	Jailbirds New Orleans	nan
4	Kota Factory	nan
...
9607	Zodiac	David Fincher
9608	Zombie Dumb	nan
9609	Zombieland	Ruben Fleischer
9610	Zoom	Peter Hewitt
9611	Zubaan	Mozez Singh

9612 rows x 2 columns

III. Unnesting listed_in

In [20]:

```
df_3 = df.loc[:, 'listed_in'].apply(lambda x: str(x).split(' ')).tolist()
df_3 = pd.DataFrame(df_3, index=df.title).stack().reset_index()
df_3 = df_3.loc[:, ['title', 0]]
df_3 = df_3.rename(columns={0: "listed_in"})
```

In [21]: df_3

Out[21]:

	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
...
19318	Zoom	Children & Family Movies
19319	Zoom	Comedies
19320	Zubaan	Dramas
19321	Zubaan	International Movies
19322	Zubaan	Music & Musicals

19323 rows x 2 columns

IV. Unnesting country

In [22]:

```
df_4 = df.loc[:, 'country'].apply(lambda x: str(x).split(', ')).tolist()
df_4 = pd.DataFrame(df_4, index=df.title).stack().reset_index()
df_4 = df_4.loc[:, ['title', 0]]
df_4 = df_4.rename(columns={0: "country"})
```

In [23]:

```
df_4
```

Out[23]:

	title	country
0	Dick Johnson Is Dead	United States
1	Blood & Water	South Africa
2	Ganglands	nan
3	Jailbirds New Orleans	nan
4	Kota Factory	India
...
10840	Zodiac	United States
10841	Zombie Dumb	nan
10842	Zombieland	United States
10843	Zoom	United States
10844	Zubaan	India

10845 rows x 2 columns

Merging all the data after unnesting

In [24]:

```
# Merging the 4 dataframes, df_1 to df_4
df_5 = pd.merge(df_1, df_2, on='title', how='inner')
df_6 = pd.merge(df_5, df_3, on='title', how='inner')
df_7 = pd.merge(df_6, df_4, on='title', how='inner')

# Renaming the cols
df_7.rename(
    columns={
        "cast": "Actors",
        "director": "Director",
        "listed_in": "Genre",
        "country": "Country"
    }, inplace=True
)

# Replacing the NaN
df_7['Actors'].replace('nan', 'Unknown Actor', inplace=True)
df_7['Director'].replace('nan', 'Unknown Director', inplace=True)
df_7['Country'].replace('nan', np.NaN, inplace=True)
```

In [25]:

```
df_7
```

Out[25]:

	title	Actors	Director	Genre	Country
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa
...
201986	Zubaan	Anita Shabdish	Mozez Singh	International Movies	India
201987	Zubaan	Anita Shabdish	Mozez Singh	Music & Musicals	India
201988	Zubaan	Chittaranjan Tripathy	Mozez Singh	Dramas	India
201989	Zubaan	Chittaranjan Tripathy	Mozez Singh	International Movies	India
201990	Zubaan	Chittaranjan Tripathy	Mozez Singh	Music & Musicals	India

201991 rows x 5 columns

In [26]:

```
df_7.nunique()
```

Out[26]:

title	8807
Actors	36440
Director	4994
Genre	42
Country	127
dtype:	int64

Merging this new dataframe (df_7) with the original df so that we can get the other columns

In [27]:

```
# dropping those 4 columns from the original df
```

```
df_8 = df.drop(['director', 'cast', 'country', 'listed_in'], axis=1)
```

```
In [28]: df_8
```

	show_id	type	title	date_added	release_year	rating	duration	description
	0	s1	Movie	Dick Johnson Is Dead	September 25, 2021	2020	PG-13 90 min	As her father nears the end of his life, filmm...
	1	s2	TV Show	Blood & Water	September 24, 2021	2021	TV-MA 2 Seasons	After crossing paths at a party, a Cape Town t...
	2	s3	TV Show	Ganglands	September 24, 2021	2021	TV-MA 1 Season	To protect his family from a powerful drug lor...
	3	s4	TV Show	Jailbirds New Orleans	September 24, 2021	2021	TV-MA 1 Season	Feuds, flirtations and toilet talk go down amo...
	4	s5	TV Show	Kota Factory	September 24, 2021	2021	TV-MA 2 Seasons	In a city of coaching centers known to train l...

	8802	s8803	Movie	Zodiac	November 20, 2019	2007	R 158 min	A political cartoonist, a crime reporter and a...
	8803	s8804	TV Show	Zombie Dumb	July 1, 2019	2018	TV-Y7 2 Seasons	While living alone in a spooky town, a young g...
	8804	s8805	Movie	Zombieland	November 1, 2019	2009	R 88 min	Looking to survive in a world taken over by zo...
	8805	s8806	Movie	Zoom	January 11, 2020	2006	PG 88 min	Dragged from civilian life, a former superhero...
	8806	s8807	Movie	Zubaan	March 2, 2019	2015	TV-14 111 min	A scrappy but poor boy worms his way into a ty...

8807 rows x 8 columns

```
In [29]: df_7.head()
```

	title	Actors	Director	Genre	Country
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa

```
In [30]: # Merging df_7 and df_8 to get the final new dataframe (df_new)
df_new = pd.merge(df_7, df_8, on='title', how='left')
df_new.head(5)
```

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...

```
In [31]: df_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 201991 entries, 0 to 201990
Data columns (total 12 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   title            201991 non-null object
1   Actors           201991 non-null object
2   Director         201991 non-null object
3   Genre            201991 non-null object
4   Country          190094 non-null object
5   show_id          201991 non-null object
6   type             201991 non-null object
7   date_added       201833 non-null object
8   release_year     201991 non-null int64
9   rating           201924 non-null object
10  duration         201988 non-null object
11  description       201991 non-null object
dtypes: int64(1), object(11)
memory usage: 20.0+ MB
```

```
In [32]: # We see many country as Null as we replaced missing data for country with NaN
df_new.isnull().sum()
```

title	0
Actors	0
Director	0
Genre	0
Country	11897
show_id	0
type	0
date_added	158
release_year	0
rating	67
duration	3
description	0
dtype:	int64

```
In [33]: df_new.nunique()
```

title	8807
Actors	36440
Director	4994
Genre	42
Country	127
show_id	8807
type	2
date_added	1767
release_year	74
rating	17
duration	220
description	8775
dtype:	int64

```
In [34]: df_new.shape
```

(201991, 12)

Fixing the duration column for better analysis

```
In [35]: df_new['duration'].isnull().sum()
```

```
Out[35]: 3

In [36]: df_new[df_new['duration'].isnull()]

Out[36]:
```

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	
126537		Louis C.K. 2017	Louis C.K.	Louis C.K.	Movies	United States	s5542	Movie	April 4, 2017	2017	74 min	NaN	Louis C.K. muses on religion, eternal love, gi...
131603		Louis C.K.: Hilarious	Louis C.K.	Louis C.K.	Movies	United States	s5795	Movie	September 16, 2016	2010	84 min	NaN	Emmy-winning comedy writer Louis C.K. brings h...
131737		Louis C.K.: Live at the Comedy Store	Louis C.K.	Louis C.K.	Movies	United States	s5814	Movie	August 15, 2016	2015	66 min	NaN	The comic puts his trademark hilarious/thought...

```


In [37]: df_new.loc[df_new['duration'].isnull(), 'duration']

Out[37]:
126537    NaN
131603    NaN
131737    NaN
Name: duration, dtype: object

In [38]: df_new.loc[df_new['duration'].isnull(), 'rating']

Out[38]:
126537    74 min
131603    84 min
131737    66 min
Name: rating, dtype: object

In [39]: # We can see that these 3 missing duration is placed in the rating column. So lets fix that

# Fixing the duration column
df_new.loc[df_new['duration'].isnull(), 'duration'] = df_new.loc[df_new['duration'].isnull(), 'rating']

# Fixing the rating column
df_new.loc[df_new['rating'].str.contains('min', na=False), 'rating'] = np.nan

In [40]: df_new['duration'].isnull().sum()

Out[40]: 0

In [41]: df_new.loc[df_new['rating'].str.contains('min', na=False), 'rating']

Out[41]: Series([], Name: rating, dtype: object)

In [42]: df_new.sample(5)

Out[42]:
```

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description
158595	Holidays	Harley Quinn Smith	Nicholas McCarthy	Horror Movies	Ireland	s6985	Movie	July 15, 2016	2016	TV-MA	106 min	Eight gleefully grim short tales set on Hallow...
71891	Ares	Janni Goslinga	Unknown Director	TV Dramas	Netherlands	s3011	TV Show	January 17, 2020	2020	TV-MA	1 Season	Aiming to become part of Amsterdam's elite, an...
81497	Tatu	Oludara Egerton Shyngle	Don Omope	International Movies	Nigeria	s3408	Movie	October 18, 2019	2017	TV-MA	101 min	Doubting her future as a nun, a woman attempts...
86399	Let's Eat	Lee Soo-kyung	Unknown Director	Korean TV Shows	South Korea	s3611	TV Show	August 8, 2019	2013	TV-PG	1 Season	With the help of her neighbors, a divorced wom...
159012	Hormones	Gunn Junhavat	Unknown Director	International TV Shows	Thailand	s6997	TV Show	June 30, 2018	2015	TV-MA	3 Seasons	High school students in Bangkok wrestle with i...

```


In [43]: # Removing the mins from the duration column (this will be used for Movies)
df_new['duration_movies'] = df_new['duration'].str.replace('min', '')

# Removing the Seasons from the duration column (this will be used for TV Shows)
df_new['duration_tv_shows'] = df_new['duration'].str.replace('Seasons', '')
df_new['duration_tv_shows'] = df_new['duration_tv_shows'].str.replace('Season', '')

In [44]: df_new.sample(4)

Out[44]:
```

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows
1184	The Smart Money Woman	Karibi Fubara	Bunmi Ajakaiye	International TV Shows	NaN	s48	TV Show	September 16, 2021	2020	TV-MA	1 Season	Five glamorous millennials strive for success ...	1 Season	1
20235	Never Back Down 2: The Beatdown	Scottie Epstein	Michael Jai White	Action & Adventure	United States	s802	Movie	June 2, 2021	2011	R	104 min	A group of mixed martial arts fighters stars i...	104	104 min
43359	Heritages	Diane Aractingi	Philippe Aractingi	International Movies	Switzerland	s1821	Movie	October 19, 2020	2014	TV-14	96 min	After being forced to leave his country three ...	96	96 min
105449	22 July	Anders Danielsen Lie	Paul Greengrass	Dramas	Norway	s4523	Movie	October 10, 2018	2018	R	144 min	After devastating terror attacks in Norway, a ...	144	144 min

```


In [45]: def filter_tv_shows(d):
            if (type(d) is not float) and ('min' in d):
                return 0
            else:
                return d

        def filter_movie(d):
            if (type(d) is not float) and ('Season' in d):
                return 0
            else:
                return d

        df_new['duration_tv_shows'] = df_new['duration_tv_shows'].apply(filter_tv_shows)
        df_new['duration_movies'] = df_new['duration_movies'].apply(filter_movie)

In [46]: df_new.sample(5)
```

Out[46]:		title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows
	175214	Painted Faces	Ching-Ying Lam	Alex Law	Dramas	Hong Kong	s7698	Movie	August 16, 2018	1988	TV-PG	108 min	In this dramatized rendition, students undergo...	108	0
	139444	Aram, Aram	John Roohinian	Christopher Chambers	Dramas	United States	s6183	Movie	February 15, 2017	2015	TV-14	79 min	Newly orphaned Aram leaves Beirut to live with...	79	0
	105968	Monty Python's Flying Circus	Connie Booth	Unknown Director	TV Comedies	West Germany	s4551	TV Show	October 2, 2018	1972	TV-MA	1 Season	The Pythons elevate the absurd to new heights ...	0	1
	132498	Belgica	Stefaan De Winter	Felix van Groeningen	International Movies	Belgium	s5852	Movie	April 15, 2016	2016	TV-MA	127 min	Two brothers become partners in a modest night...	127	0
	147863	Cop and a Half: New Recruit	Vincent Tong	Jon Rosenbaum	Children & Family Movies	Canada	s6535	Movie	August 17, 2017	2017	PG	96 min	A world-weary detective is forced to team up w...	96	0

```
In [47]: df_new['duration_tv_shows'] = df_new['duration_tv_shows'].astype('int')
df_new['duration_movies'] = df_new['duration_movies'].astype('int')
```

