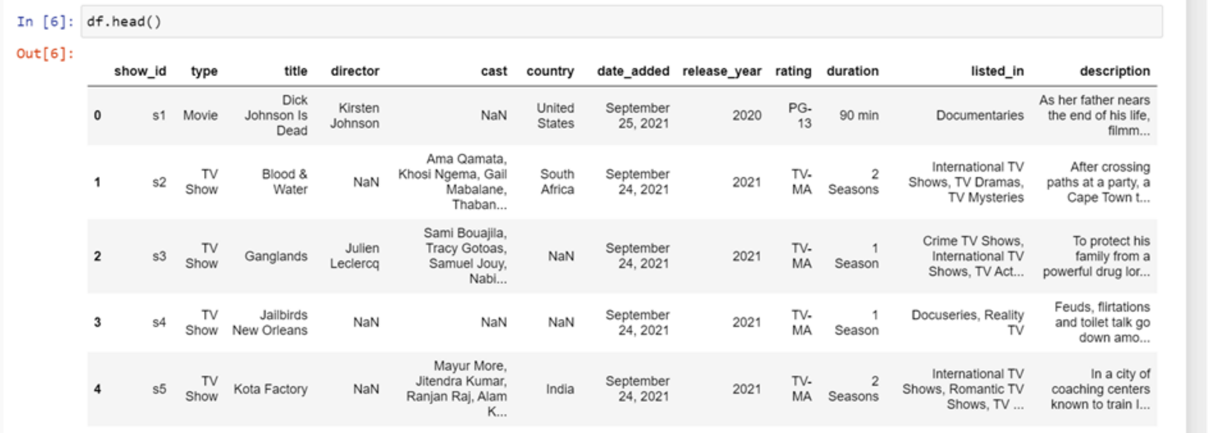
1. Problem Statement

Netflix wants to increase its growth and wants to decide which type of shows/movies to produce. Also, find other strategies to push the growth.

2. The shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required), missing value detection, statistical

2.1 The shape of the data frame is 8807x12

2.2 Used head function to observe all the columns

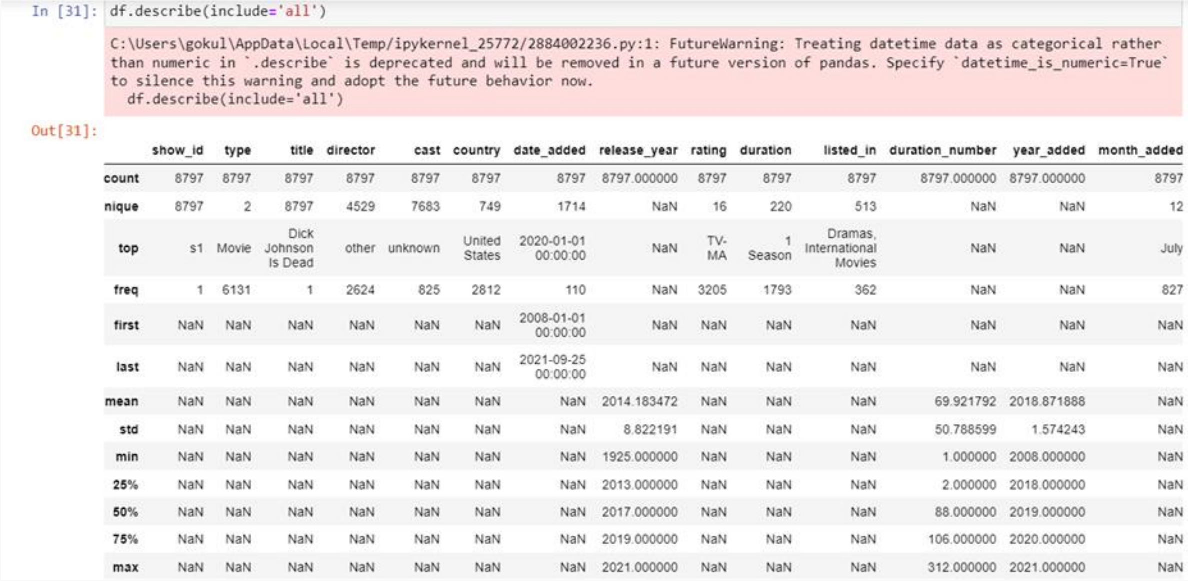


```
In [6]: df.head()
```

Out[6]:

	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 l...

2.3 Used describe function to a statistical summary of all the rows.



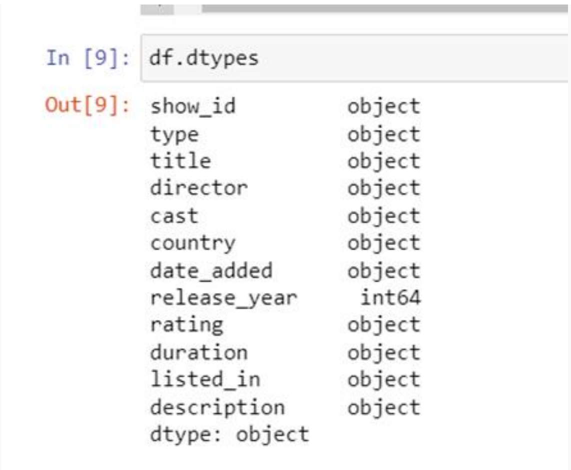
```
In [31]: df.describe(include='all')
```

```
C:\Users\gokul\AppData\Local\Temp/ipykernel_25772/2884002236.py:1: FutureWarning: Treating datetime data as categorical rather
than numeric in `.describe` is deprecated and will be removed in a future version of pandas. Specify `datetime_is_numeric=True`
to silence this warning and adopt the future behavior now.
  df.describe(include='all')
```

Out[31]:

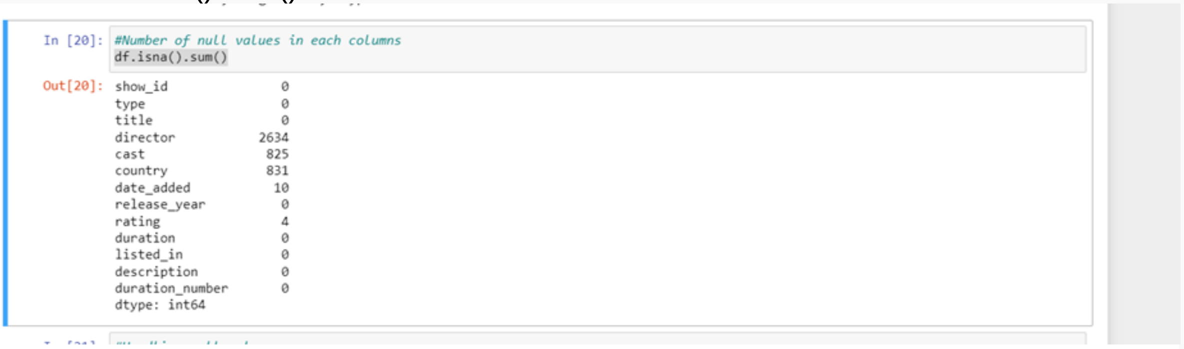
	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	duration_number	year_added	month_added
count	8797	8797	8797	8797	8797	8797	8797	8797.000000	8797	8797	8797	8797.000000	8797.000000	8797
nique	8797	2	8797	4529	7683	749	1714	NaN	16	220	513	NaN	NaN	12
top	s1	Movie	Dick Johnson Is Dead	other	unknown	United States	2020-01-01 00:00:00	NaN	TV-MA	1 Season	Dramas, International Movies	NaN	NaN	July
freq	1	6131	1	2624	825	2812	110	NaN	3205	1793	362	NaN	NaN	827
first	NaN	NaN	NaN	NaN	NaN	NaN	2008-01-01 00:00:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN
last	NaN	NaN	NaN	NaN	NaN	NaN	2021-09-25 00:00:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mean	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2014.183472	NaN	NaN	NaN	69.921792	2018.871888	NaN
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	8.822191	NaN	NaN	NaN	50.788599	1.574243	NaN
min	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1925.000000	NaN	NaN	NaN	1.000000	2008.000000	NaN
25%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2013.000000	NaN	NaN	NaN	2.000000	2018.000000	NaN
50%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2017.000000	NaN	NaN	NaN	88.000000	2019.000000	NaN
75%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2019.000000	NaN	NaN	NaN	106.000000	2020.000000	NaN
max	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2021.000000	NaN	NaN	NaN	312.000000	2021.000000	NaN

