Business Case: Netflix - Data Exploration and Visualisation

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df=pd.read_csv("netflix.csv")
df.head()
```

	show_id	type	title	director	cast	country	date_added	release_year
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021
					Sami			

```
len(df)
```

```
8807
```

```
df.shape
```

```
(8807, 12)
```

```
df.columns
```

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

```
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

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

```
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
```

Percentage Of Missing Values

```
round(df.isnull().sum()/df.shape[0]*100,1).sort_values(ascending=False)
```

```
director        29.9
cast             9.4
country          9.4
date_added       0.1
show_id          0.0
type             0.0
title            0.0
release_year     0.0
rating           0.0
duration         0.0
listed_in        0.0
description      0.0
dtype: float64
```

Top 10 Directors

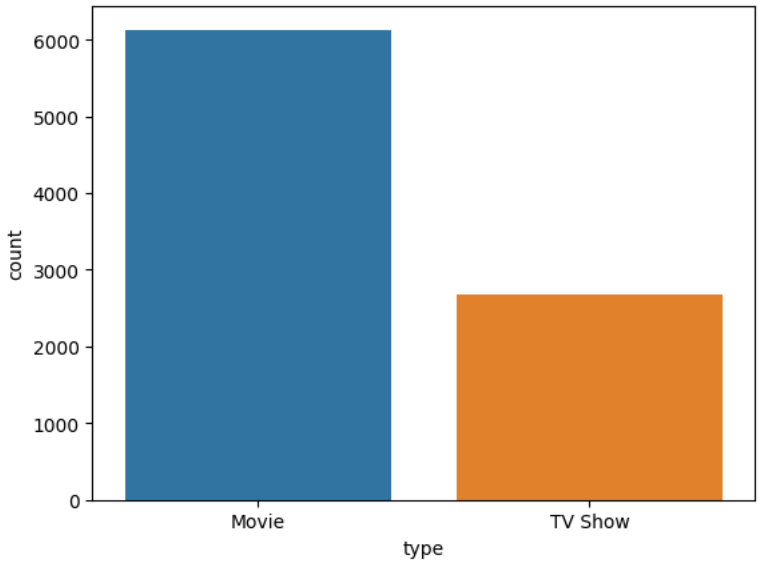
```
df['director'].value_counts().head(10)
```

```
Rajiv Chilaka             19
Raúl Campos, Jan Suter    18
Marcus Raboy              16
Suhas Kadav               16
Jay Karas                 14
Cathy Garcia-Molina       13
Martin Scorsese           12
Youssef Chahine           12
Jay Chapman               12
Steven Spielberg          11
Name: director, dtype: int64
```

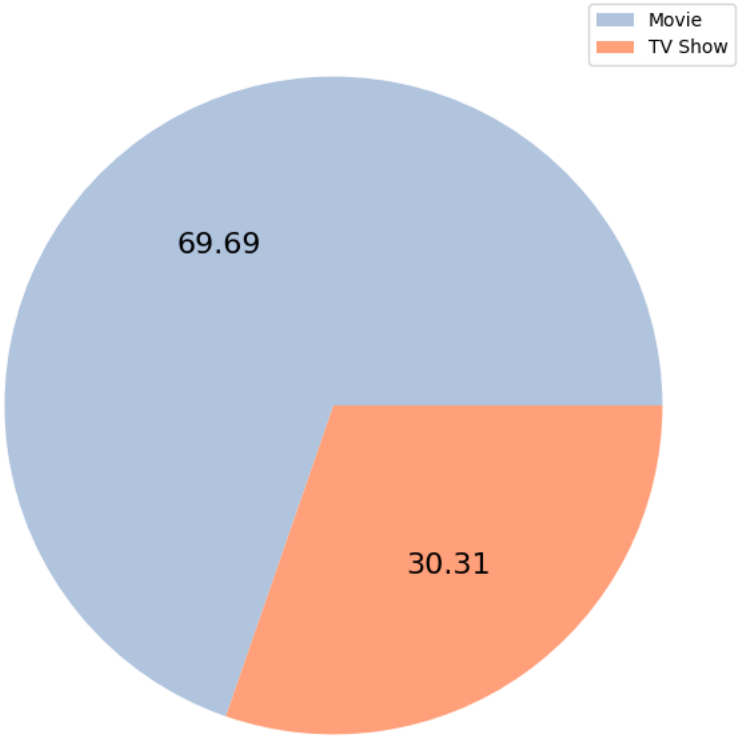
Movies VS TV Shows

```
sns.countplot(x= 'type', data=df)
```

```
<Axes: xlabel='type', ylabel='count'>
```




```
plt.figure(figsize=(10,8))

plt.pie(df.type.value_counts(),
        labels = df.type.value_counts().index,
        labeldistance = None, autopct="%.2f",
        textprops = {'fontsize': 16,},
        colors = ['lightsteelblue','lightsalmon' ] )
plt.legend()
plt.show()
```