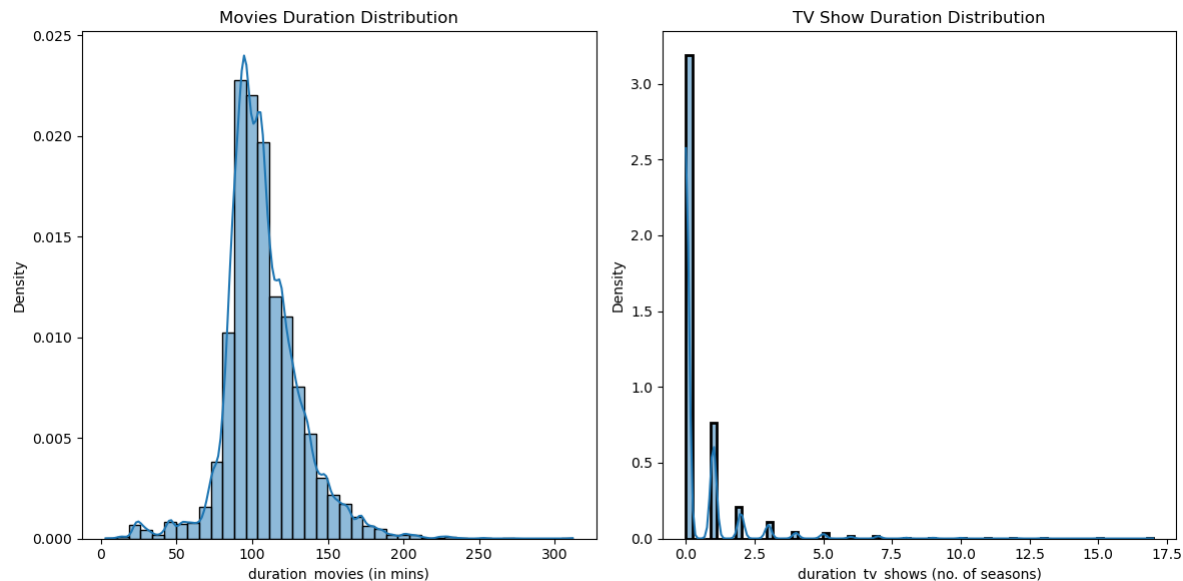
```
In [48]: # Create a figure and subplots
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 6))

# Plotting the Movies duration distribution
movie_duration = df_new.loc[df_new['duration_movies'] > 0, 'duration_movies']
sns.histplot(movie_duration, kde=True, bins=40,
             edgecolor='black', linewidth=1, stat='density', ax=ax1)
ax1.set_title('Movies Duration Distribution')
ax1.set_xlabel('duration_movies (in mins)')

# Plotting the TV Shows duration distribution
tv_duration = df_new['duration_tv_shows']
sns.histplot(tv_duration, kde=True, bins=75,
             edgecolor='black', linewidth=2, stat='density', ax=ax2)
ax2.set_title('TV Show Duration Distribution')
ax2.set_xlabel('duration_tv_shows (no. of seasons)')

# Adjust spacing between subplots
fig.tight_layout()

# Display the plot
plt.show()
```



Observation

- As we can see most of the TV shows have 1 season.
- Most the movies are around the duration of 1h 30mins

Changing the datatypes for the `date_added` column

```
In [49]: df_new.dtypes
```

```
Out[49]: title          object
Actors          object
Director        object
Genre           object
Country         object
show_id         object
type            object
date_added      object
release_year    int64
rating          object
duration        object
description     object
duration_movies int64
duration_tv_shows int64
dtype: object
```

```
In [50]: df_new['date_added'].isnull().sum()
```

```
Out[50]: 158
```

```
In [51]: df['date_added']
```



```
Out[51]: 0      September 25, 2021
1      September 24, 2021
2      September 24, 2021
3      September 24, 2021
4      September 24, 2021
...
8802   November 20, 2019
8803   July 1, 2019
8804   November 1, 2019
8805   January 11, 2020
8806   March 2, 2019
Name: date_added, Length: 8807, dtype: object
```

```
In [52]: def convert_datetime(d):
        if type(d) is not float:
            d = str(d).strip()
        return d
```

```
In [53]: df_new['date_added'] = df_new['date_added'].map(convert_datetime)
```

```
In [54]: df_new['date_added'].isna().sum()
```

Out[54]: 158

```
In [55]: df_new['date_added'] = pd.to_datetime(df_new['date_added'])
```

```
In [56]: df_new['date_added'].isna().sum()
```

Out[56]: 158

```
In [57]: df_new['date_added_month'] = df_new['date_added'].dt.month
df_new['date_added_year'] = df_new['date_added'].dt.year
df_new['date_added_day'] = df_new['date_added'].dt.day
df_new['date_added_day_name'] = df_new['date_added'].dt.day_name()
df_new['date_added_month_name'] = df_new['date_added'].dt.month_name()
```

```
In [58]: df_new.sample(10)
```

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows	date_added_month	date_added_year
87083	The Twins Effect II	Tony Leung Ka Fai	Corey Yuen	International Movies	Hong Kong	s3650	Movie	2019-07-21	2004	TV-14	106 min	An empress rules over a land where men are sub...	106	0	7.0	2019
138619	American Psycho	Jared Leto	Mary Harron	Cult Movies	United States	s6148	Movie	2019-09-01	2000	R	102 min	With chiseled good looks that belie his insani...	102	0	9.0	2019
195717	The Time Traveler's Wife	Arliss Howard	Robert Schwentke	Sci-Fi & Fantasy	United States	s8533	Movie	2019-10-01	2009	PG-13	107 min	Due to a genetic disorder, handsome librarian ...	107	0	10.0	2019
6975	Hit & Run	Sanaa Lathan	Unknown Director	TV Action & Adventure	Israel	s297	TV Show	2021-08-06	2021	TV-MA	1 Season	A man searching for the truth behind his wife...	0	1	8.0	2021
111805	Brain on Fire	Chloë Grace Moretz	Gerard Barrett	Dramas	Canada	s4820	Movie	2018-06-22	2016	PG-13	89 min	Stricken with seizures, psychosis and memory l...	89	0	6.0	2018
109928	Fashion	Mugdha Godse	Madhur Bhandarkar	International Movies	India	s4722	Movie	2018-08-02	2008	TV-MA	158 min	A small-town girl finally realizes her dream o...	158	0	8.0	2018
99627	Watership Down	Craig Parkinson	Noam Murro	International TV Shows	United States	s4263	TV Show	2018-12-23	2018	TV-PG	1 Season	A warren of rabbits battles many threats on th...	0	1	12.0	2018
186050	Taken	Michael Irby	Unknown Director	TV Action & Adventure	France	s8148	TV Show	2019-01-12	2018	TV-14	2 Seasons	After a personal tragedy upends his life, a hi...	0	2	1.0	2018
56459	Underdogs	Eugenio Derbez	Juan José Campanella	Children & Family Movies	Spain	s2386	Movie	2020-06-15	2013	PG	85 min	When his archenemy returns to settle a score, ...	85	0	6.0	2020
18297	Penguin Town	Unknown Actor	Unknown Director	Science & Nature TV	NaN	s711	TV Show	2021-06-16	2021	TV-PG	1 Season	In a picturesque South African town, an eclectic...	0	1	6.0	2021

Analysis of the Genre column

```
In [59]: df_new['Genre'].value_counts()
```

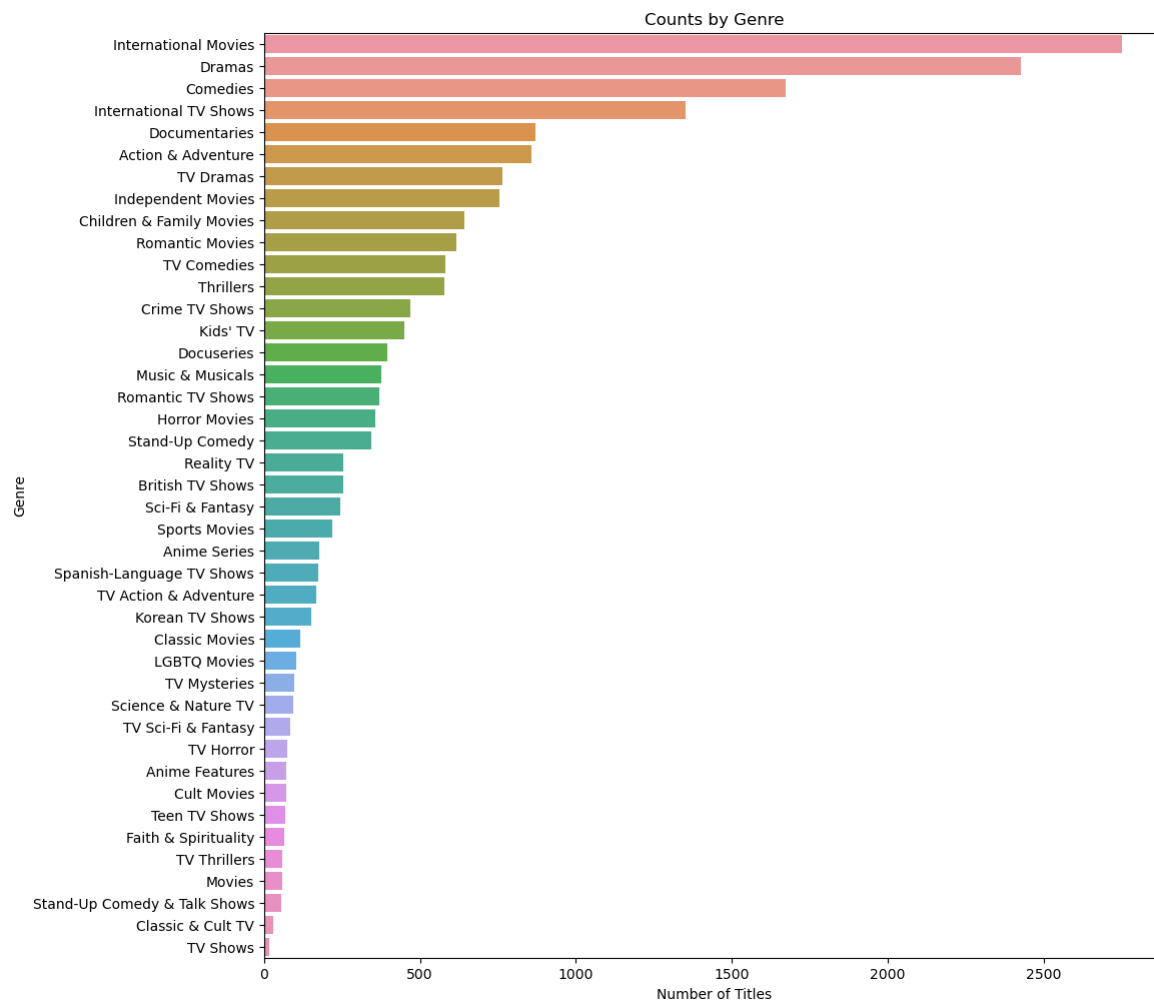
```
Out[59]:
Dramas                29775
International Movies   28211
Comedies              20829
International TV Shows 12845
Action & Adventure    12216
Independent Movies     9834
Children & Family Movies 9771
TV Dramas            8942
Thrillers            7107
Romantic Movies      6412
TV Comedies          4963
Crime TV Shows       4733
Horror Movies        4571
Kids' TV             4568
Sci-Fi & Fantasy     4037
Music & Musicals     3077
Romantic TV Shows    3049
Documentaries        2407
Anime Series         2313
TV Action & Adventure 2288
Spanish-Language TV Shows 2126
British TV Shows     1808
Sports Movies        1531
Classic Movies       1434
TV Mysteries        1281
Korean TV Shows     1122
Cult Movies          1077
TV Sci-Fi & Fantasy  1045
Anime Features       1045
TV Horror            941
Docuseries           845
LGBTQ Movies         838
TV Thrillers         768
Teen TV Shows        742
Reality TV           735
Faith & Spirituality 719
Stand-Up Comedy     540
Movies              412
TV Shows            337
Classic & Cult TV    272
Stand-Up Comedy & Talk Shows 268
Science & Nature TV  157
Name: Genre, dtype: int64
```

```
In [60]: genre_title_count = df_new.groupby('Genre').agg({"title": "nunique"}).sort_values(by=['title'],
                                                    ascending=False).reset_index()
genre_title_count.columns = ['Genre', 'Title']
```

```
In [61]: # Plotting a horizontal bar chart
plt.figure(figsize=(30, 12))
sns.barplot(x='Title', y='Genre', data=genre_title_count)

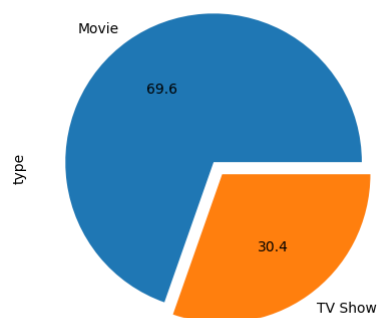
# Set plot title and axis labels
plt.title('Counts by Genre')
plt.xlabel('Number of Titles')
plt.ylabel('Genre')

plt.subplots_adjust(left=0.6)
```



