



## 1. Defining Problem statement and Analysing basic metrics

### PROBLEM STATEMENT-

The Netflix dataset poses challenges related to understanding and leveraging the vast amount of data available from Netflix's streaming platform. The dataset comprises various types of information, such as Directors, Cast, Ratings, Genre for movies and TV shows. The goal is to extract meaningful insights for analyzing 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.

### Importing Libraries

```
In [1]: import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
import warnings  
warnings.filterwarnings('ignore')
```

### Loading Dataset

```
In [2]: df = pd.read_csv(r"C:\Users\SYEDA TAYABA\OneDrive\Desktop\New folder\Netflix.csv")
```

		show_id	type	title	director	cast	country	date_added	release_year	rating
<b>0</b>		s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13
<b>1</b>		s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
<b>2</b>		s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA
<b>3</b>		s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA
<b>4</b>		s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA
<b>...</b>										
<b>8802</b>		s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007	R
<b>8803</b>		s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV-Y7
<b>8804</b>		s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	R

	show_id	type	title	director	cast	country	date_added	release_year	rating
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006	PG
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chan...	India	March 2, 2019	2015	TV-14

## 2. Observations on shape of Data, Data Types of all attributes, Statistical Summary, Missing value Detection.

### shape of data

In [3]: `df.shape`

Out[3]: `(8807, 12)`

### Statistical Summary of Data

In [4]: `df.describe()`

Out[4]: `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 [5]: `df.describe(include='object').T`

Out[5]:

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

## Data types of all attributes

In [6]:

`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
```

## Missing Value Detection

In [7]:

`df.isnull().sum().sort_values(ascending = False)`

```
Out[7]: director      2634
         country       831
         cast          825
         date_added    10
         rating         4
         duration        3
         show_id         0
         type            0
         title            0
         release_year    0
         listed_in        0
         description       0
         dtype: int64
```

```
In [8]: round((df.isnull().sum()/df.shape[0]*100),2).sort_values(ascending = False)
```

```
Out[8]: director      29.91
         country       9.44
         cast          9.37
         date_added    0.11
         rating         0.05
         duration        0.03
         show_id         0.00
         type            0.00
         title            0.00
         release_year    0.00
         listed_in        0.00
         description       0.00
         dtype: float64
```

### 3. Non-Graphical Analysis : Value counts and unique attributes

```
In [9]: # Value counts
         df['director'].value_counts()
```

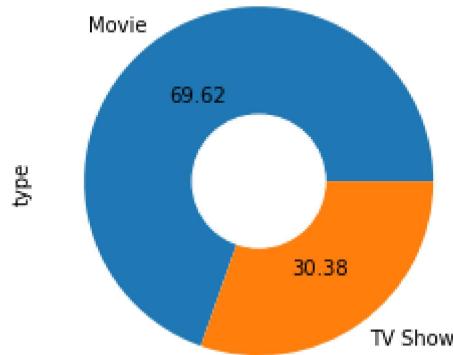
```
Out[9]: Rajiv Chilaka                  19
         Raúl Campos, Jan Suter           18
         Marcus Raboy                   16
         Suhas Kadav                   16
         Jay Karas                      14
         ..
         Raymie Muzquiz, Stu Livingston     1
         Joe Menendez                   1
         Eric Bross                      1
         Will Eisenberg                  1
         Mozez Singh                     1
         Name: director, Length: 4528, dtype: int64
```

### Movies Vs TVShows

```
In [10]: df_movie = df[df["type"] == "Movie"].reset_index()
         df_tv_show = df[df["type"] == "TV Show"].reset_index()
```

```
In [11]: plt.title("Percentage of Movies vs TV Shows available on Netflix as of 2021")
df['type'].value_counts().plot(kind = "pie", autopct=".2f")
circle = plt.Circle((0, 0), 0.38, color='white')
plt.gcf().gca().add_artist(circle)
plt.show()
```

Percentage of Movies vs TV Shows available on Netflix as of 2021



- Nearly 2/3rd of the content on netflix are movies and remaining 1/3rd of them are TV Show.

## Top 10 Directors

```
In [12]: # Unique Attributes

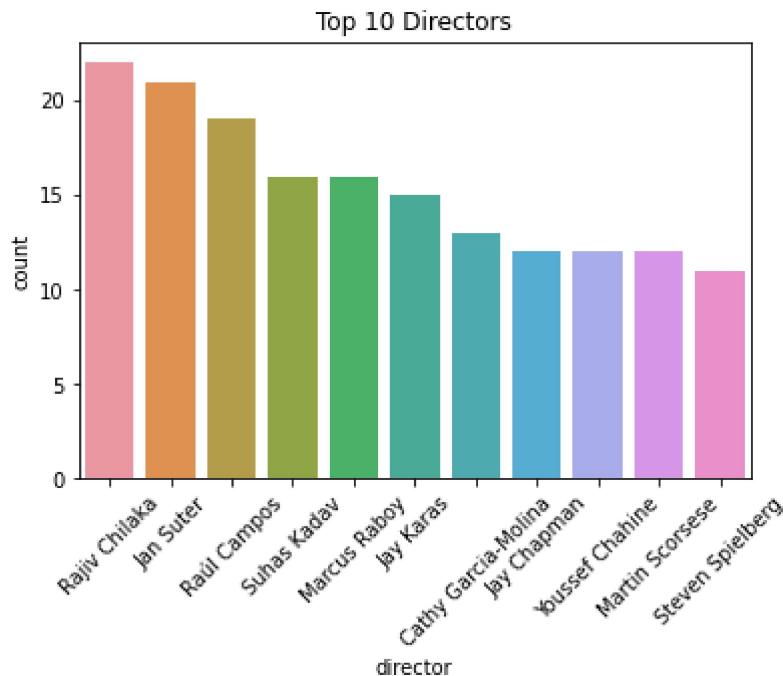
#unnesting the 'director' column to get individual directors

df_dir = df[["director"]]
df_dir["director"] = df_dir["director"].str.split(", ")
df_dir = df_dir.explode("director")
```

```
In [13]: df_dir["director"].value_counts().head(11)
```

```
Out[13]: Rajiv Chilaka      22
Jan Suter          21
Raúl Campos        19
Suhas Kadav        16
Marcus Raboy        16
Jay Karas           15
Cathy Garcia-Molina  13
Jay Chapman          12
Youssef Chahine       12
Martin Scorsese       12
Steven Spielberg      11
Name: director, dtype: int64
```