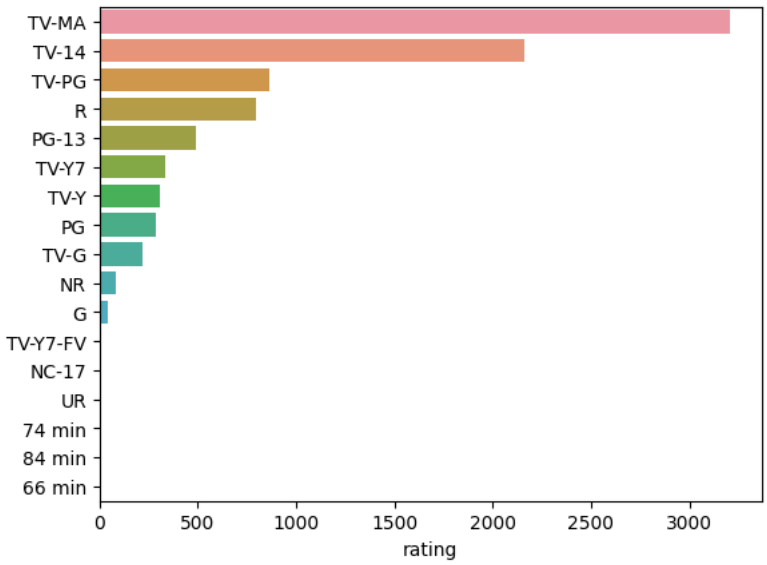

```
df.type.value_counts()
```

```
Movie      6131
TV Show    2676
Name: type, dtype: int64
```

```
df.rating.value_counts()
```

```
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
```

```
sns.barplot(x=df.rating.value_counts(),y=df.rating.value_counts().index,data=df,orient='h')
plt.show()
```




```
df.country.value_counts()
```

```
United States                                2818
India                                         972
United Kingdom                                419
Japan                                         245
South Korea                                   199
                                             ...
Romania, Bulgaria, Hungary                      1
Uruguay, Guatemala                              1
France, Senegal, Belgium                        1
Mexico, United States, Spain, Colombia          1
United Arab Emirates, Jordan                    1
Name: country, Length: 748, dtype: int64
```