Analysis based on type column (TV show vs Movies)

```
In [62]: # Check the distribution of TV Show and Movie
df['type'].value_counts(normalize=True).plot(kind='pie', autopct='%1f', explode=(0.05,0.05));
```



Observation

As we can see almost 70% of the content are Movie and 30% are TV shows

```
In [63]: df_new
```

Out[63]:

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows	date_added_month	date_added_ye
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	90	0	9.0	202
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	202
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	202
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	202
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	202
...
201986	Zubaan	Anita Shabdish	Mozes Singh	International Movies	India	s8807	Movie	2019-03-02	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	111	0	3.0	2015
201987	Zubaan	Anita Shabdish	Mozes Singh	Music & Musicals	India	s8807	Movie	2019-03-02	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	111	0	3.0	2015
201988	Zubaan	Chittaranjan Tripathy	Mozes Singh	Dramas	India	s8807	Movie	2019-03-02	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	111	0	3.0	2015
201989	Zubaan	Chittaranjan Tripathy	Mozes Singh	International Movies	India	s8807	Movie	2019-03-02	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	111	0	3.0	2015
201990	Zubaan	Chittaranjan Tripathy	Mozes Singh	Music & Musicals	India	s8807	Movie	2019-03-02	2015	TV-14	111 min	A scrappy but poor boy worms his way into a ty...	111	0	3.0	2015

201991 rows × 19 columns

Analysis based on country column

In [64]:

```
#number of distinct titles on the basis of country
country_title_count = df_new.groupby('Country').agg({"title":"nunique"}).sort_values(by=['title'],
                                             ascending=False).reset_index()
country_title_count.columns = ['Country', 'Title']
```

In [65]:

```
country_title_count[country_title_count.Title > 100]
```

Out[65]:

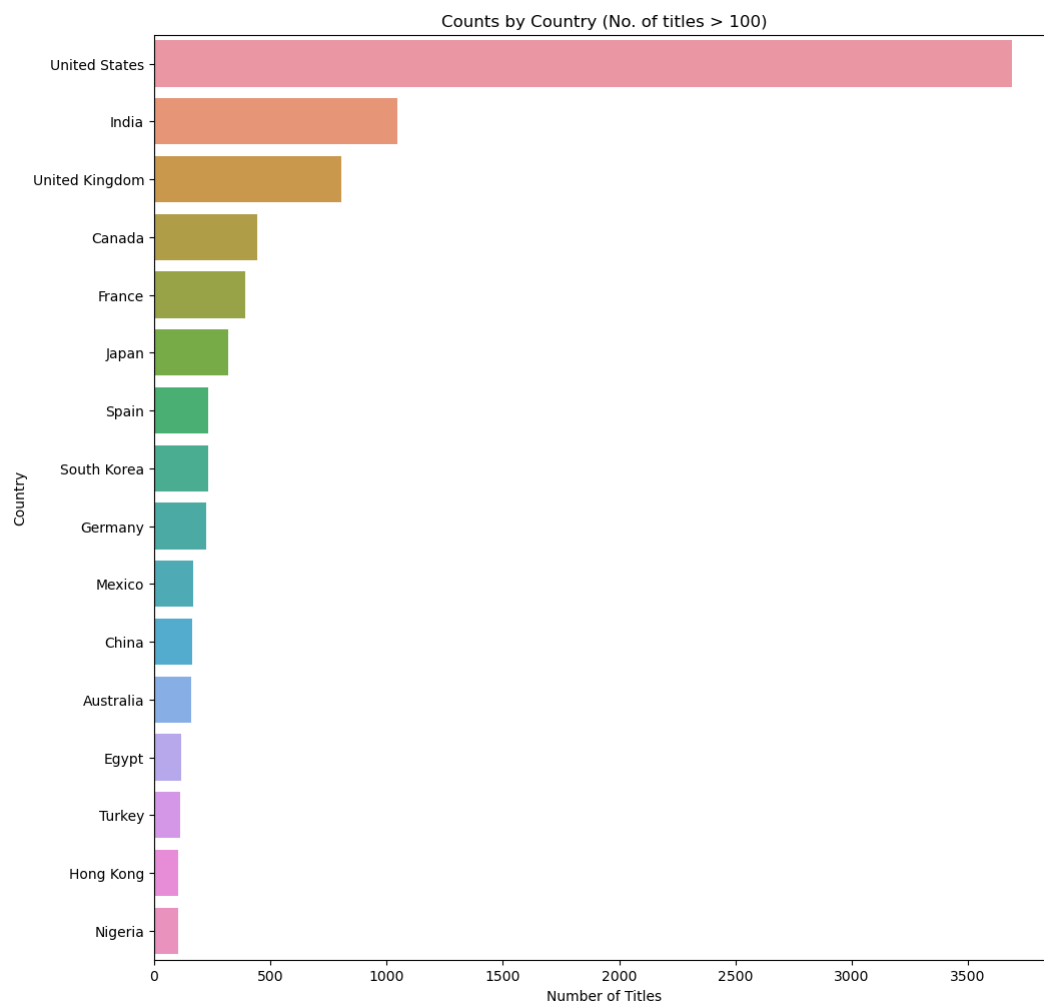
	Country	Title
0	United States	3689
1	India	1046
2	United Kingdom	804
3	Canada	445
4	France	393
5	Japan	318
6	Spain	232
7	South Korea	231
8	Germany	226
9	Mexico	169
10	China	162
11	Australia	160
12	Egypt	117
13	Turkey	113
14	Hong Kong	105
15	Nigeria	103

In [66]:

```
# Plotting a horizontal bar chart
plt.figure(figsize=(30, 12))
sns.barplot(x='Title', y='Country', data=country_title_count[country_title_count.Title > 100])

# Set plot title and axis labels
plt.title('Counts by Country (No. of titles > 100)')
plt.xlabel('Number of Titles')
plt.ylabel('Country')

plt.subplots_adjust(left=0.6)
```



Observation

US, India, UK, Canada and France are leading countries in Content Creation on Netflix

```
In [67]: #number of distinct titles on the basis of country
country_title_count = df_new.groupby(['Country', 'type']).agg({"title": "nunique"}).sort_values(by=['title'],
                                                    ascending=False).reset_index()
country_title_count.columns = ['Country', 'Type', 'Count']

In [68]: top_10_countries = country_title_count.groupby(['Country']).agg({"Count": "sum"}).sort_values(by=['Count'],
                                                    ascending=False).reset_index()[:10]['Country'].tolist()

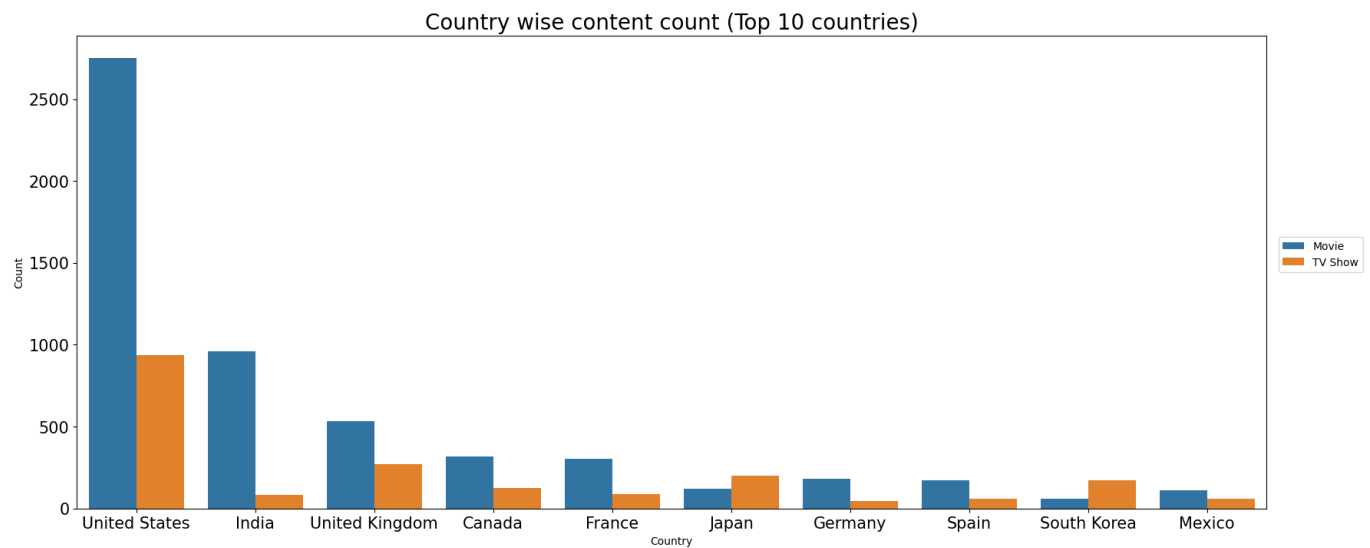
top_10 = country_title_count[country_title_count['Country'].isin(top_10_countries)]

In [69]: plt.figure(figsize=(20,8))
sns.barplot(x="Country", y="Count", data=top_10, hue="Type")
plt.xlabel('Country')
plt.ylabel('Count')

colors = ["#FF7F0E", "#1F77B5"] # Custom colors for movies and TV shows bars
sns.set_palette(sns.color_palette(colors))

plt.title("Country wise content count (Top 10 countries)", fontsize=20)
plt.xticks(fontsize=15)
plt.yticks(fontsize=15)

plt.legend(loc=(1.01,0.5))
plt.show()
```



Observation

For Japan and South Korea, Netflix should focus more on TV shows as compare to movies For India and US, most of the content are in the form of Movies

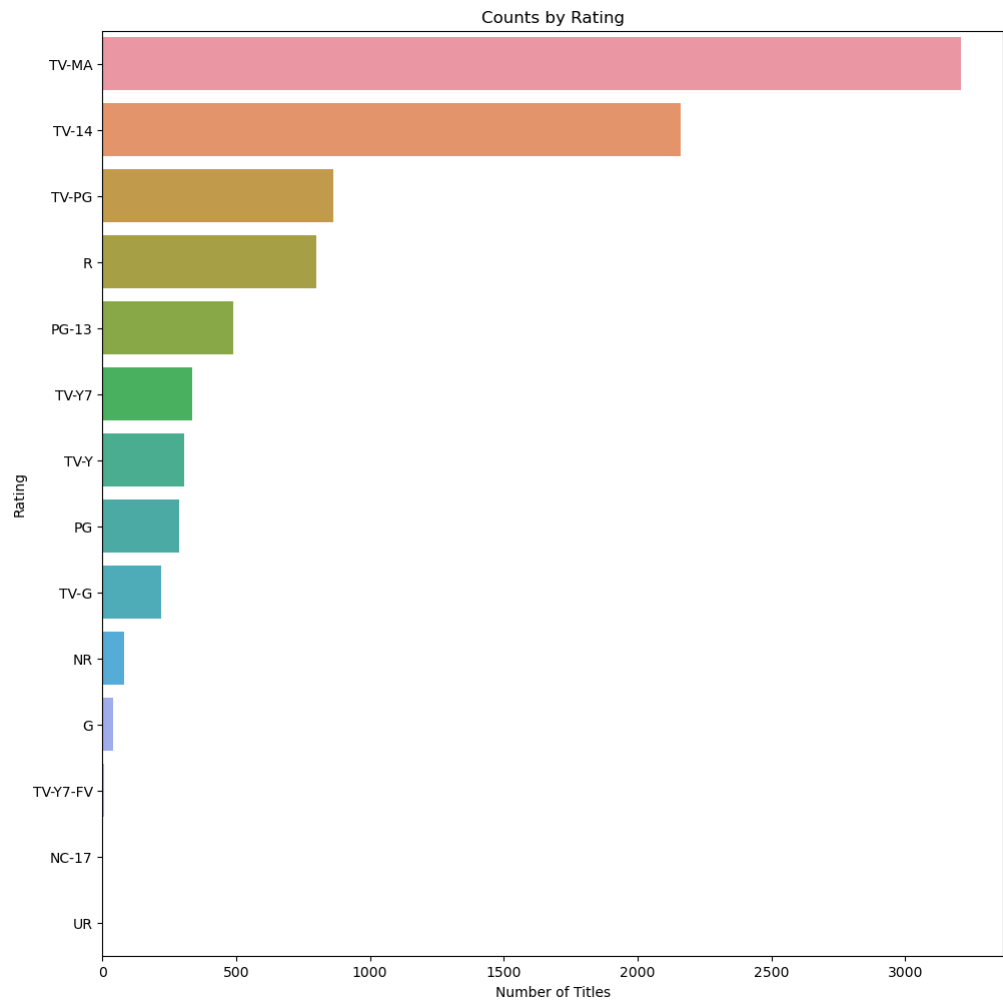
Analysis based on rating

```
In [70]: #number of distinct titles on the basis of rating
rating_title_count = df_new.groupby('rating').agg({"title": "nunique"}).sort_values(by=['title'],
                                             ascending=False).reset_index()
rating_title_count.columns = ['rating', 'Title']
```

```
In [71]: # Plotting a horizontal bar chart
plt.figure(figsize=(30, 12))
sns.barplot(x='Title', y='rating', data=rating_title_count)

# Set plot title and axis labels
plt.title('Counts by Rating')
plt.xlabel('Number of Titles')
plt.ylabel('Rating')

plt.subplots_adjust(left=0.6)
```