2.4 Used dtypes function to observe the data types of each column



```
In [9]: df.dtypes

Out[9]: show_id         object
        type            object
        title           object
        director        object
        cast            object
        country         object
        date_added      object
        release_year     int64
        rating          object
        duration        object
        listed_in       object
        description     object
        dtype: object
```

2.5 Used df.isna().sum()to calculate the number of null values in each column.

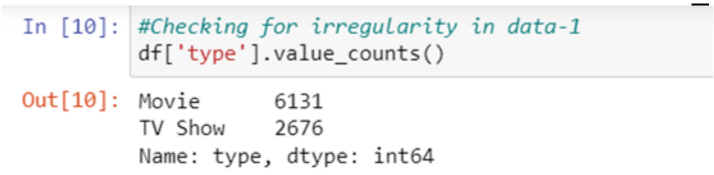


```
In [20]: #Number of null values in each columns
         df.isna().sum()

Out[20]: show_id              0
         type                 0
         title                0
         director          2634
         cast               825
         country            831
         date_added          10
         release_year         0
         rating               4
         duration             0
         listed_in            0
         description          0
         duration_number      0
         dtype: int64
```

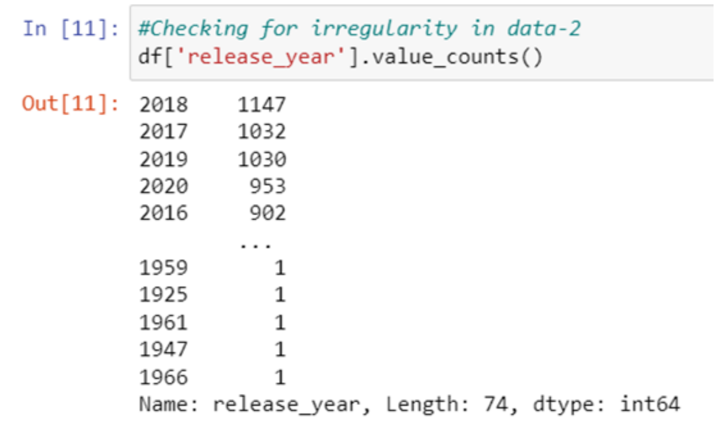
Note: I have done some data manipulation because of which the number of null values has reduced.

3. Non-Graphical Analysis and Missing Value & Outlier check

Each columns are check with function value_count().

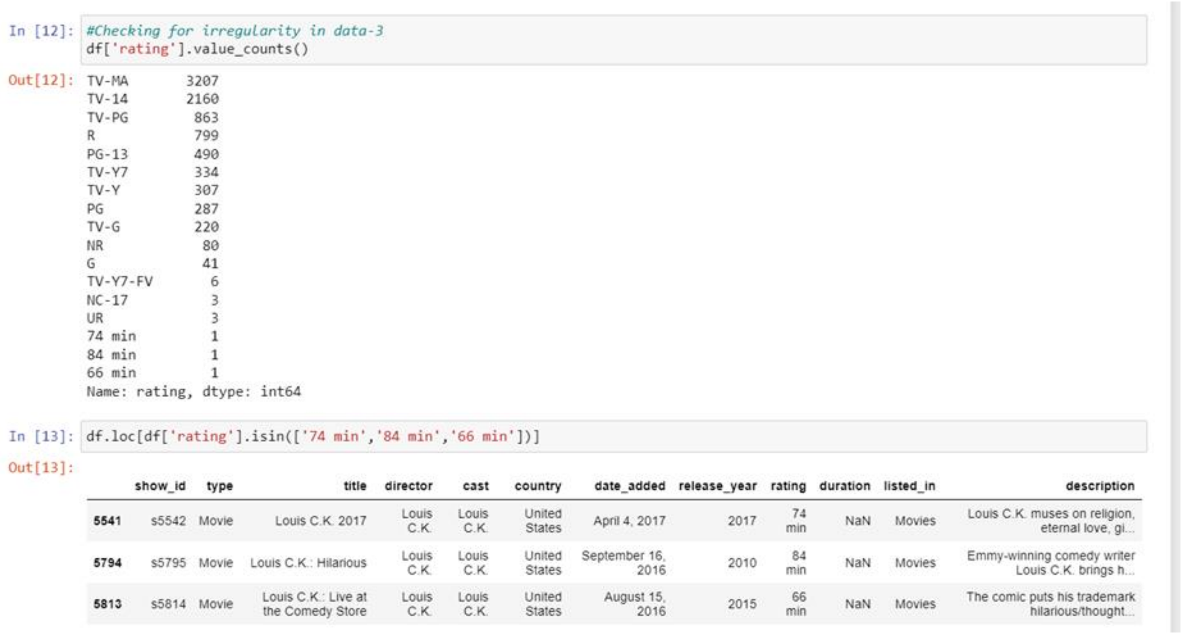


```
In [10]: #Checking for irregularity in data-1
         df['type'].value_counts()

Out[10]: Movie      6131
         TV Show    2676
         Name: type, dtype: int64
```



```
In [11]: #Checking for irregularity in data-2
         df['release_year'].value_counts()

Out[11]: 2018    1147
         2017    1032
         2019    1030
         2020     953
         2016     902
                 ...
         1959       1
         1925       1
         1961       1
         1947       1
         1966       1
         Name: release_year, Length: 74, dtype: int64
```

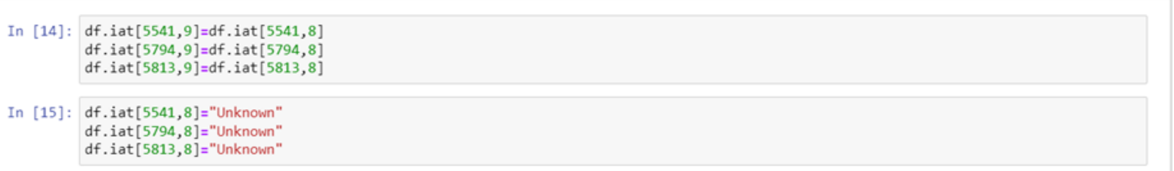
In type, release_year column no problem was found.



```
In [12]: #Checking for irregularity in data-3
         df['rating'].value_counts()

Out[12]: TV-MA       3207
         TV-14       2160
         TV-PG        863
         R            799
         PG-13        490
         TV-Y7        334
         TV-Y         307
         PG           287
         TV-G         220
         NR            80
         G             41
         TV-Y7-FV       6
         NC-17          3
         UR             3
         74 min         1
         84 min         1
         66 min         1
         Name: rating, dtype: int64

In [13]: df.loc[df['rating'].isin(['74 min','84 min','66 min'])]

Out[13]:
```