```
In [14]: sns.countplot(x=df_dir["director"], order = pd.value_counts(df_dir["director"]).head(11))
plt.xticks(rotation=45)
plt.title("Top 10 Directors")
plt.show()
```



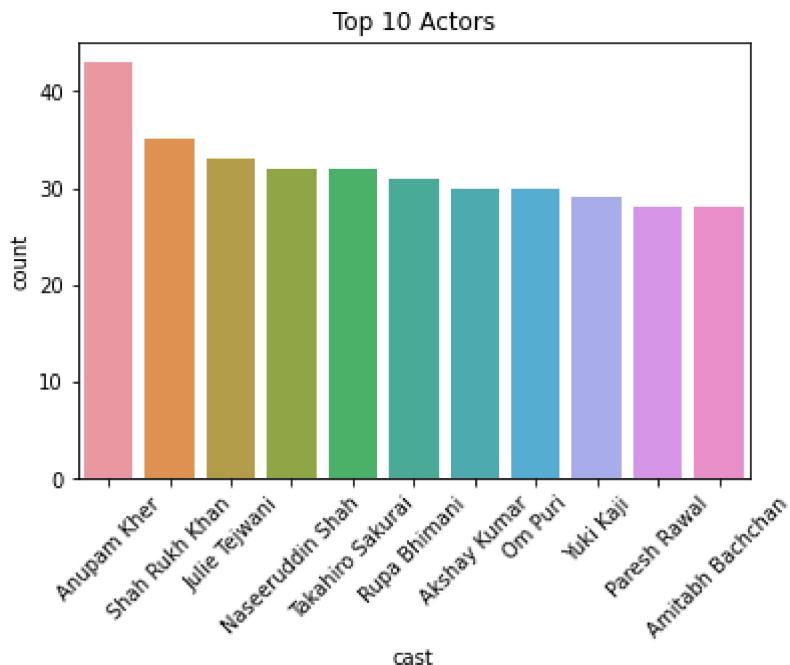
### Top 10 Actors

```
In [15]: df_actor = df[["title", "cast"]]
df_actor["cast"] = df_actor["cast"].str.split(", ")
df_actor = df_actor.explode("cast")
```

```
In [16]: df_actor["cast"].value_counts().head(10)
```

```
Out[16]: Anupam Kher      43
Shah Rukh Khan       35
Julie Tejwani        33
Naseeruddin Shah     32
Takahiro Sakurai     32
Rupa Bhimani         31
Akshay Kumar          30
Om Puri               30
Yuki Kaji              29
Paresh Rawal          28
Name: cast, dtype: int64
```

```
In [17]: sns.countplot(x=df_actor["cast"], order = pd.value_counts(df_actor["cast"]).head(11)
plt.xticks(rotation=45)
plt.title("Top 10 Actors")
plt.show()
```



- Top 5 artist's are from India.

### Top 10 countries

```
In [18]: df_country = df[["title", "country"]]
df_country["country"] = df_country["country"].str.split(", ")
df_country = df_country.explode("country")
```

```
In [19]: df_country["country"].value_counts().head(10)
```

```
Out[19]: United States      3689
          India            1046
          United Kingdom    804
          Canada           445
          France           393
          Japan             318
          Spain             232
          South Korea       231
          Germany           226
          Mexico            169
Name: country, dtype: int64
```

```
In [20]: df = pd.read_csv(r"C:\Users\SYEDA TAYABA\OneDrive\Desktop\New folder\Netflix.csv")
df1 = df['title'].value_counts().sum()
```

```
In [21]: df1
```

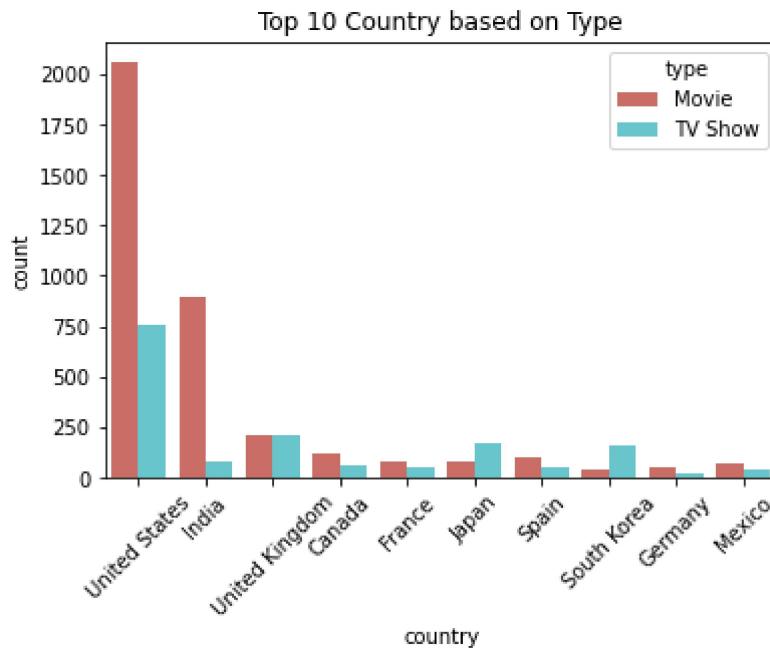
```
Out[21]: 8807
```

```
In [22]: df2 = df_country["country"].value_counts().head(5).sum()
```

```
In [23]: percentage = df2/df1*100
round(percentage, 2)
```

Out[23]: 72.41

```
In [24]: sns.countplot(x=df["country"], palette = "hls" ,hue = df['type'], order = pd.value_c
plt.xticks(rotation=45)
plt.title("Top 10 Country based on Type")
plt.show()
```



- We can see that US, India, United Kingdom, Canada and France contribute 72.4% of the Top10 countries.

### Top 10 Genre