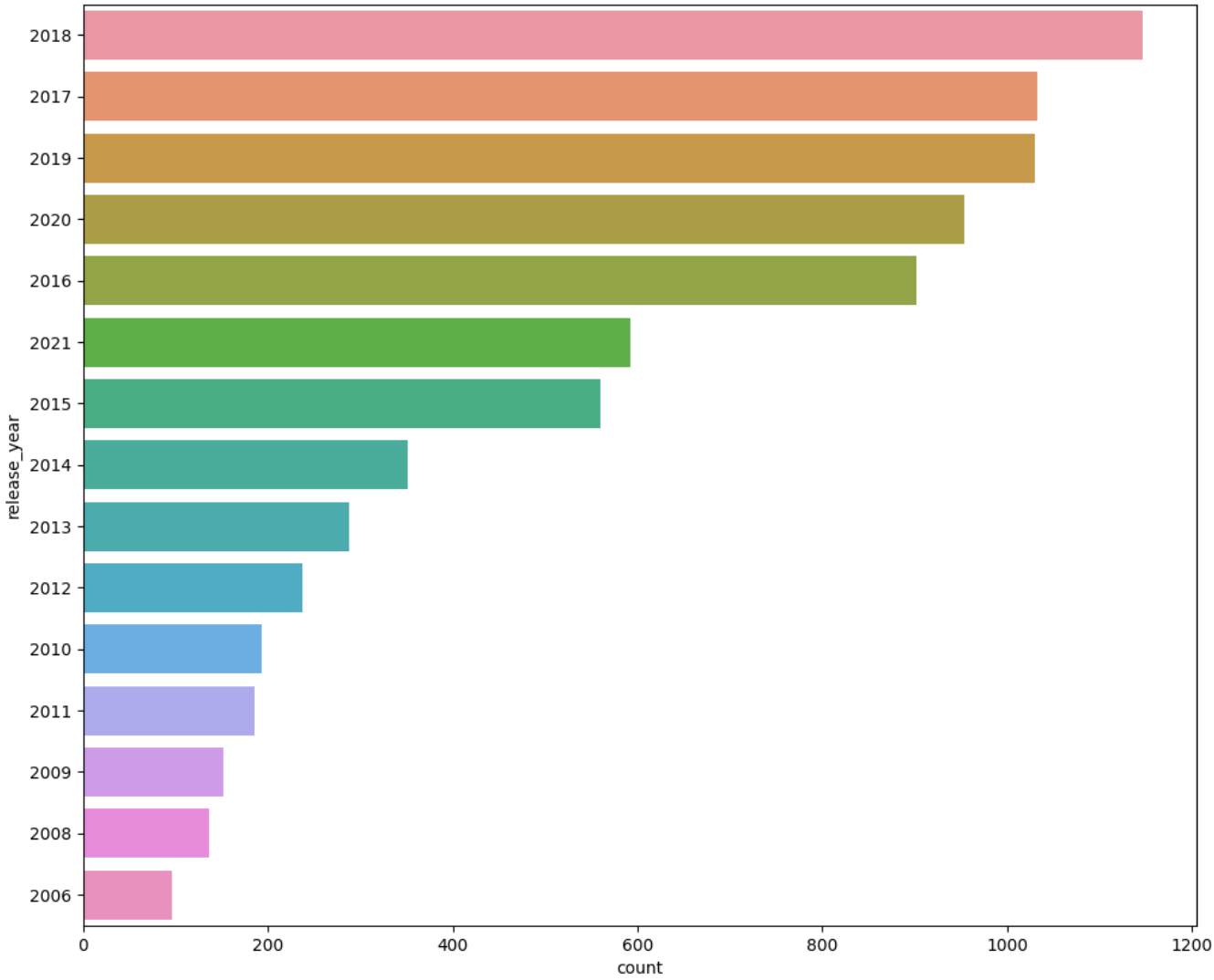
Top 10 Countries

```
df.country.value_counts().head(10)
```

```
United States     2818
India              972
United Kingdom     419
Japan              245
South Korea        199
Canada             181
Spain              145
France             124
Mexico             110
Egypt              106
Name: country, dtype: int64
```

Year Wise Counts

```
plt.figure(figsize=(12,10))
ax=sns.countplot(y='release_year',data=df, order=df.release_year.value_counts().index[0:15])
```