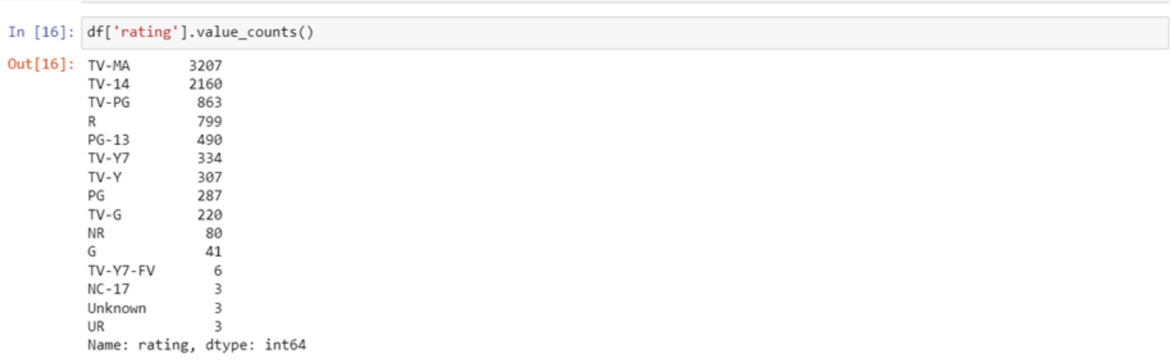
	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
5541	s5542	Movie	Louis C.K. 2017	Louis C.K.	Louis C.K.	United States	April 4, 2017	2017	74 min	NaN	Movies	Louis C.K. muses on religion, eternal love, gi...
5794	s5795	Movie	Louis C.K.: Hilarious	Louis C.K.	Louis C.K.	United States	September 16, 2016	2010	84 min	NaN	Movies	Emmy-winning comedy writer Louis C.K. brings h...
5813	s5814	Movie	Louis C.K.: Live at the Comedy Store	Louis C.K.	Louis C.K.	United States	August 15, 2016	2015	66 min	NaN	Movies	The comic puts his trademark hilarious/thought...

But in the rating column, 3 row has a duration in it. So that 3 rows were identified. In the same row, duration was missing.



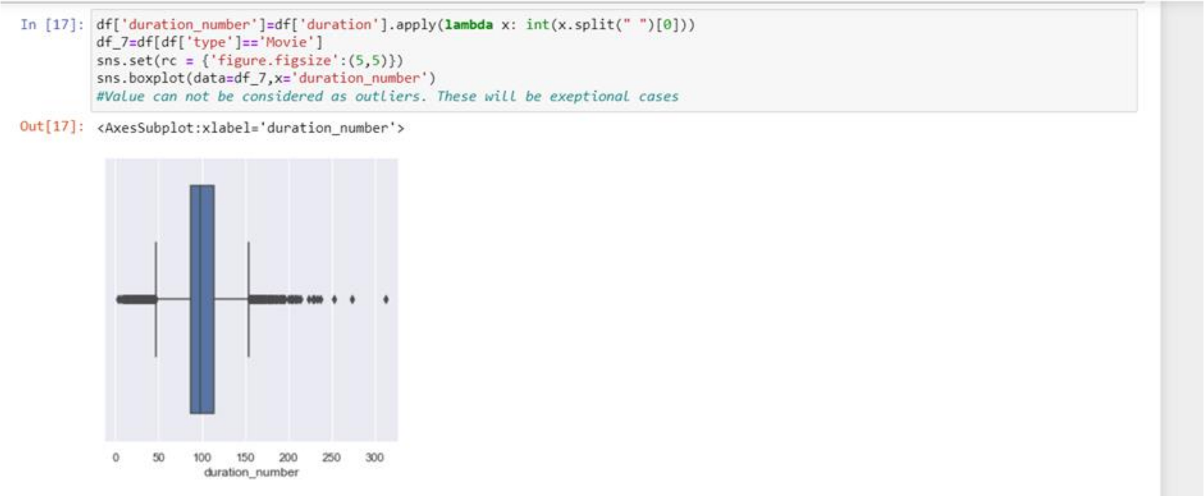
```
In [14]: df.iat[5541,9]=df.iat[5541,8]
         df.iat[5794,9]=df.iat[5794,8]
         df.iat[5813,9]=df.iat[5813,8]

In [15]: df.iat[5541,8]="Unknown"
         df.iat[5794,8]="Unknown"
         df.iat[5813,8]="Unknown"
```

So, the duration was transferred from the rating column to the duration column and the rating was marked as unknown.



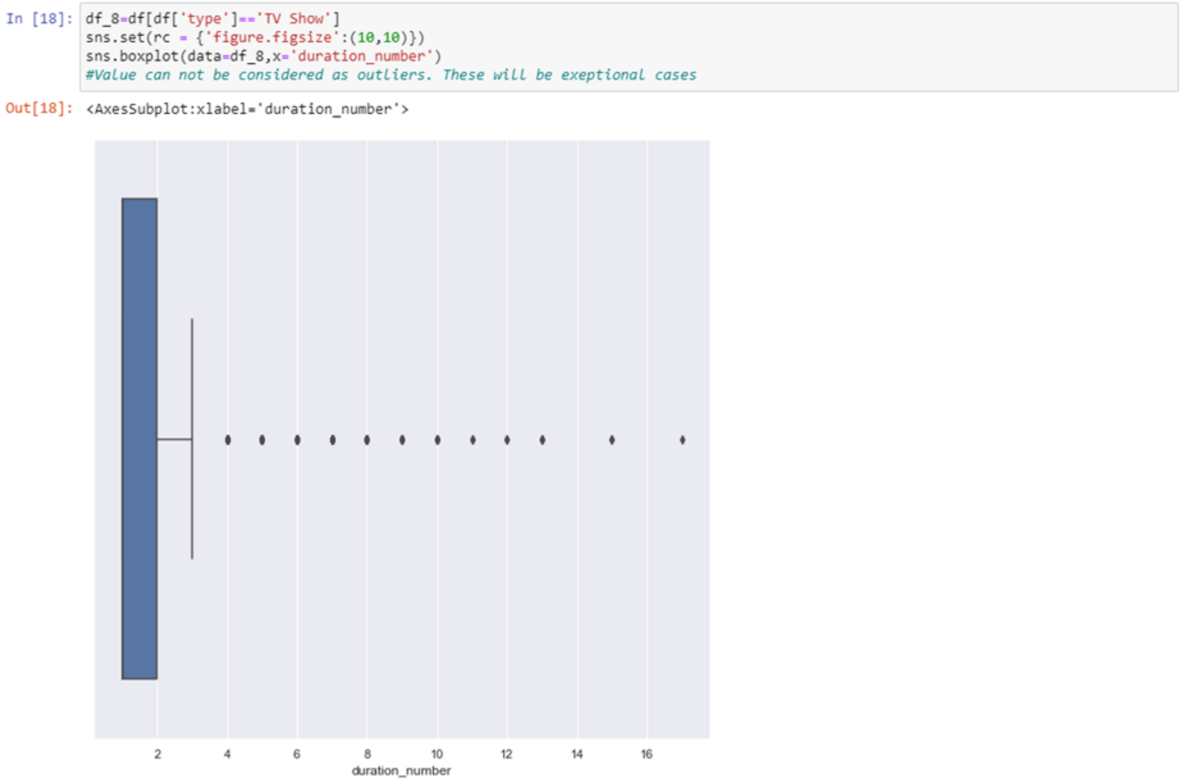
```
In [16]: df['rating'].value_counts()

Out[16]: TV-MA       3207
         TV-14       2160
         TV-PG        863
         R            799
         PG-13        490
         TV-Y7        334
         TV-Y         307
         PG           287
         TV-G         220
         NR            80
         G             41
         TV-Y7-FV       6
         NC-17          3
         Unknown        3
         UR             3
         Name: rating, dtype: int64
```

Again, the rating column was checked once again to ensure the removal of durations.