```
In [25]: df_listed_in = df[["title","listed_in"]]
df_listed_in["listed_in"] = df_listed_in["listed_in"].str.split(", ")
df_listed_in = df_listed_in.explode("listed_in")
```

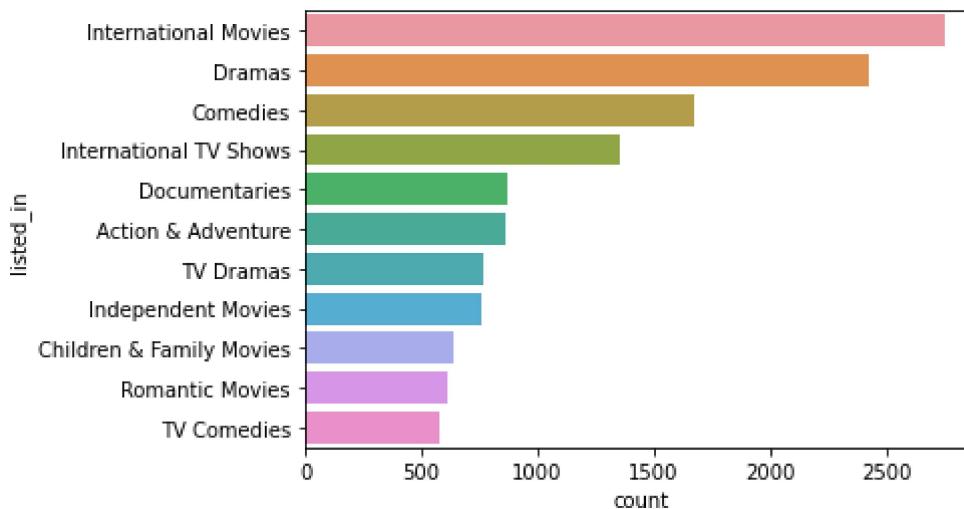
```
In [26]: df_listed_in["listed_in"].value_counts().head(10)
```

Out[26]:

International Movies	2752
Dramas	2427
Comedies	1674
International TV Shows	1351
Documentaries	869
Action & Adventure	859
TV Dramas	763
Independent Movies	756
Children & Family Movies	641
Romantic Movies	616

Name: listed\_in, dtype: int64

```
In [27]: sns.countplot(y=df_listed_in["listed_in"], order = pd.value_counts(df_listed_in["li
plt.show()
```



## Movies and TV Shows Released Year-wise

```
In [28]: df["release_year"].value_counts().head(30)
```

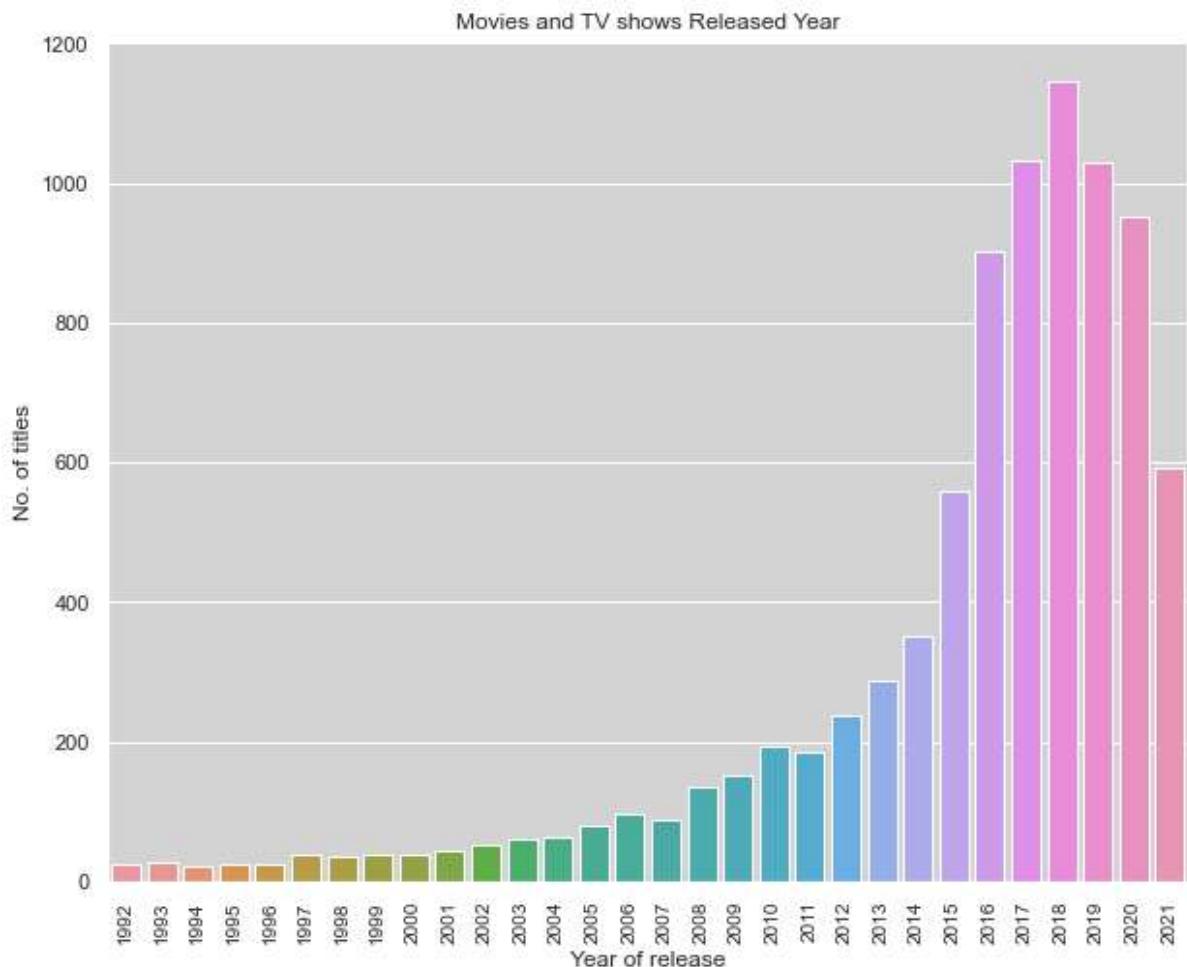
```
Out[28]:
```

2018	1147
2017	1032
2019	1030
2020	953
2016	902
2021	592
2015	560
2014	352
2013	288
2012	237
2010	194
2011	185
2009	152
2008	136
2006	96
2007	88
2005	80
2004	64
2003	61
2002	51
2001	45
1999	39
1997	38
2000	37
1998	36
1993	28
1995	25
1996	24
1992	23
1994	22

Name: release\_year, dtype: int64

```
In [29]: year_wise = pd.value_counts(df["release_year"]).reset_index().head(30)
year_wise.rename(columns={"index":"Year of release", "release_year":"No. of titles"})
sns.set(rc={'axes.facecolor':'lightgrey'})
```