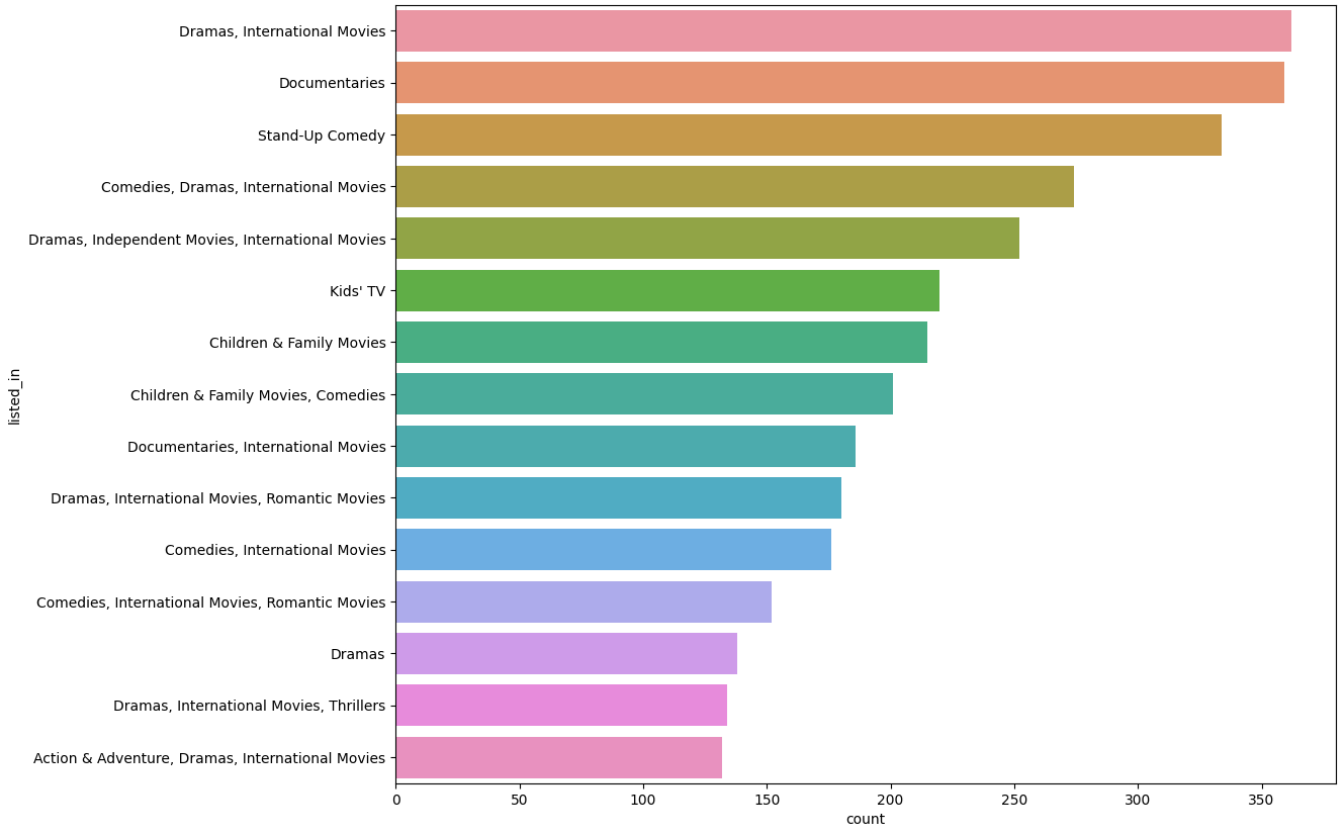
Genres

Double-click (or enter) to edit

```
df.listed_in.value_counts().head(10)
```

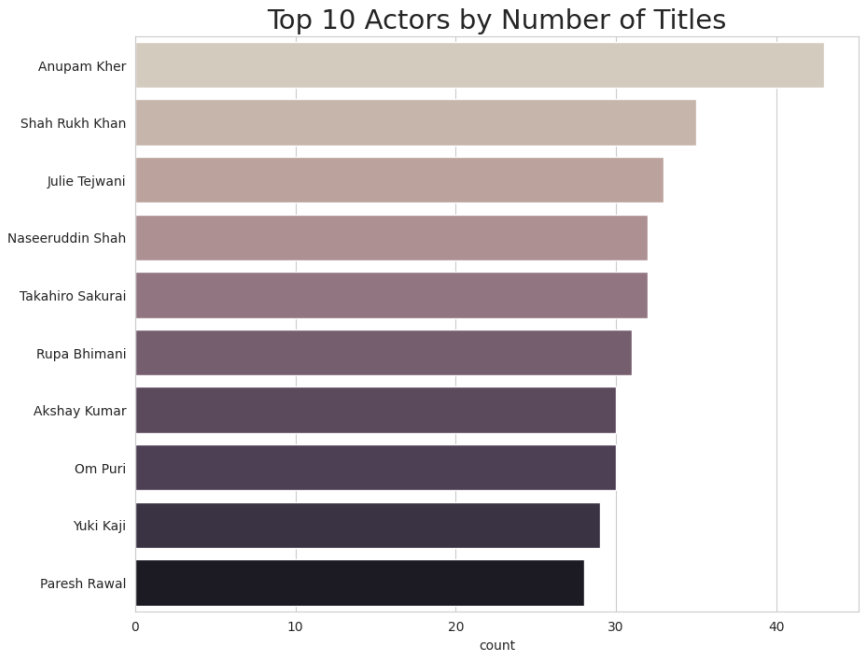
```
Dramas, International Movies                        362
Documentaries                                       359
Stand-Up Comedy                                     334
Comedies, Dramas, International Movies              274
Dramas, Independent Movies, International Movies    252
Kids' TV                                            220
Children & Family Movies                            215
Children & Family Movies, Comedies                  201
Documentaries, International Movies                 186