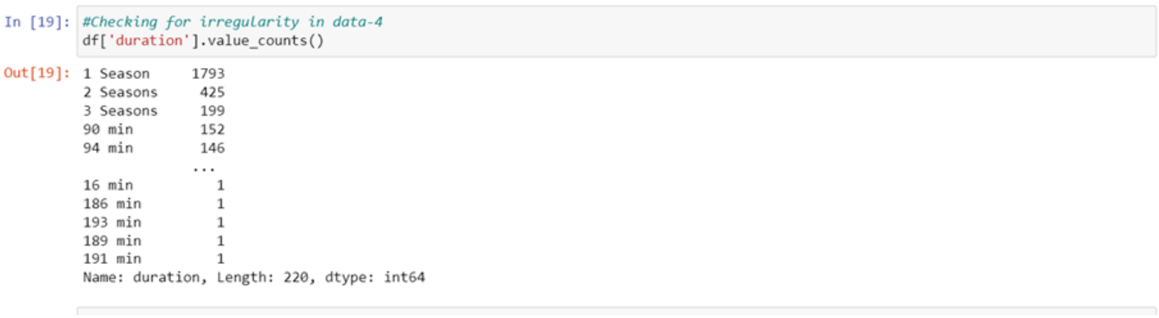
```
In [17]: df['duration_number']=df['duration'].apply(lambda x: int(x.split(" ")[0]))
         df_7=df[df['type']=='Movie']
         sns.set(rc = {'figure.figsize':(5,5)})
         sns.boxplot(data=df_7,x='duration_number')
         #Value can not be considered as outliers. These will be exceptional cases

Out[17]: <AxesSubplot:xlabel='duration_number'>
```

To check the outliners in the duration columns box for Movies and TV shows separately. The plot showed many outliers. They canâ€™t to treated as outliers and removed since there is much value and has to be considered exceptional cases.



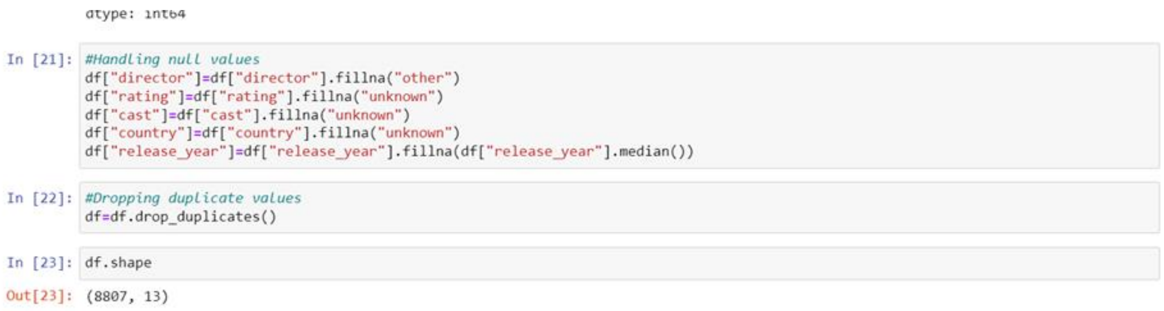
```
In [18]: df_8=df[df['type']=='TV Show']
         sns.set(rc = {'figure.figsize':(10,10)})
         sns.boxplot(data=df_8,x='duration_number')
         #Value can not be considered as outliers. These will be exceptional cases

Out[18]: <AxesSubplot:xlabel='duration_number'>
```

Same is the case with TV Shows. They canâ€™t to treated as outliers and removed since there is much value and has to be considered exceptional cases.



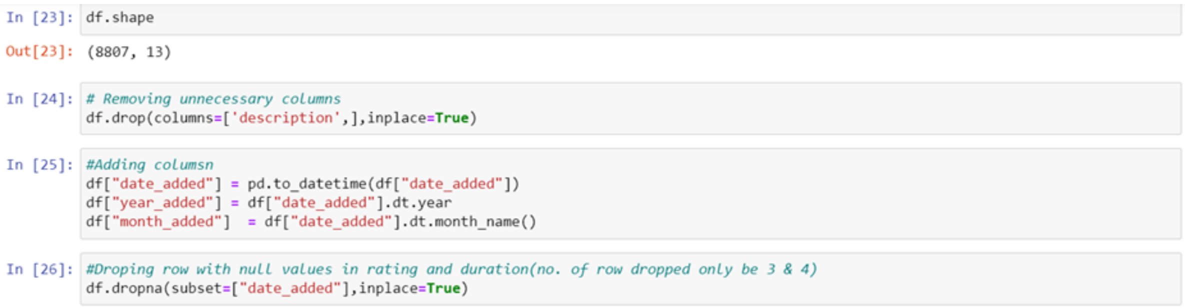
```
In [19]: #Checking for irregularity in data-4
         df['duration'].value_counts()

Out[19]: 1 Season     1793
         2 Seasons     425
         3 Seasons     199
         90 min        152
         94 min        146
                      ...
         16 min          1
         186 min         1
         193 min         1
         189 min         1
         191 min         1
         Name: duration, Length: 220, dtype: int64
```

Value count of duration was checked for any irregularities and found none.



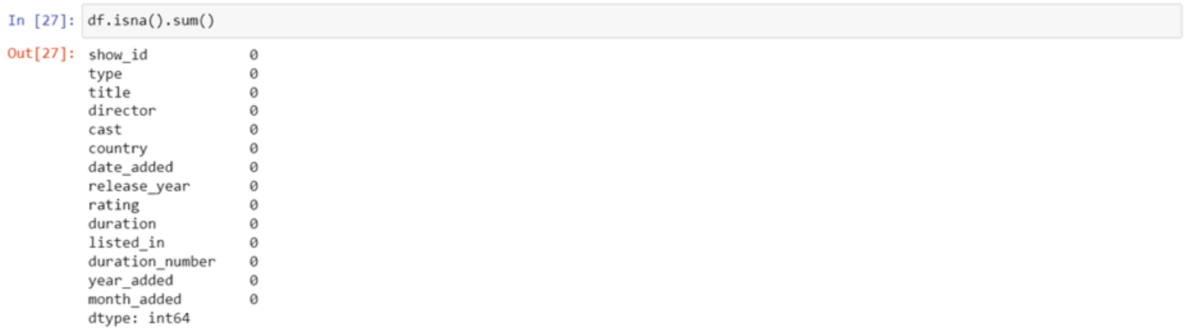
```
         dtype: int64

In [21]: #Handling null values
         df["director"]=df["director"].fillna("other")
         df["rating"]=df["rating"].fillna("unknown")
         df["cast"]=df["cast"].fillna("unknown")
         df["country"]=df["country"].fillna("unknown")
         df["release_year"]=df["release_year"].fillna(df["release_year"].median())

In [22]: #Dropping duplicate values
         df=df.drop_duplicates()

In [23]: df.shape

Out[23]: (8807, 13)
```

All the null values were replaced with as shown above.