```
In [30]: plt.figure(figsize=(10,8))
sns.barplot(data=year_wise,x="Year of release", y="No. of titles")
plt.xticks(rotation=90 ,fontsize=10)
plt.title("Movies and TV shows Released Year")
plt.show()
```



- About 1200 new movies were released in both 2018 and 2019.
- The growth in content started from 2014.

## Movies and TV Shows Released Month-wise

```
In [31]: Month = df[["date_added"]].replace(np.nan,'No_date')
Month["Release_month"] = Month['date_added'].apply(lambda x: x.lstrip().split(" ")[
```

```
In [32]: Month["Release_month"].value_counts().head(12)
```

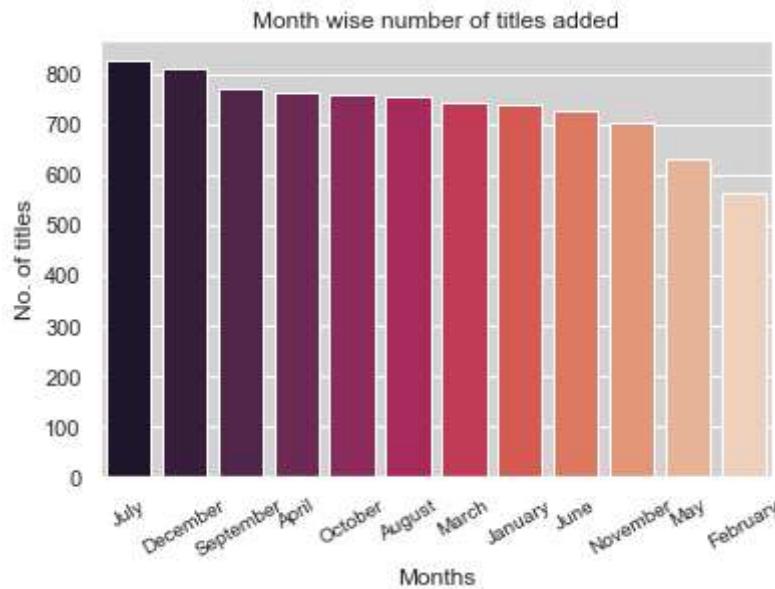
```
Out[32]:
```

July	827
December	813
September	770
April	764
October	760
August	755
March	742
January	738
June	728
November	705
May	632
February	563

Name: Release\_month, dtype: int64

```
In [33]:
```

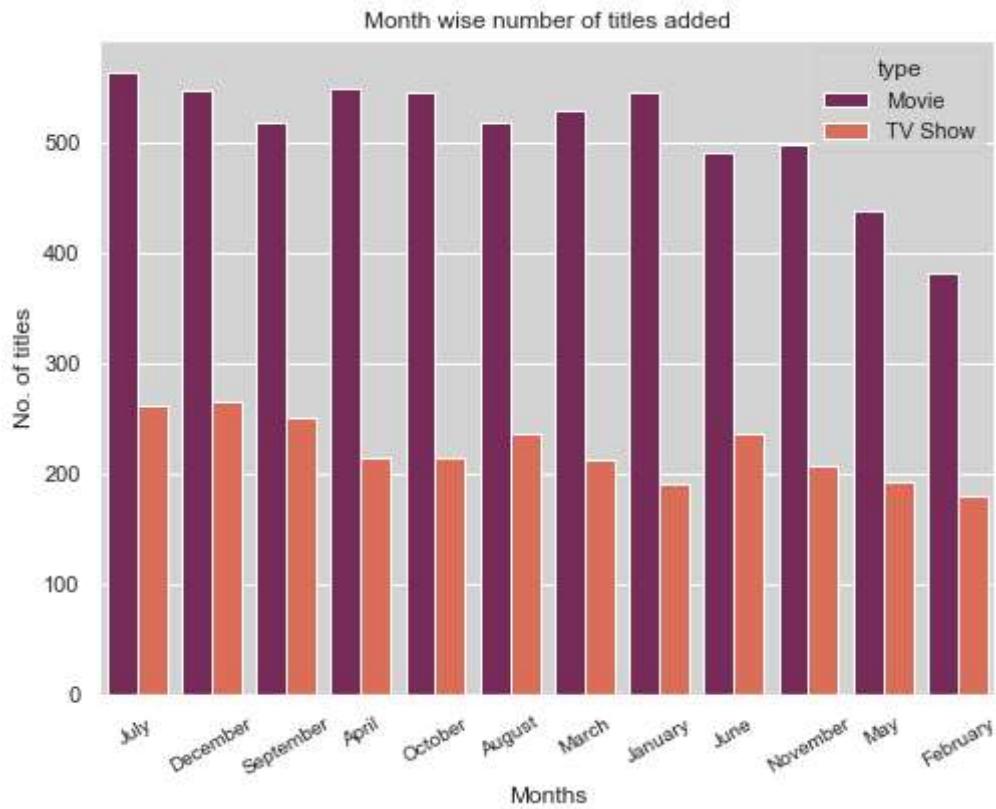
```
sns.set(rc={'axes.facecolor':'lightgrey'})
m = sns.countplot(palette = "rocket", x = Month["Release_month"], order = pd.value
plt.title("Month wise number of titles added")
m.set_ylabel('No. of titles')
m.set_xlabel('Months')
plt.xticks(rotation = 30, fontsize = 10)
plt.show()
```



- The growth in contents are higher in the first three months and the last three months of the year.
- Least number of contents are added in the month of February.