Observation

As per the above chart, most of the highly rated content on Netflix is intended for Mature Audiences, R Rated, content not intended for audience under 14 and those which require Parental Guidance

Analysis based on duration

I. For Movies

```
In [72]: #number of distinct titles on the basis of duration
df_new.groupby(['duration_movies']).agg({"title":"nunique"}).reset_index()

duration_title_count = df_new.groupby(['duration_movies']).agg({"title":"nunique"}).sort_values(by=['title'],
                                                    ascending=False).reset_index()

duration_title_count.columns = ['Duration (in mins)', 'Title']
duration_title_count.drop(0, inplace=True)

In [73]: duration_title_count

Out[73]:
```

	Duration (in mins)	Title
1	90	152
2	97	146
3	93	146
4	94	146
5	91	144
...
201	18	1
202	43	1
203	178	1
204	5	1
205	312	1

205 rows x 2 columns

```
In [74]: # Set bin ranges
bins = [1] + list(range(60, duration_title_count['Duration (in mins)'].max() + 30, 30))

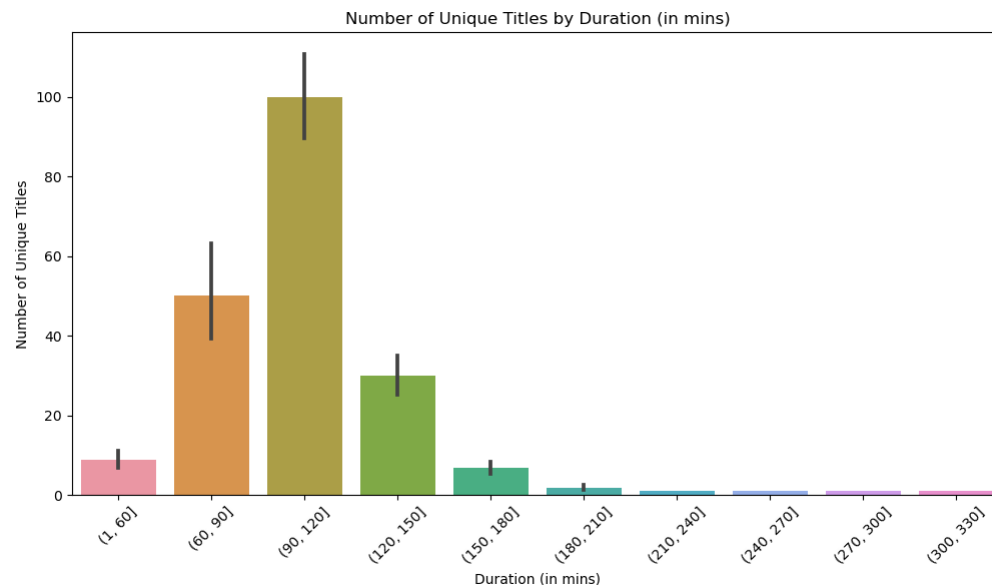
# Plotting the bar chart with bin size of 30 starting from 1
plt.figure(figsize=(12, 6))
sns.barplot(x=pd.cut(duration_title_count['Duration (in mins)'], bins=bins), y='Title', data=duration_title_count)

# Set plot title and axis labels
plt.title('Number of Unique Titles by Duration (in mins)')
```

```
plt.xlabel('Duration (in mins)')
plt.ylabel('Number of Unique Titles')

# Rotate x-axis labels for better readability (if needed)
plt.xticks(rotation=45)

# Display the plot
plt.show()
```



Observation

As per the above chart, the most watched movie has a duration from 90mins to 120mins

II. For TV Shows

```
In [75]: #number of distinct titles on the basis of duration
df_new.groupby(['duration_tv_shows']).agg({"title": "nunique").reset_index()

seasons_title_count = df_new.groupby(['duration_tv_shows']).agg({"title": "nunique").sort_values(by=['title'],
                                                                                               ascending=False).reset_index()

seasons_title_count.columns = ['No. of Seasons', 'Title']
seasons_title_count.drop(0, inplace=True)
```

```
In [76]: seasons_title_count
```

```
Out[76]:
```

	No. of Seasons	Title
1	1	1793
2	2	425
3	3	199
4	4	95
5	5	65
6	6	33
7	7	23
8	8	17
9	9	9
10	10	7
11	13	3
12	11	2
13	12	2
14	15	2
15	17	1

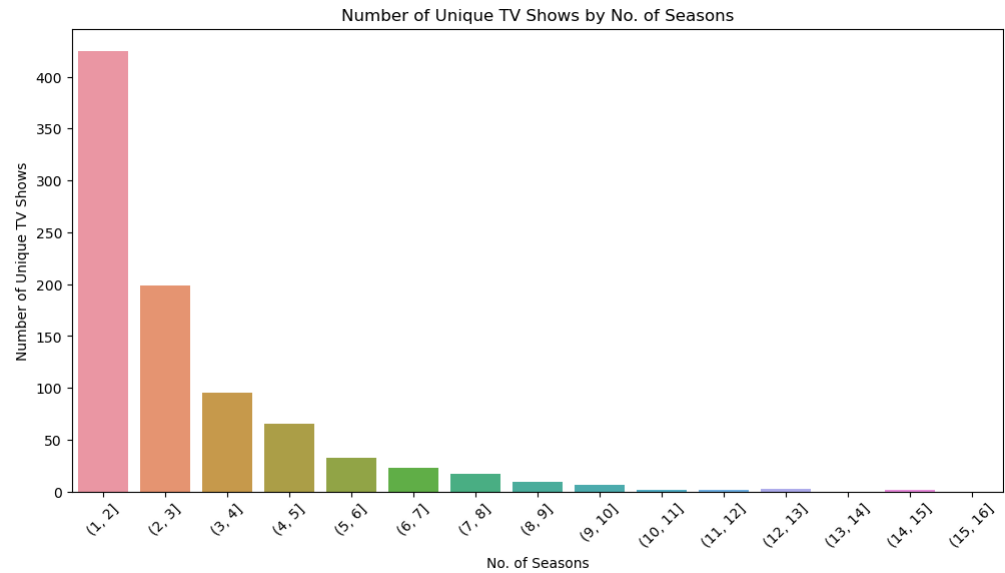
```
In [77]: # Set bin ranges
bins = list(range(1, seasons_title_count['No. of Seasons'].max()))

# Plotting the bar chart with bin size of 30 starting from 1
plt.figure(figsize=(12, 6))
sns.barplot(x=pd.cut(seasons_title_count['No. of Seasons'], bins=bins), y='Title', data=seasons_title_count)

# Set plot title and axis labels
plt.title('Number of Unique TV Shows by No. of Seasons')
plt.xlabel('No. of Seasons')
plt.ylabel('Number of Unique TV Shows')

# Rotate x-axis labels for better readability (if needed)
plt.xticks(rotation=45)

# Display the plot
plt.show()
```

```
In [78]: bins
Out[78]: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16]
```

Observation

As per the above chart, the most watched TV Show has a 1 or 2 seasons

Analysis based on Actor

```
In [79]: #number of distinct titles on the basis of duration
df_new.groupby(['Actors']).agg({"title":"nunique"}).reset_index()

actor_title_count = df_new.groupby(['Actors']).agg({"title":"nunique"}).sort_values(by=['title'],
                                         ascending=False).reset_index()
actor_title_count.columns = ['Actors', 'Title']
```

```
In [80]: actor_title_count
```

	Actors	Title
0	Unknown Actor	825
1	Anupam Kher	43
2	Shah Rukh Khan	35
3	Julie Tejwani	33
4	Naseeruddin Shah	32
...
36435	Jamie Lee	1
36436	Jamie Kenna	1
36437	Jamie Kaler	1
36438	Jamie Johnston	1
36439	Şöpe Dirisü	1

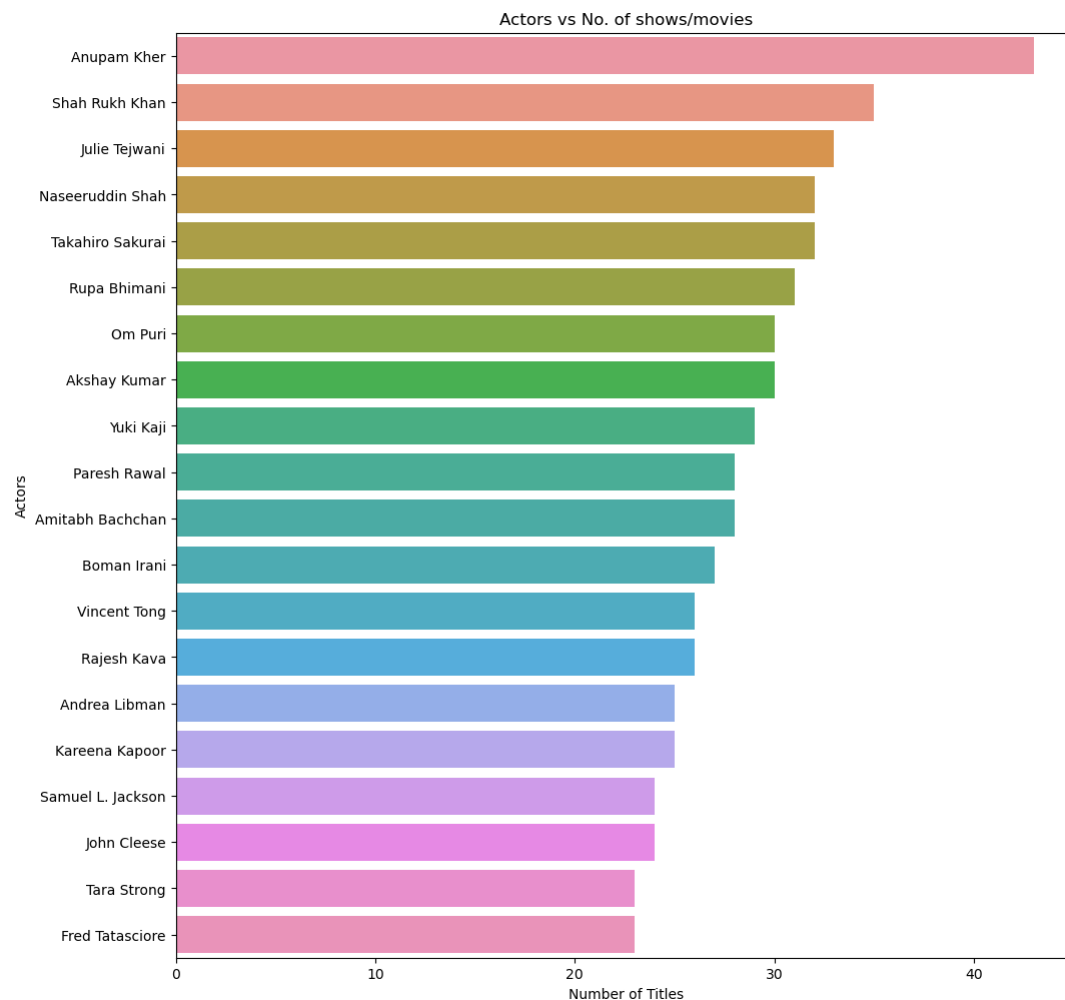
36440 rows x 2 columns

```
In [81]: top_20_actors = actor_title_count[actor_title_count['Actors'] != 'Unknown Actor'].head(20)
```

```
In [82]: # Plotting a horizontal bar chart
plt.figure(figsize=(30, 12))
sns.barplot(x='Title', y='Actors', data=top_20_actors)

# Set plot title and axis labels
plt.title('Actors vs No. of shows/movies')
plt.xlabel('Number of Titles')
plt.ylabel('Actors')

plt.subplots_adjust(left=0.6)
```



Observation

As per the above chart, Anupam Kher, Shah Rukh Khan, Julie Tejewani, Naseeruddin Shah and Takahiro Sakurai occupy the top spot in Most Watched content.