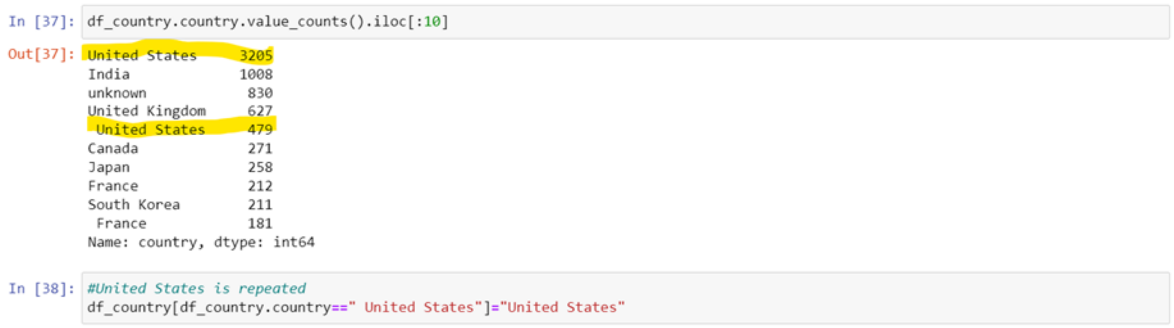
```
In [23]: df.shape

Out[23]: (8807, 13)

In [24]: # Removing unnecessary columns
         df.drop(columns=['description',],inplace=True)

In [25]: #Adding columsn
         df["date_added"] = pd.to_datetime(df["date_added"])
         df["year_added"] = df["date_added"].dt.year
         df["month_added"]  = df["date_added"].dt.month_name()

In [26]: #Droping row with null values in rating and duration(no. of row dropped only be 3 & 4)
         df.dropna(subset=["date_added"],inplace=True)
```

Description column was dropped, the data type of date_added was converted to datetime from object ,and add 2 additional columns for year_added & month_added.



```
In [27]: df.isna().sum()

Out[27]: show_id            0
         type               0
         title              0
         director           0
         cast               0
         country            0
         date_added         0
         release_year       0
         rating             0
         duration           0
         listed_in          0
         duration_number    0
         year_added         0
         month_added        0
         dtype: int64
```

Then it was checked whether all the null values were removed.



```
In [37]: df_country.country.value_counts().iloc[:10]

Out[37]: United States     3205
         India             1008
         unknown            830
         United Kingdom     627
          United States     479
         Canada             271
         Japan              258
         France             212
         South Korea        211
          France            181
         Name: country, dtype: int64

In [38]: #United States is repeated
         df_country[df_country.country==" United States"]="United States"
```

During data pre-processing, it was found that the United States occurred twice because of extra white space in front. So, the same was corrected.

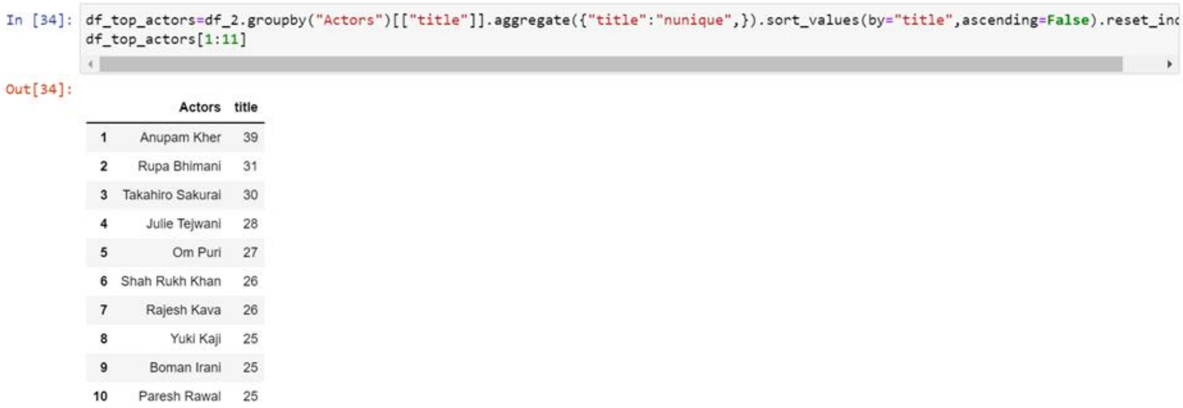
4. Prepossessing, Visual Analysis, and Insights

Cast and directors were unpacked and merged into a single data frame.

Count and listed in were unpacked and merged into a single data frame.

Note:

Every column was not merged into the same data frame due to the limitation of RAM.



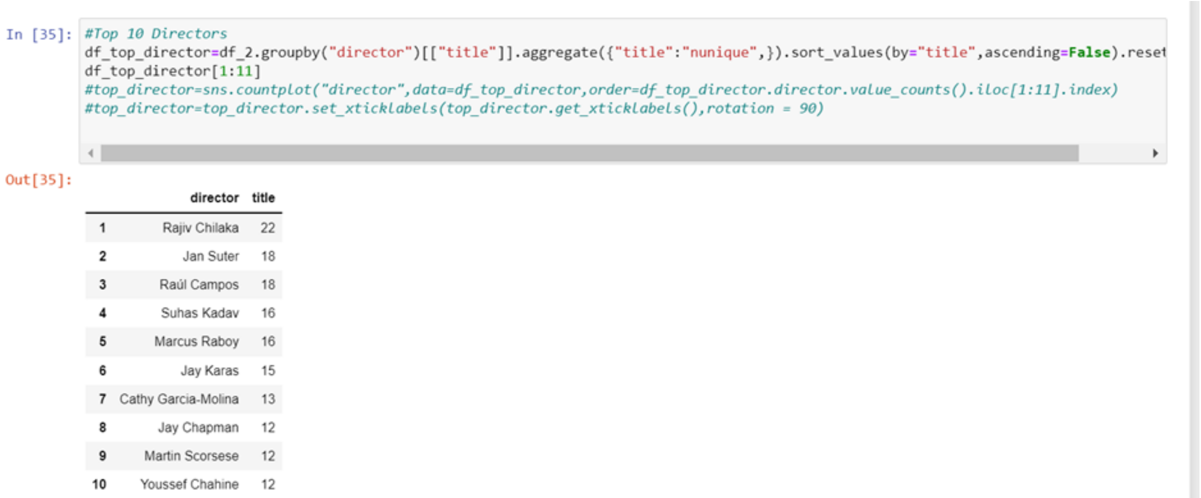
```
In [34]: df_top_actors=df_2.groupby("Actors")[["title"]].aggregate({"title":"nunique",}).sort_values(by="title",ascending=False).reset_inc
         df_top_actors[1:11]
```

Out[34]:

	Actors	title
1	Anupam Kher	39
2	Rupa Bhimani	31
3	Takahiro Sakurai	30
4	Julie Tejwani	28
5	Om Puri	27
6	Shah Rukh Khan	26
7	Rajesh Kava	26
8	Yuki Kaji	25
9	Boman Irani	25
10	Paresh Rawal	25

The top 10 actors were identified.

This table shows the top 10 actors who have acted in most contents. There is no large variation number of the content the actors have acted.