Dramas, International Movies, Romantic Movies       180
Name: listed_in, dtype: int64
```

```
plt.figure(figsize=(12,10))
ax=sns.countplot(y='listed_in',data=df, order=df.listed_in.value_counts().index[0:15])
```



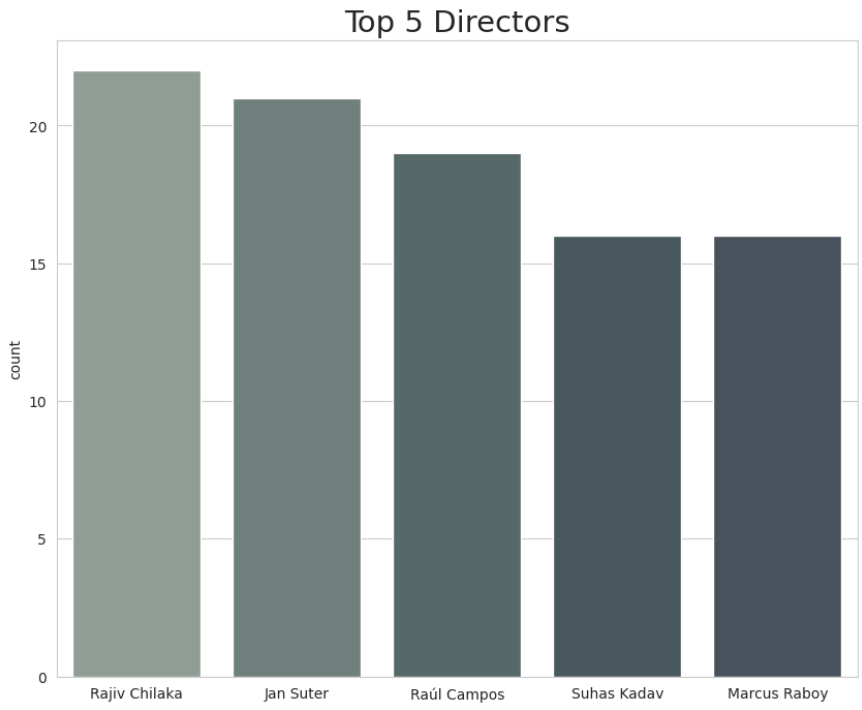
Top 10 Actors by Number of Titles

```
plt.figure(figsize=(10,8))
netflix_cast = df[df.cast != 'No Cast'].set_index('title').cast.str.split(', ', expand=True).stack().reset_index(level=1, drop=True)
sns.countplot(y = netflix_cast, order=netflix_cast.value_counts().index[:10], palette='magma_r', saturation=.2)
plt.title('Top 10 Actors by Number of Titles', fontsize=21);
plt.show()
```