Analysis based on Directors

```
In [83]: #number of distinct titles on the basis of duration
df_new.groupby(['Director']).agg({"title": "nunique").reset_index()

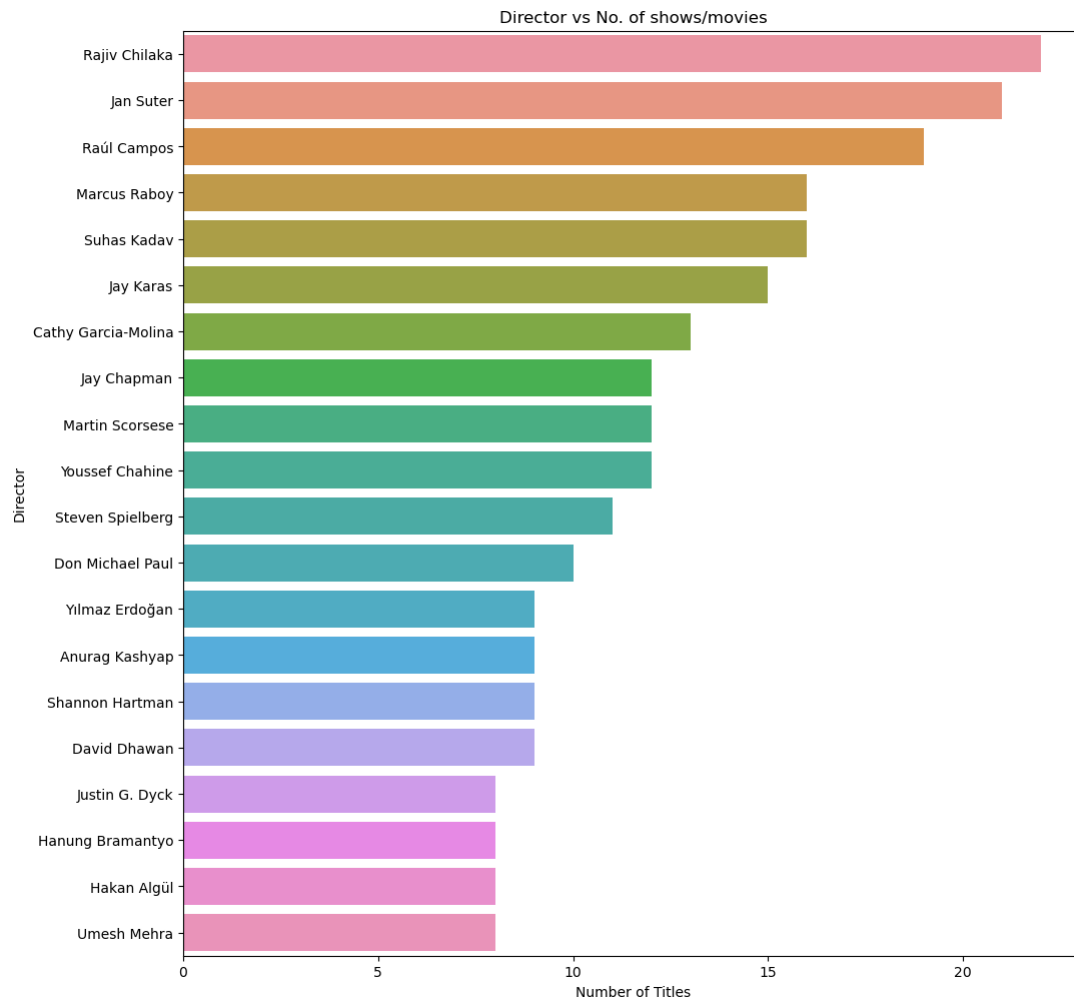
directors_title_count = df_new.groupby(['Director']).agg({"title": "nunique").sort_values(by='title',
                                              ascending=False).reset_index()
directors_title_count.columns = ['Director', 'Title']

In [84]: top_20_director = directors_title_count[directors_title_count['Director'] != 'Unknown Director'].head(20)

In [85]: # Plotting a horizontal bar chart
plt.figure(figsize=(30, 12))
sns.barplot(x='Title', y='Director', data=top_20_director)

# Set plot title and axis labels
plt.title('Director vs No. of shows/movies')
plt.xlabel('Number of Titles')
plt.ylabel('Director')

plt.subplots_adjust(left=0.6)
```



Observation

As per the above chart, Rajiv Chilaka, Jan Suter and Raul Campos are the most popular directors across Netflix.

Analysis based on number of TV Show/Movie released over years

```
In [86]: df_new.head(2)
```

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows	date_added_month	date_added_year	date
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	90	0	9.0	2021.0	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	

```
In [87]: release_count_by_year = df_new.groupby(['date_added_year']).agg({"title": "nunique"}).sort_values(by=['title'], ascending=False).reset_index()
```

```
release_count_by_year.columns = ['Year', 'Count']
release_count_by_year.Year = release_count_by_year.Year.astype('int')
```

```
In [88]: release_count_by_year
```

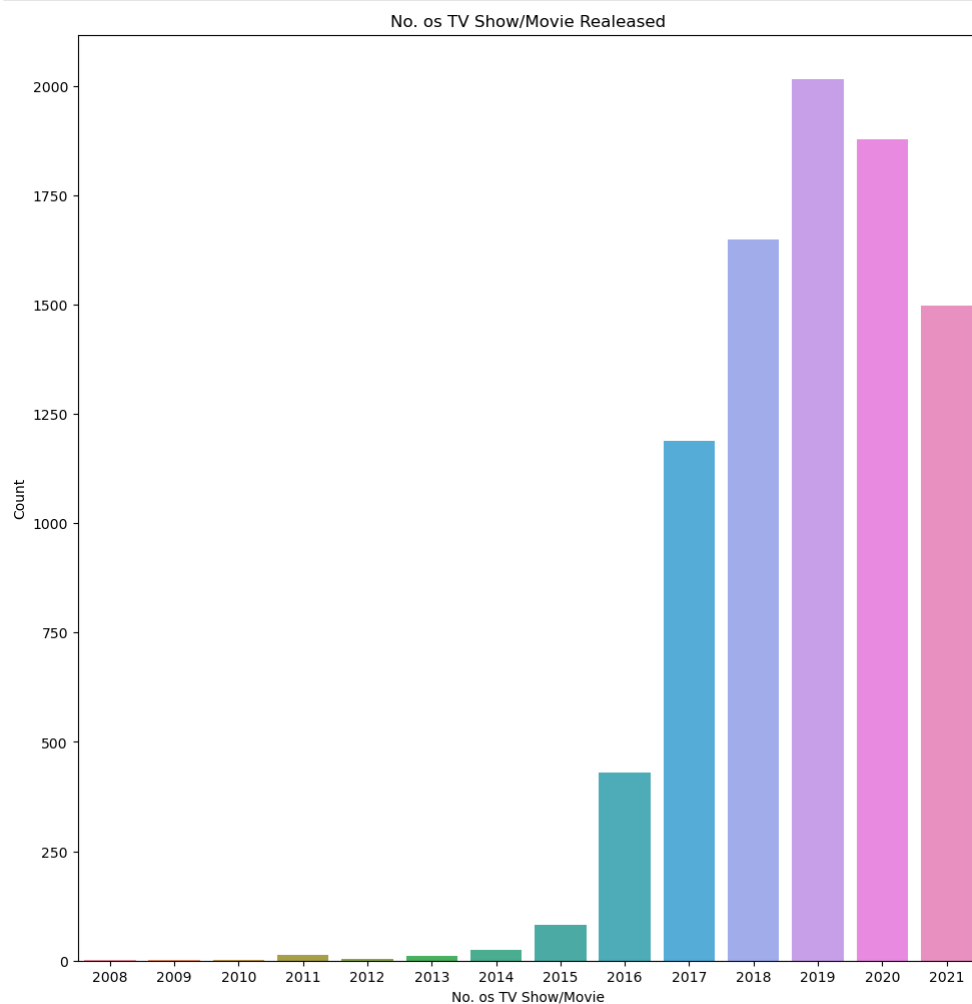
```
Out[88]:
```

	Year	Count
0	2019	2016
1	2020	1879
2	2018	1649
3	2021	1498
4	2017	1188
5	2016	429
6	2015	82
7	2014	24
8	2011	13
9	2013	11
10	2012	3
11	2008	2
12	2009	2
13	2010	1

```
In [89]: # Plotting a horizontal bar chart
plt.figure(figsize=(30, 12))
sns.barplot(x='Year', y='Count', data=release_count_by_year)

# Set plot title and axis labels
plt.title('No. os TV Show/Movie Realeased')
plt.xlabel('No. os TV Show/Movie')
plt.ylabel('Count')

plt.subplots_adjust(left=0.6)
```



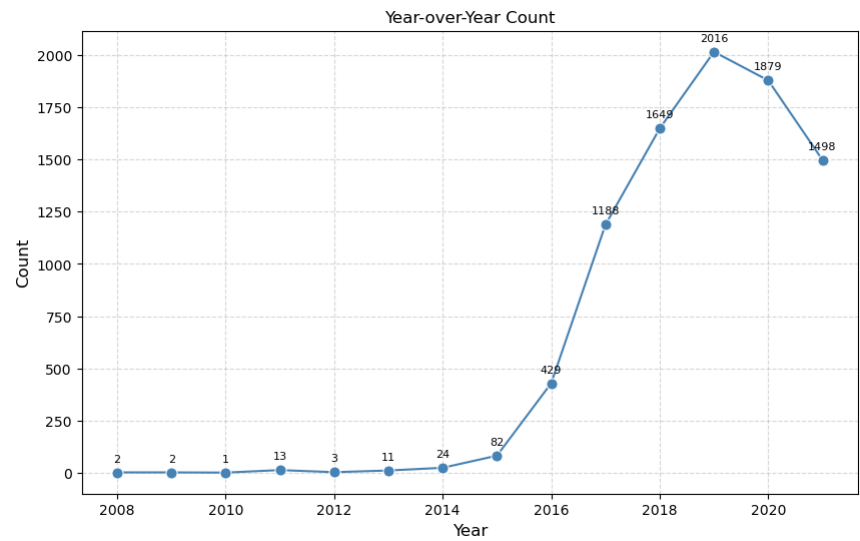
```
In [90]: # Plotting the year-over-year count
plt.figure(figsize=(10, 6))
sns.lineplot(x='Year', y='Count', data=release_count_by_year, marker='o', markersize=8, color='steelblue')

# Set plot title and axis labels
plt.title('Year-over-Year Count')
plt.xlabel('Year', fontsize=12)
plt.ylabel('Count', fontsize=12)

# Customize tick labels and gridlines
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.grid(True, linestyle='--', alpha=0.5)

# Add data labels to each point
for i, count in enumerate(release_count_by_year['Count']):
    plt.text(release_count_by_year['Year'][i], count + 50, str(count), ha='center', fontsize=8, color='black')

# Display the plot
plt.show()
```



Observation

The Amount of Content across Netflix has increased from 2008 continuously till 2019. Then started decreasing from here(that might be due to COVID)

Analysis of TV Shows and Movies over time

```
In [91]: df_new.head()
```

```
Out[91]:
```

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows	date_added_month	date_added_year	date
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	90	0	9.0	2021.0	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	

```
In [92]: df_datetime_month = df_new.groupby(['type', 'date_added_month', 'date_added_month_name']).agg({"title": "nunique"}).reset_index()
```

```
df_datetime_month
```