```
In [34]:
```

```
sns.set(rc={'axes.facecolor':'lightgrey'})
plt.figure(figsize=(8,6))
m = sns.countplot(palette = "rocket", x = Month["Release_month"], hue = df['type'],
plt.title("Month wise number of titles added")
m.set_ylabel('No. of titles')
m.set_xlabel('Months')
plt.xticks(rotation = 30, fontsize = 10)
plt.show()
```



- Most movies added in the months of July, December, April, October and January.
- Least movies added in the months of June, May and February.
- Most TV Shows added in the months of July and December.
- Least TV Shows added in the months of January and February.

## 4. Visual Analysis - Univariate, Bivariate

### Correlation of Month and Year of content added

```
In [35]: df = pd.read_csv(r"C:\Users\SYEDA TAYABA\OneDrive\Desktop\New folder\Netflix.csv")
Date = df[['date_added']].dropna()
Date['date'] = Date['date_added'].apply(lambda x : x.lstrip().split(' ')[1].replace('-', ''))
Date['months'] = Date['date_added'].apply(lambda x : x.lstrip().split(' ')[0])
Date['years'] = Date['date_added'].apply(lambda x : x.split(', ')[-1])
Order_Of_Month = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December']
df8 = Date.groupby('years')[['months']].value_counts().unstack().fillna(0)[Order_Of_Month]
```

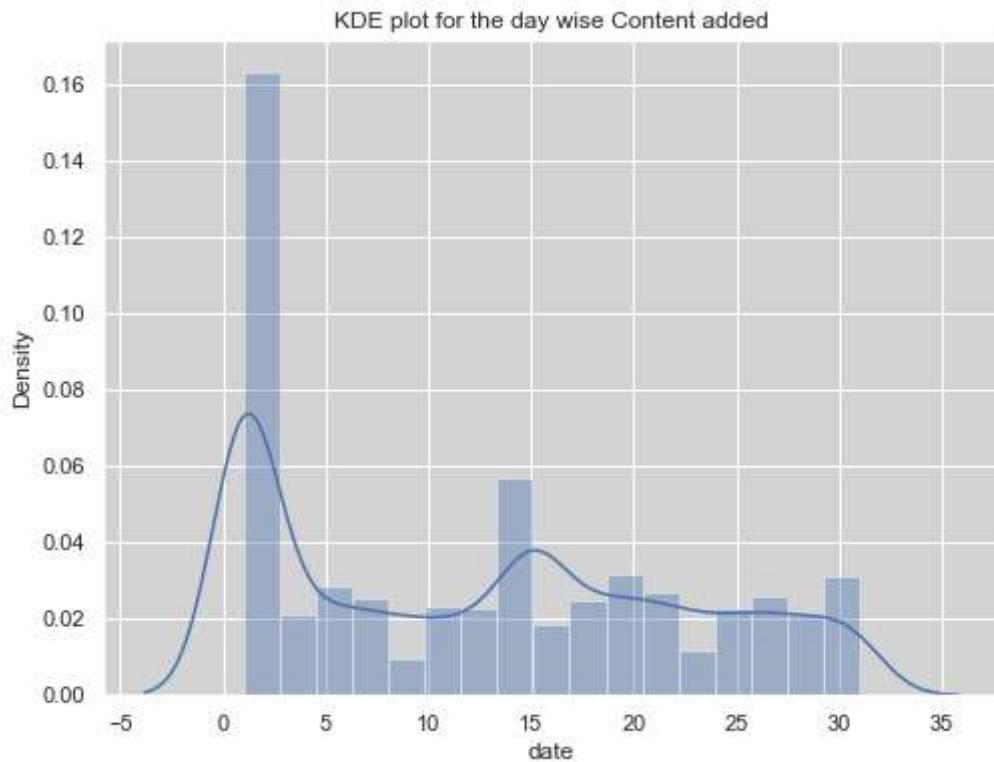
```
In [36]: perc_1 = (Date['date'].value_counts()[1]/Date['date'].value_counts().sum())*100
print((round(perc_1,2)), "%")
```

7.81 %

```
In [37]: Date['date'].value_counts().sort_values(ascending = True)
```

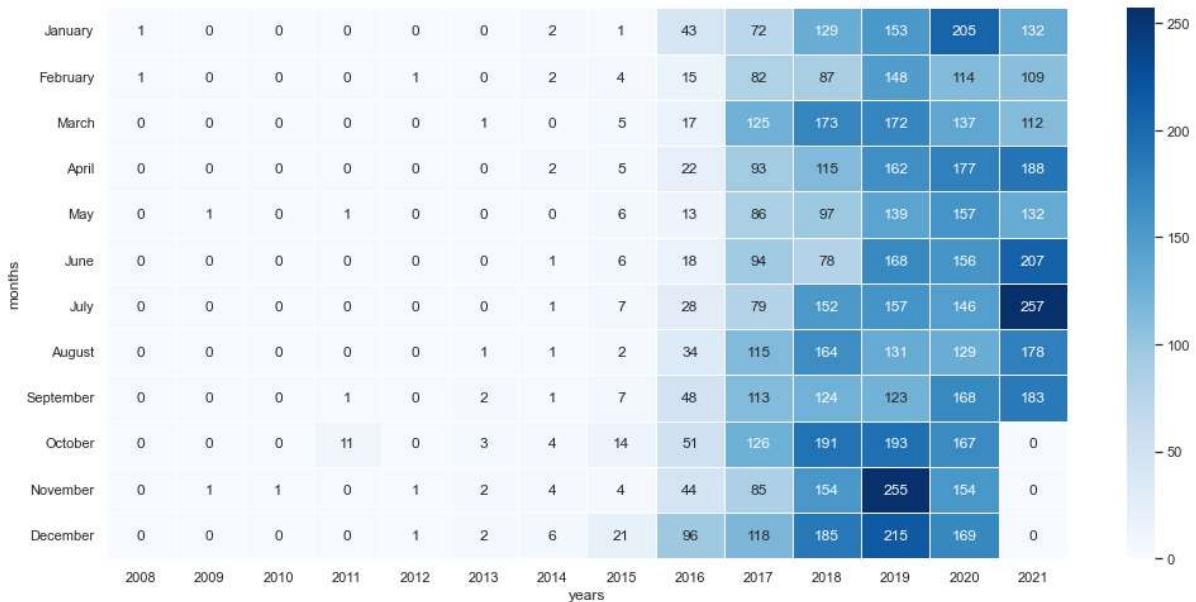
```
Out[37]: 29      141
         9      147
        11     149
        3     151
       24     159
       13     175
        4     175
       17     180
       12     181
       23     184
       28     190
       21     193
        7     194
       27     195
       25     197
       14     198
        8     201
       26     206
       18     207
        6     210
       30     210
       10     214
       22     230
        5     231
       19     243
       20     249
       31     274
       16     289
        2     325
       15     687
        1    2212
Name: date, dtype: int64
```