Top 5 Directors

```
plt.figure(figsize=(10,8))
netflix_directors = df[df.director != 'No Director'].set_index('title').director.str.split(', ', expand=True).stack().reset_index(level=1
sns.countplot(x = netflix_directors, order=netflix_directors.value_counts().index[:5], palette='crest', saturation=.2)
plt.title('Top 5 Directors', fontsize=21)
plt.show()
```



Handling Missing Values

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

```
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
```

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

```
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
```

Dropping rows for small percentage of Null

```
df.shape
```

```
(8807, 12)
```

```
df.dropna(subset=['rating','duration'],axis=0,inplace=True)
```

```
df.shape
```

```
(8800, 12)
```

```
df.dropna(subset=['date_added'],axis=0,inplace=True)
```

```
df.shape
```

```
(8790, 12)
```

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

```
director        2621
country          829
cast             825
show_id            0
type               0
title              0
date_added         0
release_year       0
rating             0
duration           0
listed_in          0
description        0
dtype: int64
```

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

```
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
```

Replacing Missing Values in Countries with Unknown

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

```
df.country.value_counts().head()
```

```
United States     2809
India              972
Unknown            829
United Kingdom     418
Japan              243
Name: country, dtype: int64
```

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

```
director        29.82
cast             9.39
show_id          0.00
type             0.00
title            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
```

Replacing Missing Values in Cast with No Cast

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

```
df.cast.value_counts().head()
```

```
No Cast                                                                         825
David Attenborough                                                               19
```

```
Vatsal Dubey, Julie Tejwani, Rupa Bhimani, Jigna Bhardwaj, Rajesh Kava, Mousam, Swapnil     14
Samuel West                                                                                 10
Jeff Dunham                                                                                  7
Name: cast, dtype: int64
```

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

```
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
```

Replacing Missing Values in Director with No Director

```
df['director'].replace(np.NaN,'No Director',inplace=True)
```

```
df.director.value_counts().head()
```

```
No Director               2621
Rajiv Chilaka               19
Raúl Campos, Jan Suter      18
Suhas Kadav                 16
Marcus Raboy                16
Name: director, dtype: int64
```

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

```
show_id         0.0
type            0.0
title           0.0
director        0.0
cast            0.0
country         0.0
date_added      0.0
release_year    0.0
rating          0.0
duration        0.0
listed_in       0.0
description     0.0
dtype: float64
```

Creating Date Format of date_added as Release_Datetime

```
df['Release_Datetime']=pd.to_datetime(df['date_added'])
```

```
df['Release_year']=df['Release_Datetime'].dt.year
```

```
df = df.rename(columns = {"release_year" : "Release_Year"})
```

```
df.head()
```

	show_id	type	title	director	cast	country	date_added	Release_Year
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	No Cast	United States	September 25, 2021	2020
1	s2	TV Show	Blood & Water	No Director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021
					Sami			

```
df.dtypes
```

```
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
Release_Datetime    datetime64[ns]
dtype: object
```