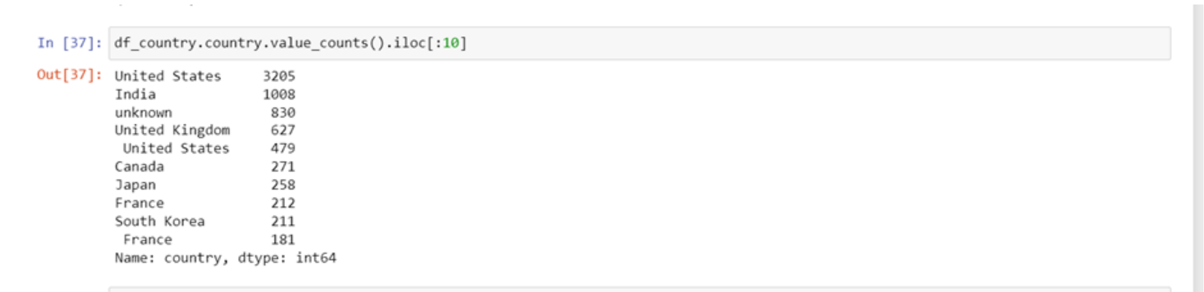
```
In [35]: #Top 10 Directors
         df_top_director=df_2.groupby("director")[["title"]].aggregate({"title":"nunique",}).sort_values(by="title",ascending=False).reset
         df_top_director[1:11]
         #top_director=sns.countplot("director",data=df_top_director,order=df_top_director.director.value_counts().iloc[1:11].index)
         #top_director=top_director.set_xticklabels(top_director.get_xticklabels(),rotation = 90)
```

Out[35]:

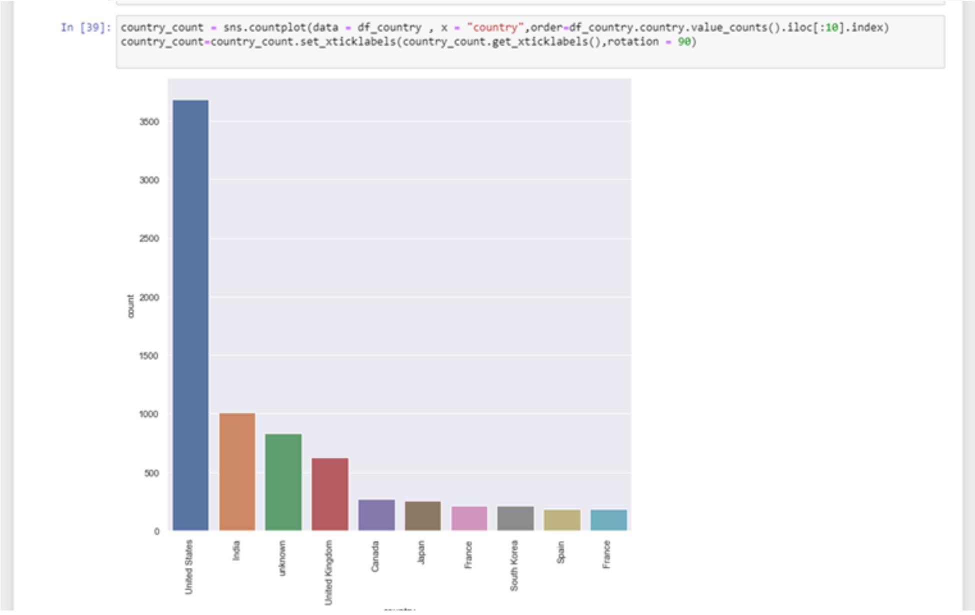
	director	title
1	Rajiv Chilaka	22
2	Jan Suter	18
3	Raúl Campos	18
4	Suhas Kadav	16
5	Marcus Raboy	16
6	Jay Karas	15
7	Cathy Garcia-Molina	13
8	Jay Chapman	12
9	Martin Scorsese	12
10	Youssef Chahine	12

The top 10 directors were identified.

This table shows the top 10 actors who have acted in most contents. Here also there is no large variation number of the content directed by each director.

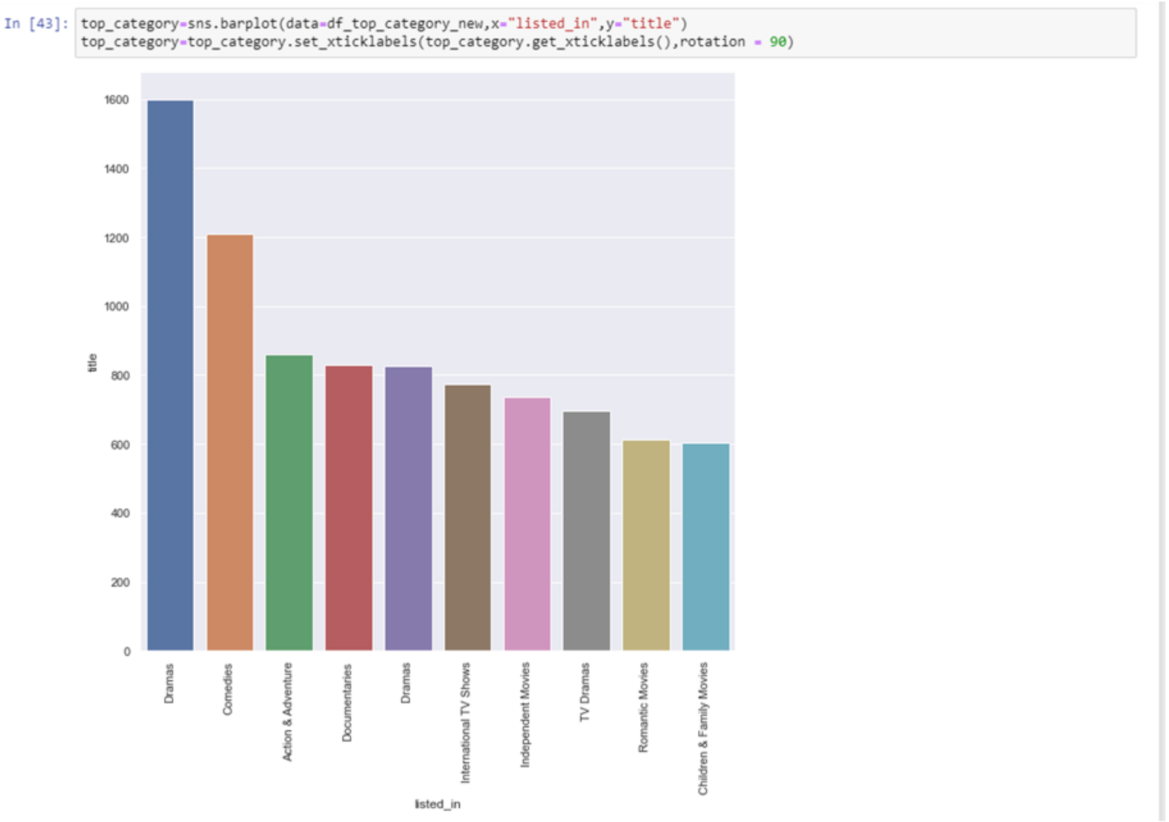


```
In [37]: df_country.country.value_counts().iloc[:10]

Out[37]: United States     3205
         India             1008
         unknown            830
         United Kingdom     627
          United States     479
         Canada             271
         Japan              258
         France             212
         South Korea        211
          France            181
         Name: country, dtype: int64
```