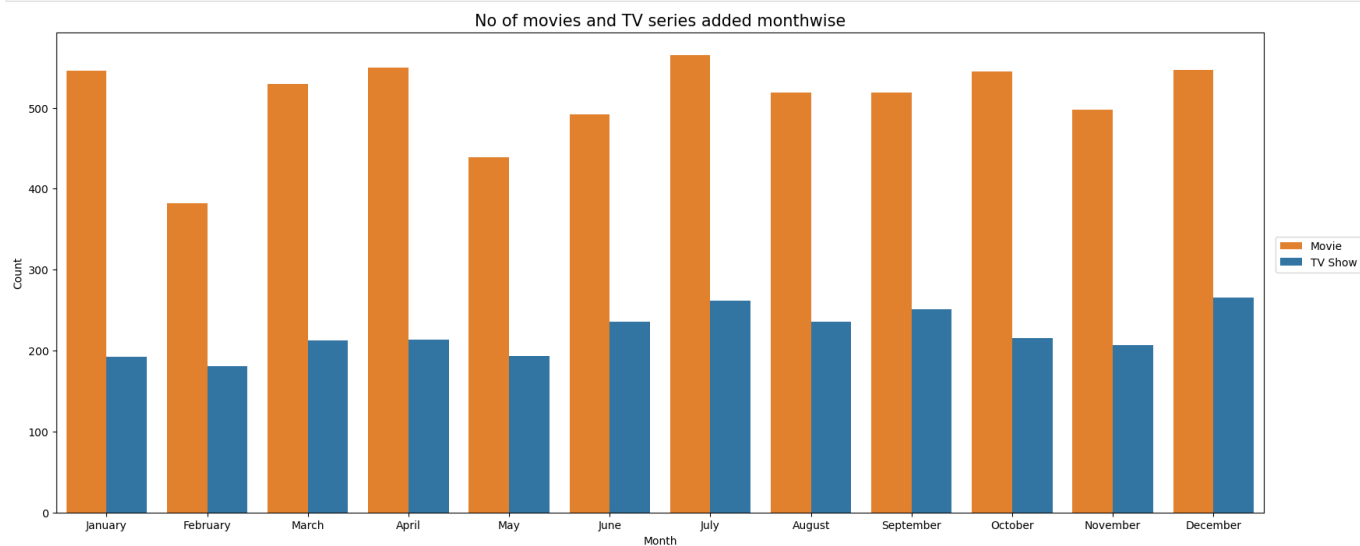
```
Out[92]:
```

	type	date_added_month	date_added_month_name	title
0	Movie	1.0	January	546
1	Movie	2.0	February	382
2	Movie	3.0	March	529
3	Movie	4.0	April	550
4	Movie	5.0	May	439
5	Movie	6.0	June	492
6	Movie	7.0	July	565
7	Movie	8.0	August	519
8	Movie	9.0	September	519
9	Movie	10.0	October	545
10	Movie	11.0	November	498
11	Movie	12.0	December	547
12	TV Show	1.0	January	192
13	TV Show	2.0	February	181
14	TV Show	3.0	March	213
15	TV Show	4.0	April	214
16	TV Show	5.0	May	193
17	TV Show	6.0	June	236
18	TV Show	7.0	July	262
19	TV Show	8.0	August	236
20	TV Show	9.0	September	251
21	TV Show	10.0	October	215
22	TV Show	11.0	November	207
23	TV Show	12.0	December	266

```
In [93]: plt.figure(figsize=(20,8))
sns.barplot(x= "date_added_month_name", y= "title", data = df_datetime_month, hue="type")
plt.xlabel('Month')
plt.ylabel('Count')

colors = ["#FF7F0F", "#1F77B5"] # Custom colors for movies and TV shows bars
sns.set_palette(sns.color_palette(colors))

plt.title("No of movies and TV series added monthwise", fontsize=15)
plt.legend(loc=(1.01,0.5))
plt.show()
```



Observation

July and December are the months when most content both for Movies and TV Shows, thats mostly beacuse of Vacation Time (schools/colleges), or may be beacuse of Summer/Winter break

No of movies added per month is greater then no of TV shows added per month.

```
In [94]: df_datetime_year = df_new.groupby(['type', 'date_added_year']).agg({"title": "nunique").reset_index()
df_datetime_year['date_added_year'] = df_datetime_year['date_added_year'].astype('int')
df_datetime_year_last_7years = df_datetime_year[df_datetime_year['date_added_year'] > 2013]
df_datetime_year_last_7years
```

```
Out[94]:
```

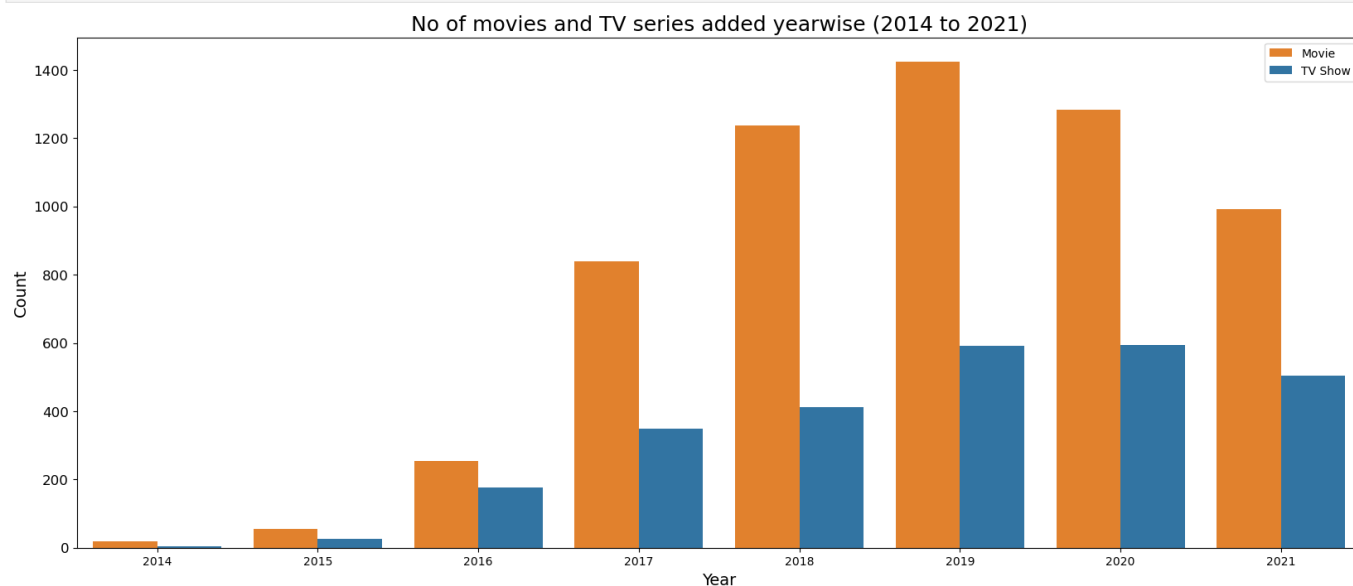
	type	date_added_year	title
6	Movie	2014	19
7	Movie	2015	56
8	Movie	2016	253
9	Movie	2017	839
10	Movie	2018	1237
11	Movie	2019	1424
12	Movie	2020	1284
13	Movie	2021	993
16	TV Show	2014	5
17	TV Show	2015	26
18	TV Show	2016	176
19	TV Show	2017	349
20	TV Show	2018	412
21	TV Show	2019	592
22	TV Show	2020	595
23	TV Show	2021	505

```
In [95]: plt.figure(figsize=(20,8))
sns.barplot(x= "date_added_year", y= "title", data = df_datetime_year_last_7years, hue="type")
plt.xlabel('Year', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.yticks(fontsize=12)

colors = ["#FF7F0F", "#1F77B5"] # Custom colors for movies and TV shows bars
sns.set_palette(sns.color_palette(colors))

plt.title("No of movies and TV series added yearwise (2014 to 2021)", fontsize=18)

plt.legend()
plt.show()
```



```
In [96]: TVshows = df_datetime_year[df_datetime_year['type'] == 'TV Show']
Movie = df_datetime_year[df_datetime_year['type'] == 'Movie']

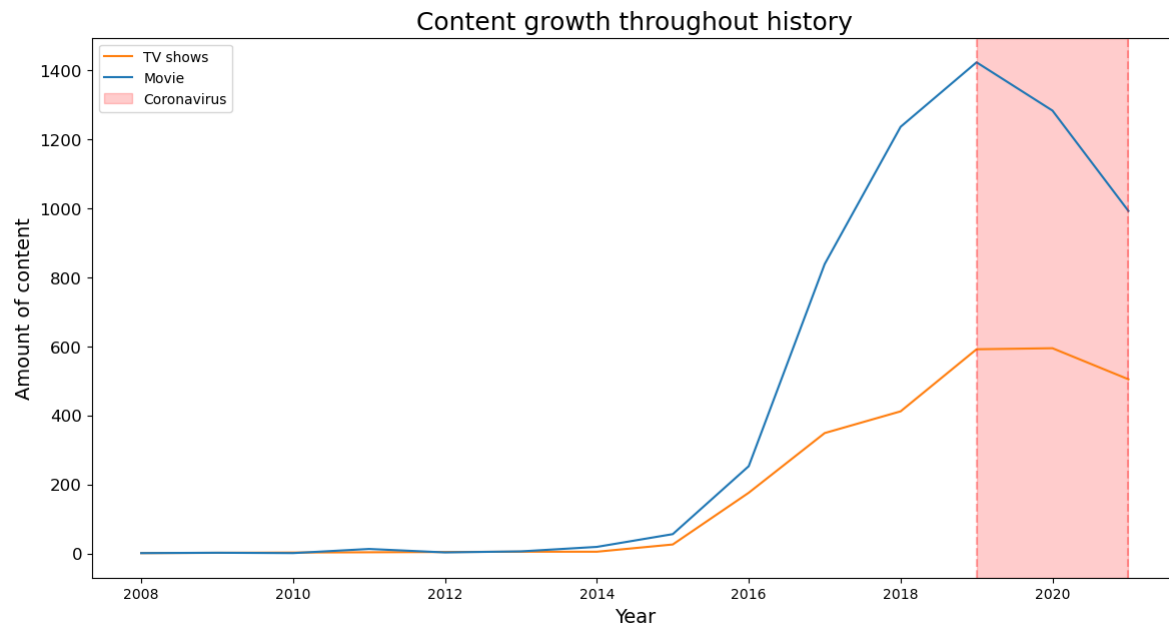
TVshows_progress = TVshows['date_added_year']
Movie_progress = Movie['date_added_year']

plt.figure(figsize=(14, 7))

plt.plot(TVshows_progress, TVshows.title, label='TV shows')
plt.plot(Movie_progress, Movie.title, label='Movie')

plt.axvline(2019, alpha=0.3, linestyle='--', color='r')
plt.axvline(2021, alpha=0.3, linestyle='--', color='r')
plt.axvspan(2019, 2021, alpha=0.2, color='r', label='Coronavirus')

# plt.xticks(list(range(1925, 2026, 5)), fontsize=12)
plt.title('Content growth throughout history', fontsize=18)
plt.xlabel('Year', fontsize=14)
plt.ylabel('Amount of content', fontsize=14)
plt.yticks(fontsize=12)
plt.legend()
plt.show()
```



Observation

- The no. of content on Netflix increased drastically from 2015 onwards
- 2019 is the year when Netflix added the maximum no. of TV shows and movies
- We can also see that after 2019, the no. of contents decreased, mostly due to COVID pandemic.