```
In [38]: plt.figure(figsize=(8,6))
sns.distplot(Date['date'])
plt.title('KDE plot for the day wise Content added')
plt.show()
```



- Analysed that content is added to Netflix on 1st date of the month.

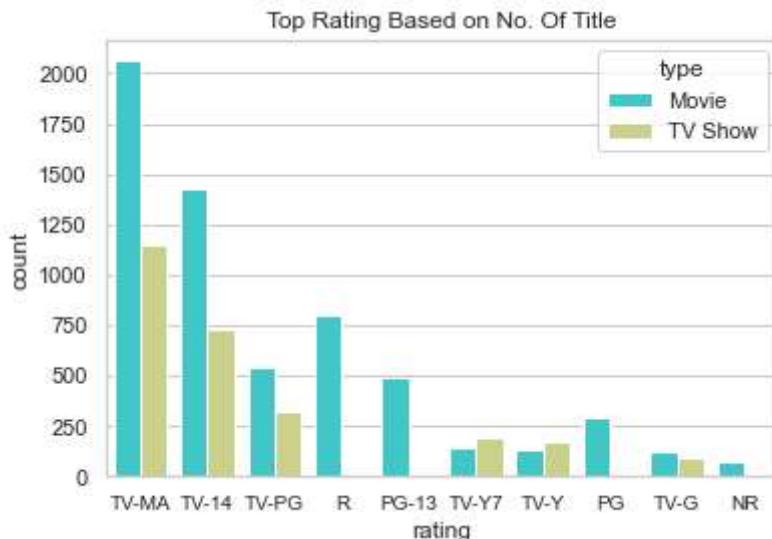
```
In [62]: plt.figure(figsize=(17,8))
sns.heatmap(data=df8, annot=True, linewidth=0.5, fmt=".0f", cmap="Blues")
plt.show()
```



- Most amount of content was added in july 2021 (257) followed by November 2019(255).
- There is a significant growth in content added on Netflix from 2016.

## Rating Based on Content

```
In [40]: sns.set_theme(style="whitegrid")
sns.countplot(palette="rainbow",x=df["rating"], hue = df['type'], order=pd.value_c
plt.xticks( fontsize=10)
plt.title("Top Rating Based on No. Of Title")
plt.show()
```



- TV-Y :Designed to be appropriate for all children
- TV-Y7 :Suitable for ages 7 and up
- G :Suitable for General Audiences
- TV-G :Suitable for General Audiences
- PG: Parental Guidance suggested
- TV-PG: Parental Guidance suggested
- PG-13:Parents strongly cautioned. May be inappropriate for ages under 13.
- TV-14 : Parents strongly cautioned. May not be suitable for ages under 14.
- R : Restricted. May be inappropriate for ages under 17.
- TV-MA : For Mature Audiences
- NC-17 : Inappropriate for ages 17 and under
  
- TV-MA rating has the highest number of Movies and TV Shows available followed by TV-14 rating.
- Nearly 2/3rd of the content on netflix are movies and remaining 1/3rd of them are TV Show.

## Movies

```
In [41]: movies_df = df.loc[(df['type']=="Movie")]
movies_df.head(5)
```

Out[41]:

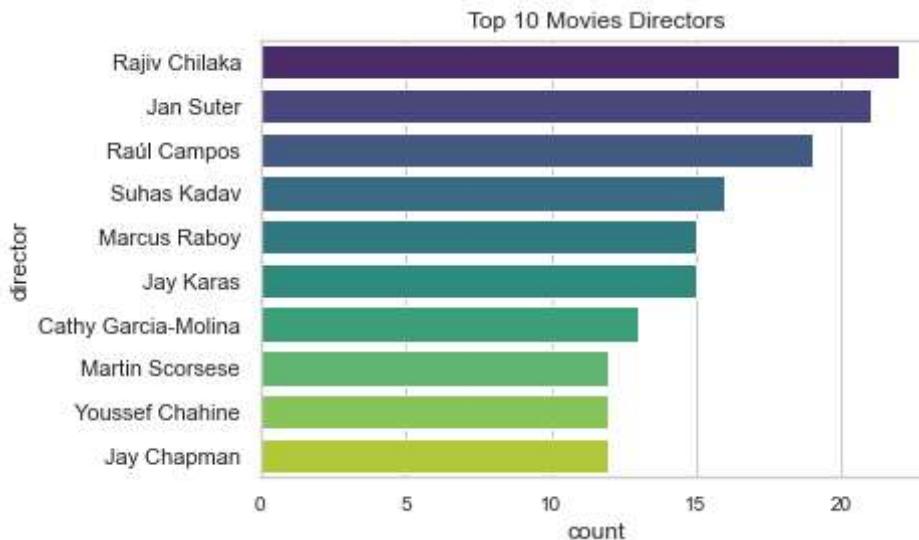
	show_id	type	title	director	cast	country	date_added	release_year	ratir
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-
6	s7	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	NaN	September 24, 2021	2021	F
7	s8	Movie	Sankofa	Haile Gerima	Kofi Ghanaba, Oyafunmike Ogunlano, Alexandra D...	United States, Ghana, Burkina Faso, United Kin...	September 24, 2021	1993	T
9	s10	Movie	The Starling	Theodore Melfi	Melissa McCarthy, Chris O'Dowd, Kevin Kline, T...	United States	September 24, 2021	2021	PG-
12	s13	Movie	Je Suis Karl	Christian Schwochow	Luna Wedler, Jannis Niewöhner, Milan Peschel, ...	Germany, Czech Republic	September 23, 2021	2021	T

In [42]:

```
df_movie_dir=movies_df[["director"]]
df_movie_dir["director"]=df_movie_dir["director"].str.split(", ")
df_movie_dir=df_movie_dir.explode("director")
```

In [43]:

```
sns.set_theme(style="whitegrid")
sns.countplot(palette="viridis",y=df_movie_dir["director"],order=pd.value_counts(df
plt.xticks(fontsize=10)
plt.title("Top 10 Movies Directors")
plt.show()
```



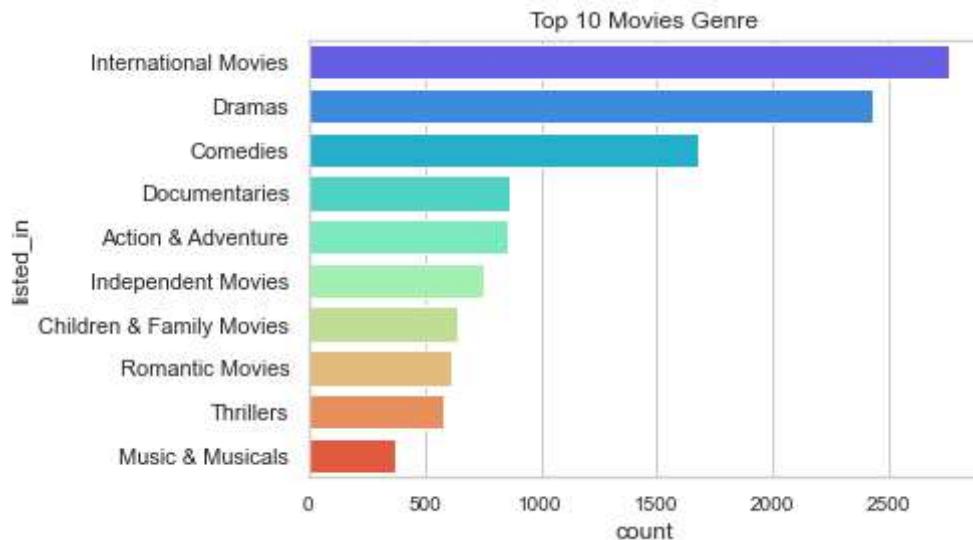
```
In [44]: df_movie_cast=movies_df[["cast"]]
df_movie_cast["cast"]=df_movie_cast["cast"].str.split(", ")
df_movie_cast= df_movie_cast.explode("cast")
```

```
In [45]: sns.set_theme(style="whitegrid")
sns.countplot(palette="rainbow",y=df_movie_cast["cast"],order=pd.value_counts(df_mc
plt.xticks( fontsize=10)
plt.title("Top 10 Movies Actors")
plt.show()
```



```
In [46]: df_movie_genre=movies_df[["listed_in"]]
df_movie_genre["listed_in"]=df_movie_genre["listed_in"].str.split(", ")
df_movie_genre= df_movie_genre.explode("listed_in")
```

```
In [47]: sns.set_theme(style="whitegrid")
sns.countplot(palette="rainbow",y=df_movie_genre["listed_in"],order=pd.value_counts
plt.xticks( fontsize=10)
plt.title("Top 10 Movies Genre")
plt.show()
```



```
In [48]: movies_df['duration'] = movies_df['duration'].astype(str).str.replace(' min', '')
movies_df.head(5)
```

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-
6	s7	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	NaN	September 24, 2021	2021	F
7	s8	Movie	Sankofa	Haile Gerima	Kofi Ghanaba, Oyafunmike Ogunlano, Alexandra D...	United States, Ghana, Burkina Faso, United Kin...	September 24, 2021	1993	T
9	s10	Movie	The Starling	Theodore Melfi	Melissa McCarthy, Chris O'Dowd, Kevin Kline, T...	United States	September 24, 2021	2021	PG-
12	s13	Movie	Je Suis Karl	Christian Schwochow	Luna Wedler, Jannis Niewöhner, Milan Peschel, ...	Germany, Czech Republic	September 23, 2021	2021	T

```
In [49]: movies_df['duration']
```

```
Out[49]: 0      90
          6      91
          7     125
          9     104
         12     127
          ...
        8801     96
        8802    158
        8804     88
        8805     88
        8806    111
Name: duration, Length: 6131, dtype: object
```

```
In [50]: movies_df["duration"].dropna()
```

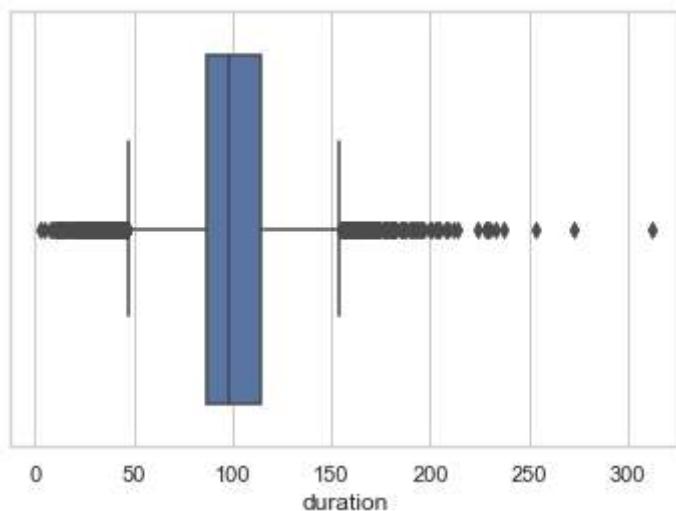
```
Out[50]: 0      90
          6      91
          7     125
          9     104
         12     127
          ...
        8801     96
        8802    158
        8804     88
        8805     88
        8806    111
Name: duration, Length: 6131, dtype: object
```

```
In [51]: movies_df["duration"] = movies_df["duration"].astype(float)
```

```
In [63]: round(movies_df["duration"].describe(), 2)
```

```
Out[63]: count    6128.00
          mean     99.58
          std      28.29
          min      3.00
          25%     87.00
          50%     98.00
          75%    114.00
          max    312.00
Name: duration, dtype: float64
```

```
In [53]: sns.boxplot(movies_df["duration"])
plt.show()
```

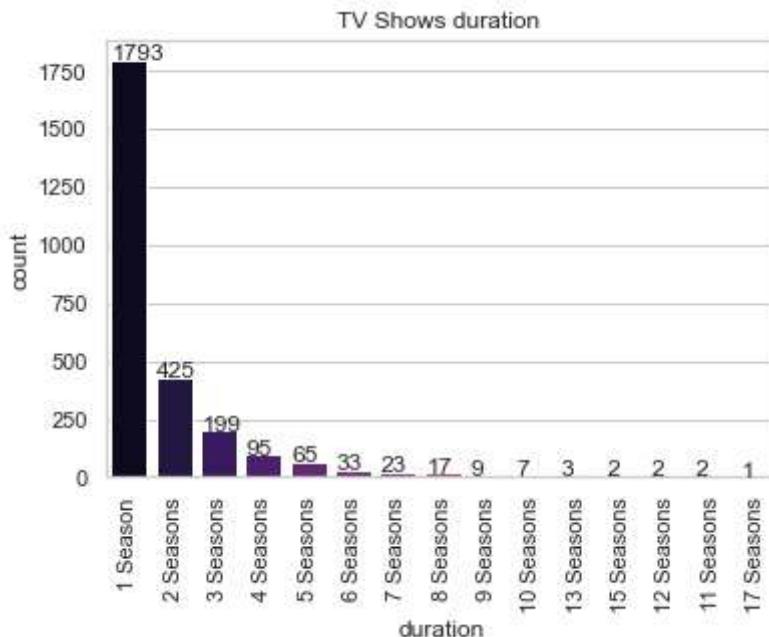


- we analyzed that there are few outliers in Duration of Movies
- The average duration of Movies is between 80 to 120 minutes
- The minimum duration of Movies is 50 minutes

## TV Shows