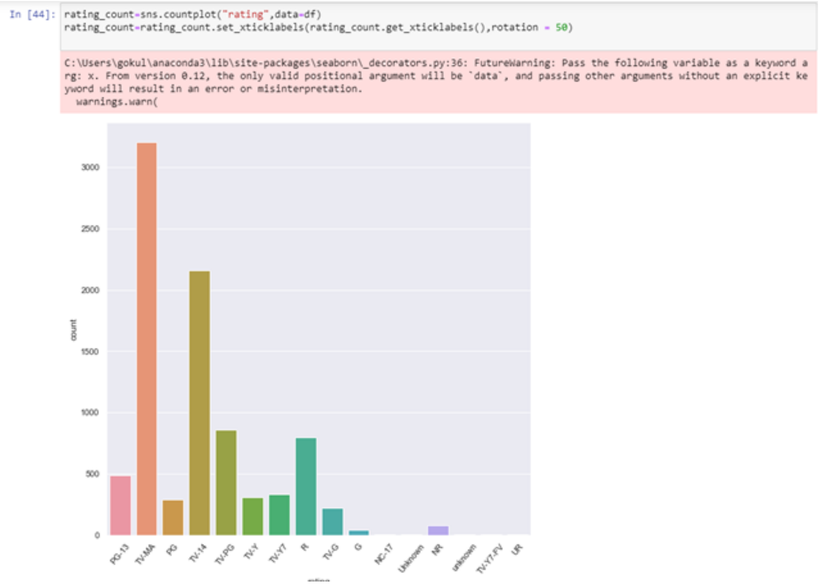
```
In [39]: country_count = sns.countplot(data = df_country , x = "country",order=df_country.country.value_counts().iloc[:10].index)
         country_count=country_count.set_xticklabels(country_count.get_xticklabels(),rotation = 90)
```

The country for which the most content is added was identified. The table shows the content added for each country. The USA has the most content released for it followed by India and UK. This might be because Netflix was stated in the US and most of the content will be from Hollywood movies. Apart from the US rest of the countries are showing no large variations.



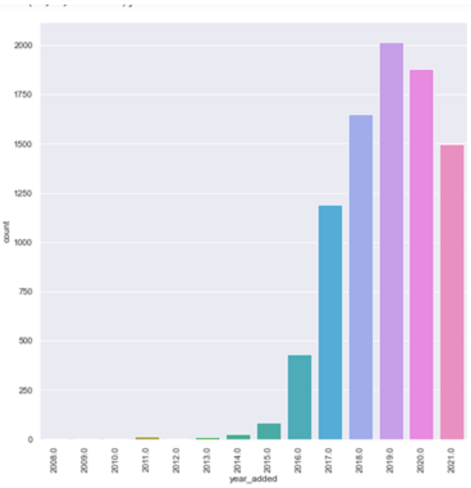
```
In [43]: top_category=sns.barplot(data=df_top_category_new,x="listed_in",y="title")
         top_category=top_category.set_xticklabels(top_category.get_xticklabels(),rotation = 90)
```

Most watched genres were identified. Here we can see dramas being watched followed by comedies, action & adventure, and documentaries. Dramas are watched far more than any other genre on Netflix.

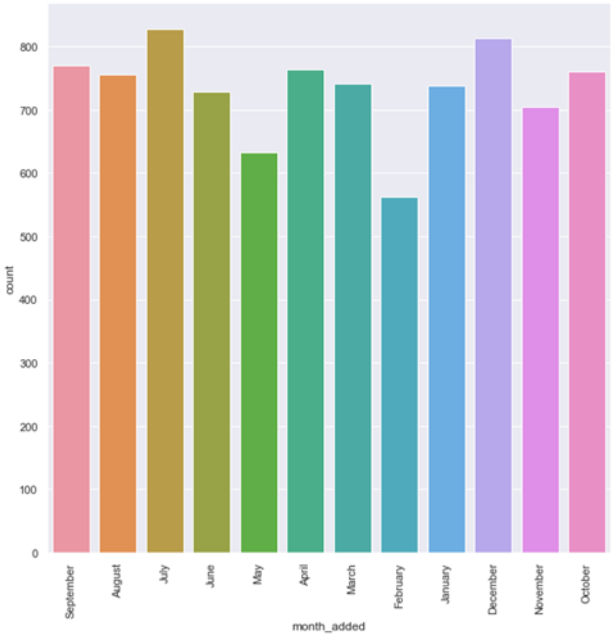


```
In [44]: rating_count=sns.countplot("rating",data=df)
         rating_count=rating_count.set_xticklabels(rating_count.get_xticklabels(),rotation = 50)

C:\Users\gokul\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword a
rg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit ke
yword will result in an error or misinterpretation.
  warnings.warn(
```

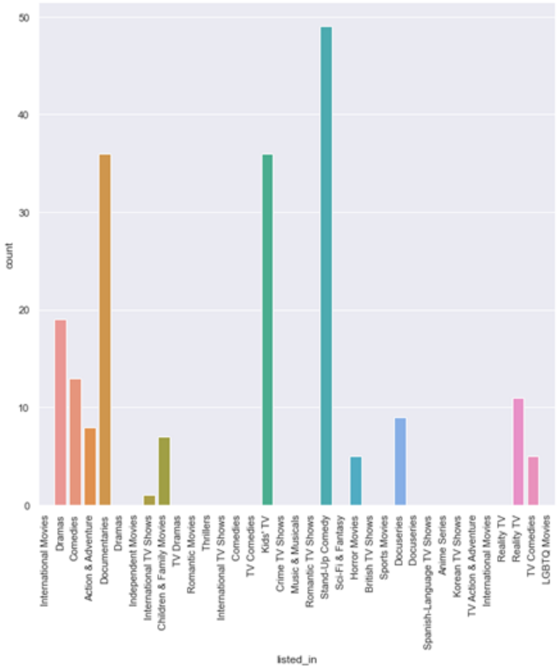
Most rated categories of content were identified. TV-14 and TV-MA have the highest number of content in Netflix. These are more Mature audiences and audiences requiring parental guidance & above 14 year of age.



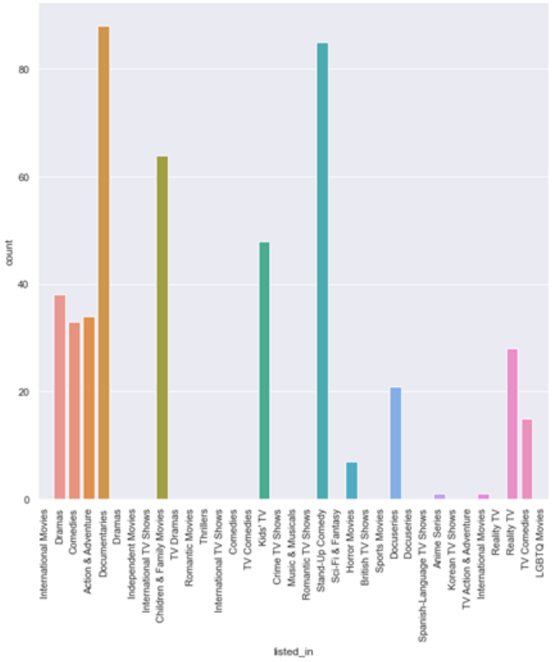
The year with the most added content was identified. It shows the number of content added in each year. Till 2019 the content added to Netflix was going up exponentially but in 2020 there was a decrease in the number of content added to Netflix compared to 2020 and further reduced to 2021.



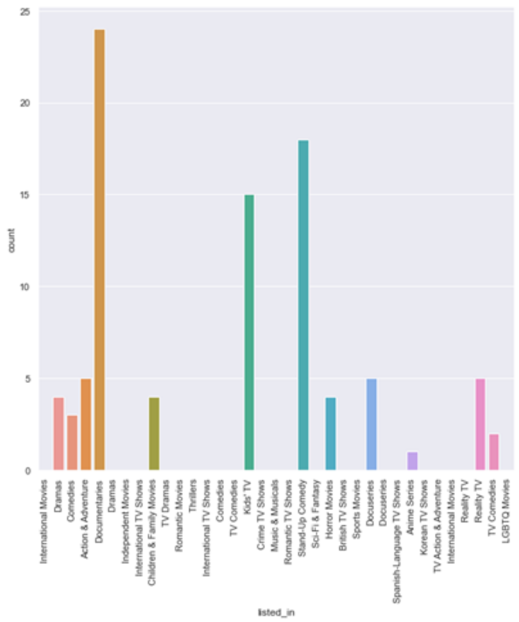
Which month has seen the most content added was identified. The month vs addition of content. Most contents are added in the month of Dec and Jul. And least content was added in the month of Feb and May.