In [97]: `df_new.head(5)`

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows	date_added_month	date_added_year	date
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	90	0	9.0	2021.0	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	

```
In [98]: df_datetime_day = df_new.groupby(['type', 'date_added_day_name']).agg({"title": "nunique"}).reset_index()

order = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]
df_datetime_day['date_added_day_name'] = pd.Categorical(df_datetime_day['date_added_day_name'], categories=order, ordered=True)
df_datetime_day = df_datetime_day.sort_values(['type', 'date_added_day_name'])

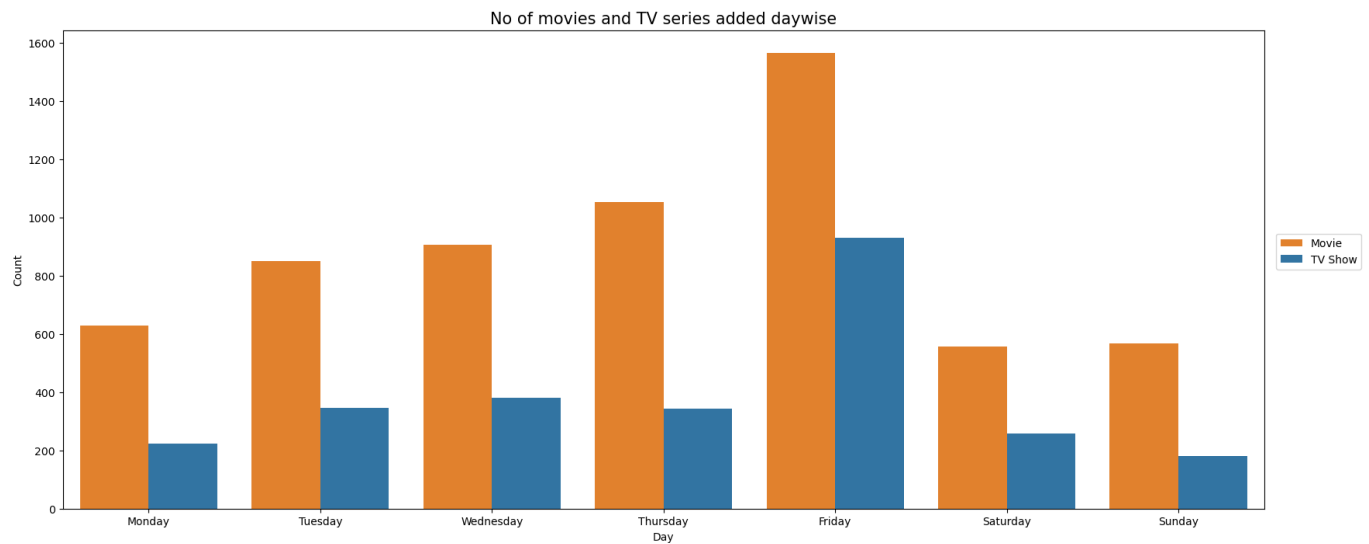
df_datetime_day
```

	type	date_added_day_name	title
1	Movie	Monday	628
5	Movie	Tuesday	852
6	Movie	Wednesday	906
4	Movie	Thursday	1053
0	Movie	Friday	1566
2	Movie	Saturday	557
3	Movie	Sunday	569
8	TV Show	Monday	223
12	TV Show	Tuesday	345
13	TV Show	Wednesday	382
11	TV Show	Thursday	343
7	TV Show	Friday	932
9	TV Show	Saturday	259
10	TV Show	Sunday	182


```
In [99]: plt.figure(figsize=(20,8))
sns.barplot(x= "date_added_day_name", y= "title", data = df_datetime_day, hue="type")
plt.xlabel('Day')
plt.ylabel('Count')

colors = ["#FF7F0F", "#1F77B5"] # Custom colors for movies and TV shows bars
sns.set_palette(sns.color_palette(colors))

plt.title("No of movies and TV series added daywise", fontsize=15)
plt.legend(loc=(1.01,0.5))
plt.show()
```



Observation

Number of content added on Netflix on "Friday" is the max. followed by Thursday as weekend approaches after these days. New content on Monday, Tuesday are lower mostly because it's the start of the week, and we expect people would be more engrossed with work/school/college

Analysis of TV Shows and Movies over Genre

```
In [100]: df_new.head()
```

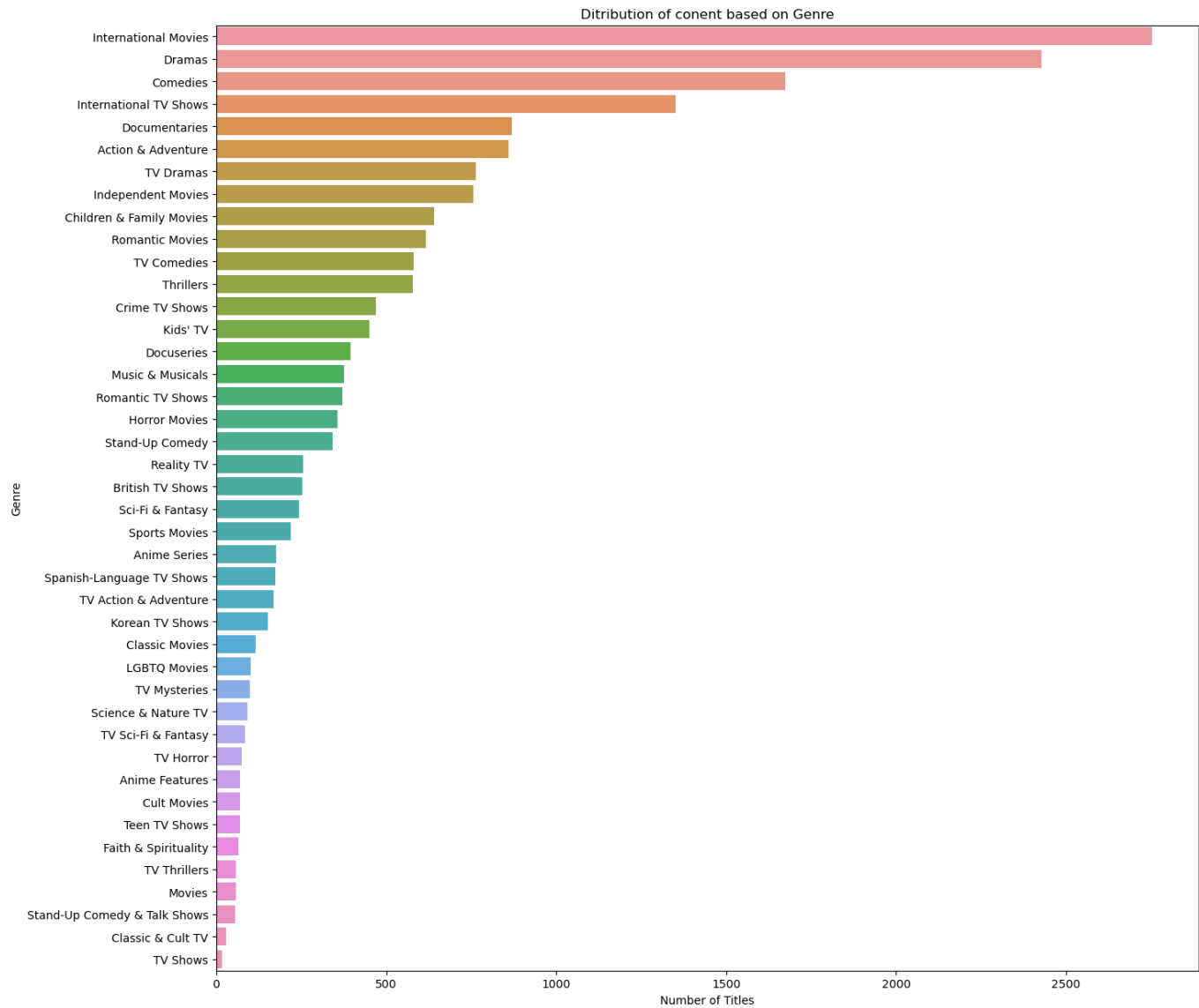
	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows	date_added_month	date_added_year	date
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	90	0	9.0	2021.0	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	

```
In [101]: #number of distinct titles on the basis of duration
genre_title_count = df_new.groupby(['Genre']).agg({"title": "nunique"}).sort_values(by=['title'], ascending=False).reset_index()
genre_title_count.columns = ['Genre', 'Title']
```

```
In [102]: # Plotting a horizontal bar chart
plt.figure(figsize=(40, 15))
sns.barplot(x='Title', y='Genre', data=genre_title_count)

# Set plot title and axis labels
plt.title('Distribution of content based on Genre')
plt.xlabel('Number of Titles')
plt.ylabel('Genre')

plt.subplots_adjust(left=0.6)
```



Observation

Most appearing category in netflix movies and TV shows are:

- International Movies
- Dramas
- Comedies
- International TV show
- Documentaries

Non-Graphical Analysis (Top Actors/Directors in different countries)

```
In [103... df_new.head()
```

Out[103]:

	title	Actors	Director	Genre	Country	show_id	type	date_added	release_year	rating	duration	description	duration_movies	duration_tv_shows	date_added_month	date_added_year	dat
0	Dick Johnson Is Dead	Unknown Actor	Kirsten Johnson	Documentaries	United States	s1	Movie	2021-09-25	2020	PG-13	90 min	As her father nears the end of his life, filmm...	90	0	9.0	2021.0	
1	Blood & Water	Ama Qamata	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
2	Blood & Water	Ama Qamata	Unknown Director	TV Dramas	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
3	Blood & Water	Ama Qamata	Unknown Director	TV Mysteries	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	
4	Blood & Water	Khosi Ngema	Unknown Director	International TV Shows	South Africa	s2	TV Show	2021-09-24	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	0	2	9.0	2021.0	

In [104...

```
dir_country = df_new[df_new['Director'] != 'Unknown Director']
dir_country_title_count = dir_country.groupby(['Country', 'Director']).agg({"title": "nunique"}).sort_values(by=['title', 'Country'], ascending=False).reset_index()
```

In [105...

```
# all_countries = dir_country_title_count['Country'].unique()

# for country in all_countries:
#     print("*****")
#     print(f"Top Directors from the country : {country}")
#     print("*****")
#     top_dir = dir_country_title_count[dir_country_title_count['Country'] == country]
#     print(top_dir.iloc[0:2, [1,2]].to_string(index=False))
#     print()
```

In [106...

```
countries = ['India', 'United States', 'United Kingdom', 'France']

for country in countries:
    print("*****")
    print(f"Top Directors from the country : {country}")
    print("*****")
    top_dir = dir_country_title_count[dir_country_title_count['Country'] == country]

    print(top_dir.iloc[0:2, [1,2]].to_string(index=False))
    print()
```

```
*****
Top Directors from the country : India
*****
      Director  title
0  Anurag Kashyap    9
1   David Dhawan    9

*****
Top Directors from the country : United States
*****
      Director  title
0        Jay Karas   15
1   Marcus Raboy   15

*****
Top Directors from the country : United Kingdom
*****
      Director  title
0 Alastair Fothergill    4
1   Edward Catterill    4

*****
Top Directors from the country : France
*****
      Director  title
0  Thierry Donard     5
1  Yousef Chahine     4
```

Observation

- Anurag Kashyap and David Dhawan are the most famous directors for Inida.
- Jay Karas and Marcus Raboyare are the most famous directors in United States.
- Alastair Fothergill and Edward Cotterill are the most famous directors in UK.

In [107...

```
actor_country = df_new[df_new['Actors'] != 'Unknown Actor']
actor_country_title_count = actor_country.groupby(['Country', 'Actors']).agg({"title": "nunique"}).sort_values(by=['title', 'Country'], ascending=False).reset_index()
```

In [108...

```
actor_country_title_count
```

Out[108]:

	Country	Actors	title
0	India	Anupam Kher	40
1	India	Shah Rukh Khan	34
2	India	Naseeruddin Shah	31
3	Japan	Takahiro Sakurai	29
4	India	Akshay Kumar	29
...
50884		Nisreen Faour	1
50885		Son Suk-ku	1
50886		Souad Massi	1
50887		Suhail Haddad	1
50888		Walid Abdul Salam	1

50889 rows x 3 columns

```
In [109.. # all_countries = actor_country_title_count['Country'].unique()

# for country in all_countries:
#     print("*****")
#     print(f"Top Actors from the country : {country}")
#     print("*****")
#     top_dir = actor_country_title_count[actor_country_title_count['Country'] == country]

#     print(top_dir.iloc[0:2, [1,2]].to_string(index=False))
#     print()
```

```
In [110.. countries = ['India', 'United States', 'United Kingdom', 'France']

for country in countries:
    print("*****")
    print(f"Top Actors from the country : {country}")
    print("*****")
    top_dir = actor_country_title_count[actor_country_title_count['Country'] == country]

    print(top_dir.iloc[0:2, [1,2]].to_string(index=False))
    print()
```

```
*****
Top Actors from the country : India
*****
  Actors  title
Anupam Kher    40
Shah Rukh Khan  34

*****
Top Actors from the country : United States
*****
  Actors  title
Samuel L. Jackson    22
Tara Strong          22

*****
Top Actors from the country : United Kingdom
*****
  Actors  title
David Attenborough    17
John Cleese           16

*****
Top Actors from the country : France
*****
  Actors  title
Benoit Magimel     5
Wille Lindberg     5
```

Observation

- Anupam Kher and Shah Rukh Khan are the most famous actors for India.
- Samuel L. Jackson and Tara Strong are the most famous actors for USA.
- David Attenboroug and John Cleese are the most famous actors for UK.

```
In [111.. top_20_actors = actor_country_title_count.sort_values('title', ascending=False)[:20]

print("*****")
print("      Top 20 actors across the globe")
print("*****")

print(top_20_actors)
print("*****")
```

```
*****
Top 20 actors across the globe
*****
```