```
In [54]: tv_show_df=df.loc[(df["type"]=="TV Show")]
tv_show_df.reset_index()
tv_show_df.columns.name=None
```

```
In [55]: sns.set_theme(style="whitegrid")
ax=sns.countplot(palette="magma",x=tv_show_df['duration'],order=pd.value_counts(tv_
plt.xticks(rotation = 90)
for p in ax.patches:
    ax.annotate(str(p.get_height()), (p.get_x() * 1.005, (p.get_height() * 1.005))
plt.title("TV Shows duration ")
plt.show()
```



- When we analyze the Tv show data we came to know 1793 shows have only 1 Season.
- whereas we there only few shows which have more than 10 seasons.
- there is one show which has 17 Seasons.

## 5. MISSING VALUE TREATMENT (Dropping Null Values)

```
In [56]: df.dropna(subset=["duration", "rating", "date_added"], inplace = True)
```

```
In [57]: round((df.isnull().sum()/df.shape[0]*100),2).sort_values(ascending = False)
```

```
Out[57]: director      29.82
country       9.43
cast          9.39
show_id       0.00
type          0.00
title          0.00
date_added    0.00
release_year   0.00
rating         0.00
duration       0.00
listed_in      0.00
description     0.00
dtype: float64
```

```
In [58]: # Replacing NaN values in cast column with "No Cast"
```

```
df["cast"].replace(np.NaN, "No Cast", inplace = True)
```

```
In [59]: # Replacing NaN values in country column with "Unknown"
```

```
df["country"].replace(np.NaN, "Unknown", inplace = True)
```

```
In [60]: round((df.isna().sum()/df.shape[0]*100),2).sort_values(ascending = False)
```

```
Out[60]: director      29.82
          show_id       0.00
          type          0.00
          title         0.00
          cast          0.00
          country        0.00
          date_added    0.00
          release_year   0.00
          rating         0.00
          duration        0.00
          listed_in      0.00
          description     0.00
          dtype: float64
```

## 6. Insights based on Non-Graphical and Visual Analysis

- Nearly 2/3rd of the content on netflix are movies and remaining 1/3rd of them are TV Show.
- We can see that US, India, United Kingdom, Canada and France contribute 72.4% of the Top10 countries.
- About 1200 new movies were released in both 2018 and 2019.
- The growth in content started from 2014.
- The growth in contents are higher in the first three months and the last three months of the year.
- Least number of contents are added in the month of February.
- Most movies added in the months of July, December, April, october and January.
- Least movies added in the months of June, May and February.
- Most TV Shows added in the months of July and December.
- Least TV Shows added in the months of January and February.
- Analysed that content is added to Netflix on 1st date of the month.
- Most amount of content was added in july 2021 (257) followed by November 2019(255).
- There is a significant growth in content added on Netflix from 2016.
- TV-MA rating has the highest number of Movies and TV Shows available followed by TV-14 rating.
- we analyzed that there are few outliers in Duration of Movies.
- The average duration of Movies is between 80 to 120 minutes.
- The minumum duration of Movies is 50 minutes.
- When we analyze the Tv show data we came to know 1793 shows have only 1 Season.
- where as we there only few shows which have more than 10 seasons.
- there is one show which has 17 Seasons.

## 7. Business Insights

- 72% of the content available on Netflix belongs to the movie category while Tv shows consists of the rest 30%.

- Rajiv Chilaka, Jan Suter and Raul Campos have directed most amount of titles among Movies and TV Shows.
- Anupam Kher has acted in the most number of movies followed by Shah Rukh Khan and Naseeruddin Shah.
- All of the top 10 actors based on number of movies are of India.
- United States is the largest country producing movies on Netflix.
- International Movies is most watched Genre.
- In the year 2018, most movies and Tv shows were added.
- TV-MA rating has the highest number of rating for Movies and TV Shows.

## 8. Recommendations

### 1. Which countries to target?

- United States, India, United Kingdom, Canada and France contribute 72.41% of the total content. So including content produced from these countries is recommended.

### 2. Which Movie/TV Show genre to add?

- International Movies, Dramas, Comedies are the top 3 genres for Movies on Netflix.
- International TV Shows, TV Dramas, TV Comedies are the top 3 genres for TV Shows on Netflix.

### 3. Average duration of the Movie/TV shows?

- Around 87 to 114 minutes for the movies
- 67% of the total number of TV Shows have only 1 Season. Not more than 3 seasons is recommended.

### 4. In which month to add a Movie/TV show?

- For Movies, most months are suitable however, February, May and June should be looked into as they have the least amount of Movies added.
- For TV Shows, most months are suitable however, January, February and May should be looked into as they have the least amount of TV Shows added.

### 5. Which directors can be preferred for Movies/TV Shows?

- Rajiv Chilaka, Jan Suter, Raul Campos, Suhas Kadav and Marcus Raboy are top 5 most recurring directors that can be preferred.

### 6. Which actors can be considered Movies/TV Shows?

- Anupam Kher, Shah Rukh Khan, Naseeruddin Shah, Akshay Kumar and Om Puri are the top 5 most recurring Movie actors.

- Takahiro Sakurai, Yuji Kaji, Junichi Suwabe, Daisuke Ono and Ai Kayano are the top 5 most recurring TV Shows actors.

## 7. When to add new content?

- Usually 1st day of the month is most recurring day when new content is added however, 15th day of the month can also be considered as a second option which is considerably less than 1st day of the month.

## 8. Which rating to consider?

- TV-MA is the most recurring rating for Movies as well as TV-Shows, so it is most recommended.

## Additional point to be considered:

There is a significant drop in the amount of movies added to the Netflix after 2018 and for TV shows there is a drop after 2020, which must be looked into while making future business regarding adding content to Netflix.