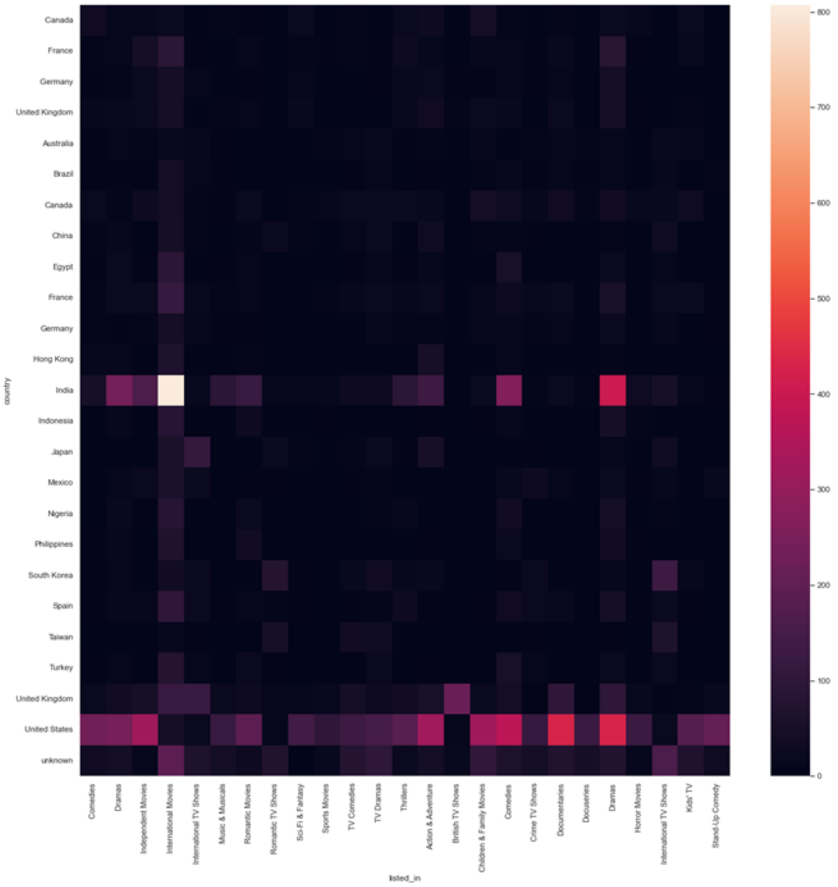
The most watched category in India was identified. The plot shows no. of content vs each Genre. Most uploaded content in India is Stand-up-comedy.



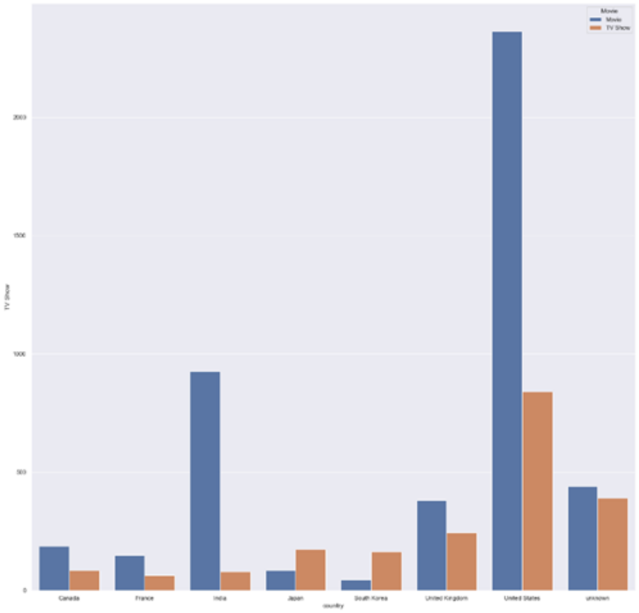
The most watched category in the USA was identified. The plot shows no. of content vs each Genre. Most uploaded content in India is documentaries followed by Stand-up-comedy.



The most watched category in the UK was identified. The plot shows no. of content vs each Genre. Most uploaded content in India is documentaries followed by Stand-up-comedy.

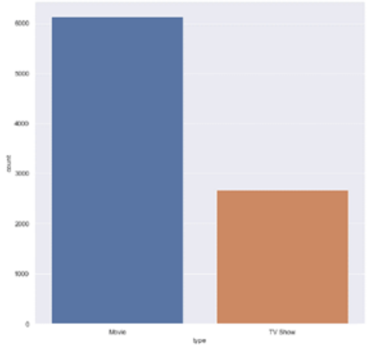


Heatmap was plotted to identify the popular genre in the top most released countries. The heat map can be used to identify which is the most popular content in each country. Light color shows more popular genres and darker the plot shows least watched.



Movies v/s TV show release in each country(counties having at least 200 content added) was identified. The plot shows counties v/s movies/TV shows released. Movies are generally more favored than TV shows except in Japan and South Korea.

5.5. Business Insight



More movies are being added than TV-Shows if you check the overall picture. Movies are generally more favored than TV shows except in Japan and South Korea.

The USA has the most content released for it followed by India and UK. This might be because Netflix was stated in the US and most of the content will be from Hollywood movies. Apart from the US rest of the countries are showing no large variations.

Dramas are watched far more than any other genre on Netflix.

Actors are not producing a large impact on the success of content as there is no actor who has acted in more the 32-content compared to the dataset size of more than 8000.

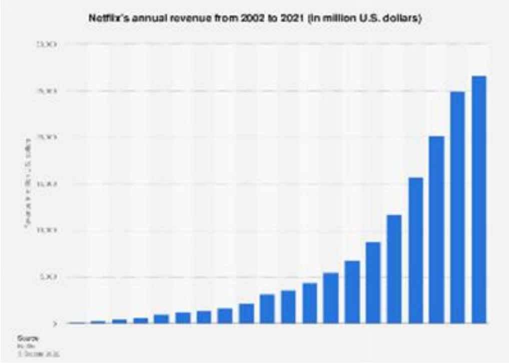
These are more Mature audiences and audiences requiring parental guidance & above 14 years of age.

Even though India has large population, penetration is comparatively less. This can be seen from the number of content release in US v/s India.

Till 2019 the content added to Netflix was going up exponentially by 2020 there was a decrease in the number of contents added to Netflix compared to 2020 and further reduced to 2021.

Most contents are added in the month of Dec and Jul. And least content was added in the month of Feb and May.

More content is being watched by people who are more than 14 years of age.



Source: https://www.statista.com/statistics/272545/annual-revenue-of-netflix/

More content may not produce more growth. Even the revenue was continuously growing. The number of content added reduced in 2020 and 2021.

5. 6. Recommendations

a. More quality content should be produced using the best directors such as Rajiv Chilaka, Jan Suter, RaÃºl Campos, etc.

b. More content should be released in the holiday season.

c. Movies should be more preferred than TV-Shows(Expect in Japan and South Korea)

d. Country wise target should be made while making content by referring to the heatmap given above. As a whole, more drama content should be produced

e. Releasing of new content should be kept on the weekend for more engagement.

f. New actors can be tried out.

g. Indiaâ€™s audience should be more targeted.

h. More content should be made for adolescents and adults.