	Country	Actors	title
0	India	Anupam Kher	40
1	India	Shah Rukh Khan	34
2	India	Naseeruddin Shah	31
3	Japan	Takahiro Sakurai	29
4	India	Akshay Kumar	29
5	India	Om Puri	29
6	Japan	Yuki Kaji	28
7	India	Amitabh Bachchan	28
8	India	Pareesh Rawal	28
9	India	Boman Irani	27
10	India	Kareena Kapoor	25
11	United States	Samuel L. Jackson	22
12	United States	Tara Strong	22
13	Japan	Daisuke Ono	22
14	United States	Fred Tatasciore	21
15	India	Ajay Devgn	21
17	India	Salman Khan	20
16	United States	Adam Sandler	20
18	United States	James Franco	19
19	United States	Nicolas Cage	19

```
*****
```

Heatmaps (Country v/s Ratings)

```
In [112]: #number of distinct titles on the basis of rating
rating_country_title_count = df_new.groupby(['Country', 'rating']).agg({'title': "nunique"}).sort_values(by=['title'],
                                                                                                     ascending=False).reset_index()
rating_country_title_count.columns = ['Country', 'Rating', 'Count']
```

```
In [113]: rating_country_title_count
```

```
Out[113]:
```

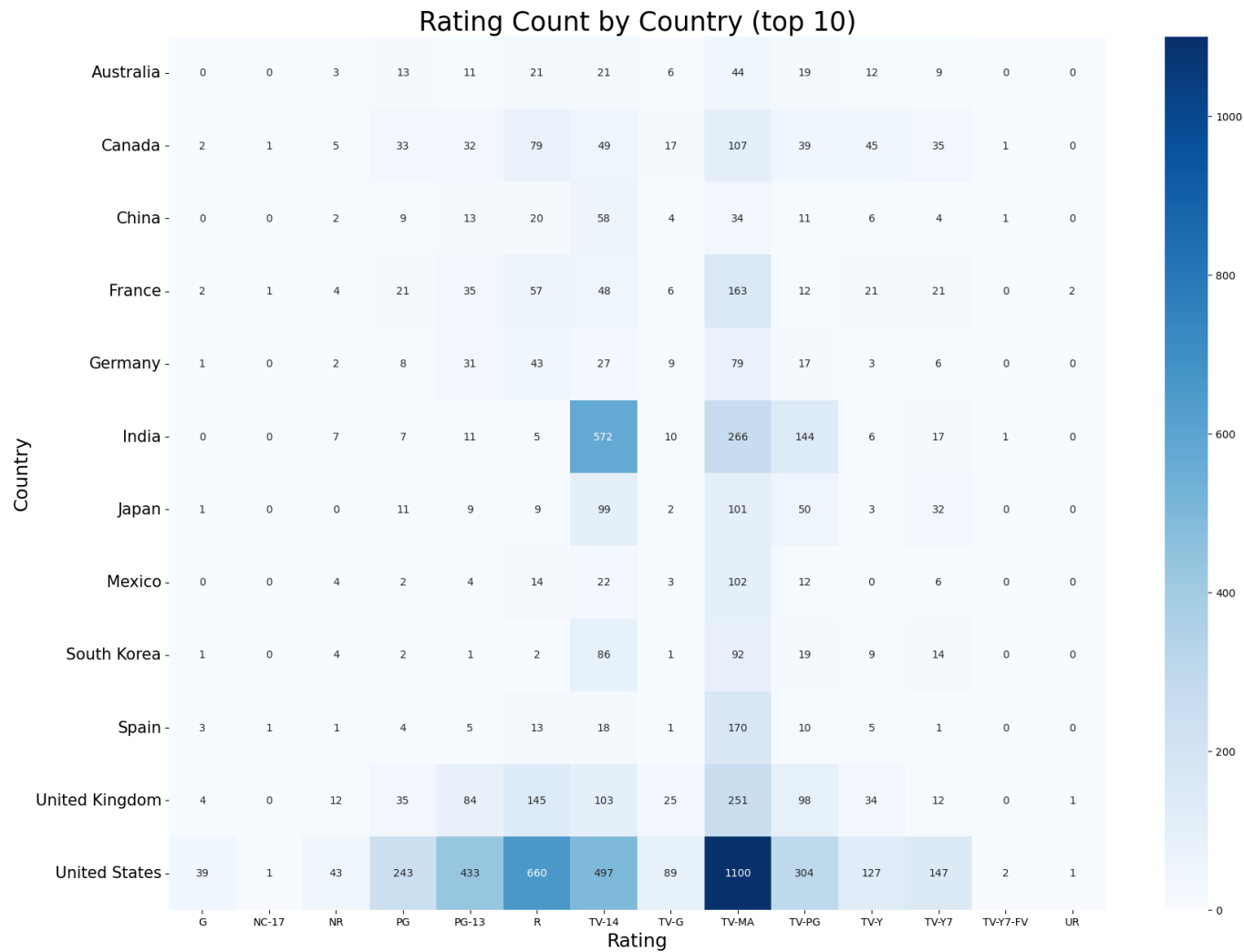
	Country	Rating	Count
0	United States	TV-MA	1100
1	United States	R	660
2	India	TV-14	572
3	United States	TV-14	497
4	United States	PG-13	433
...
511	Malawi	TV-PG	1
512	Malaysia	TV-Y	1
513	Malta	PG-13	1
514	Malta	R	1
515	Zimbabwe	TV-MA	1

516 rows x 3 columns

```
In [114]: to_10_countries_names = rating_country_title_count.groupby('Country').agg({'Count': 'sum'}).sort_values(by=['Count'],
                                                                                                     ascending=False).reset_index().loc[:11, 'Country']
heatmap_data = rating_country_title_count[rating_country_title_count['Country'].isin(to_10_countries_names)]
heatmap_data = heatmap_data.pivot(index='Country', columns='Rating', values='Count')
heatmap_data = heatmap_data.fillna(0).astype('int')
plt.figure(figsize=(20, 15))

sns.heatmap(heatmap_data, annot=True, cmap="Blues", fmt = "d")
plt.title('Rating Count by Country (top 10)', fontsize=25)
plt.xlabel('Rating', fontsize=18)
plt.ylabel('Country', fontsize=18)
plt.yticks(fontsize=15)

plt.show()
```



Observation

- Top 10 countries are having most content that belongs to TV-MA category
- India and United States are having large content in TV-14 category
- United Kingdom and United States are having large content in R category
- Very less no. of TV shows which are meant for Children (TV-Y and TV-Y7)

Heatmaps (Country v/s Genre)

```
In [115]: #number of distinct titles on the basis of rating
genre_country_title_count = df_new.groupby(['Country', 'Genre']).agg({"title": "nunique").sort_values(by=['title'],
                                                    ascending=False).reset_index()
genre_country_title_count.columns = ['Country', 'Genre', 'Count']

In [116]: genre_country_title_count

Out[116]:
   Country  Genre  Count
0      India  International Movies  864
1  United States  Dramas  835
2  United States  Comedies  680
3      India  Dramas  662
4  United States  Documentaries  511
...      ...      ...      ...
1417  Mauritius  International TV Shows  1
1418  Mauritius  TV Dramas  1
1419  Mexico  Classic Movies  1
1420  Mexico  Faith & Spirituality  1
1421  Zimbabwe  Romantic Movies  1

1422 rows x 3 columns

In [117]: top_10_countries_names = genre_country_title_count.groupby('Country').agg({'Count': 'sum'}).sort_values(by=['Count'],
                                                    ascending=False).reset_index().loc[:11, 'Country']

top_10_genre_names = genre_country_title_count.groupby('Genre').agg({'Count': 'sum'}).sort_values(by=['Count'],
                                                    ascending=False).reset_index().loc[:11, 'Genre']

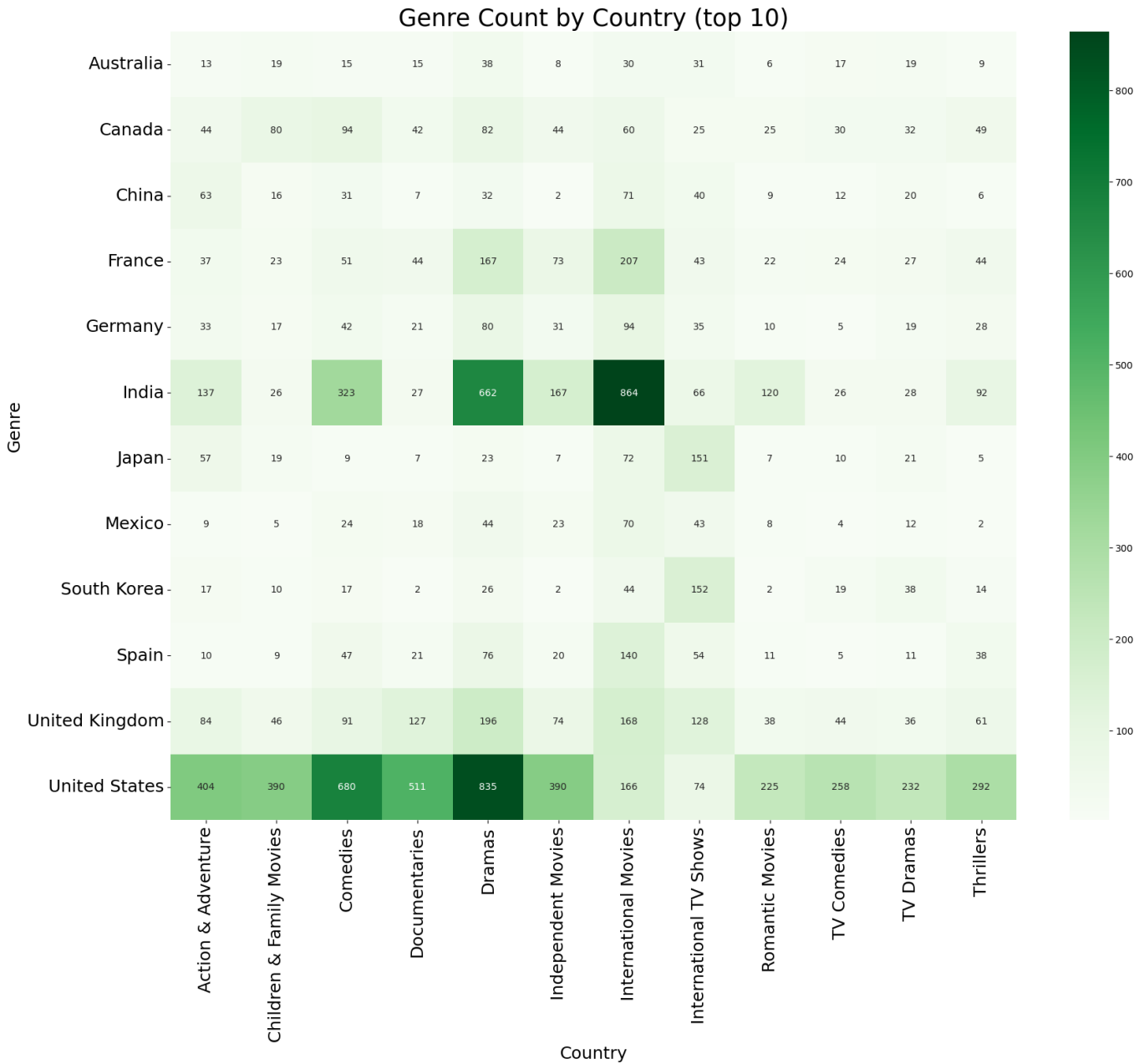
top_10_genre = genre_country_title_count[genre_country_title_count['Genre'].isin(top_10_genre_names)]
```

```
top_10_countries = top_10_genre[top_10_genre['Country'].isin(top_10_countries_names)]

heatmap_data = top_10_countries.pivot(index='Country', columns='Genre', values='Count')
heatmap_data = heatmap_data.fillna(0).astype('int')
plt.figure(figsize=(20, 15))

sns.heatmap(heatmap_data, annot=True, cmap="Greens", fmt = "d")
plt.title('Genre Count by Country (top 10)', fontsize=25)
plt.xlabel('Country', fontsize=18)
plt.ylabel('Genre', fontsize=18)
plt.yticks(fontsize=18)
plt.xticks(fontsize=18)

plt.show()
```



Observation

- For United States , Netflix should add more content of genre Dramas and Comedy.
- For India, Netflix should add more content of genre International movies , Dramas and Comedies.

End of Analysis