We Will Create Two Seperate Dataframes netflix_movies_df,netflix_shows_df And Perform Some Of The Data Preparation

```
netflix_movies_df=df[df['type']=='Movie'].copy()
```

```
netflix_movies_df.head()
```

	show_id	type	title	director	cast	country	date_added	Release_Y
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	No Cast	United States	September 25, 2021	2
6	s7	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	Unknown	September 24, 2021	2
7	s8	Movie	Sankofa	Haile	Kofi Ghanaba, Oyafunmike	United States, Ghana, Burkina	September	1

```
netflix_shows_df=df[df['type']=='TV Show'].copy()
```

```
netflix_shows_df.head()
```

	show_id	type	title	director	cast	country	date_added	Release_Year
1	s2	TV Show	Blood & Water	No Director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel	Unknown	September 24, 2021	2021

```
netflix_shows_df.rename(columns={'duration':'Seasons'},inplace=True)
netflix_shows_df.replace({'Seasons':{'1 Season':'1 Seasons'}},inplace=True)
```

```
netflix_shows_df.Seasons=netflix_shows_df.Seasons.str.replace('Seasons','').astype(int)
```

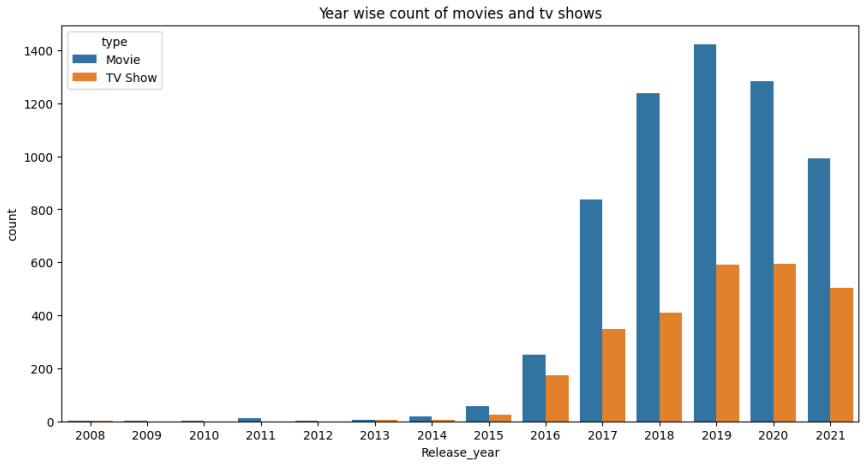
```
netflix_shows_df.head()
```

	show_id	type	title	director	cast	country	date_added	Release_Year
1	s2	TV Show	Blood & Water	No Director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel	Unknown	September 24, 2021	2021

```
df.head()
```

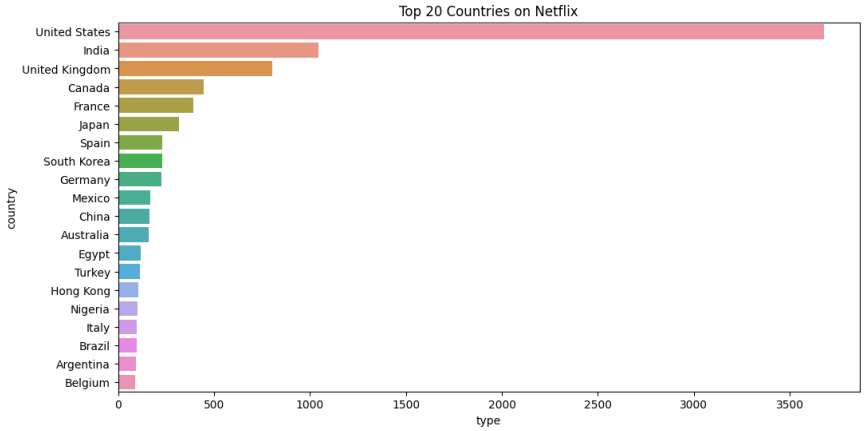
	show_id	type	title	director	cast	country	date_added	Release_Year
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	No Cast	United States	September 25, 2021	2020
1	s2	TV Show	Blood & Water	No Director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021

```
plt.figure(figsize=(12,6))
sns.countplot(data=df,x='Release_year',hue='type')
plt.title('Year wise count of movies and tv shows');
```



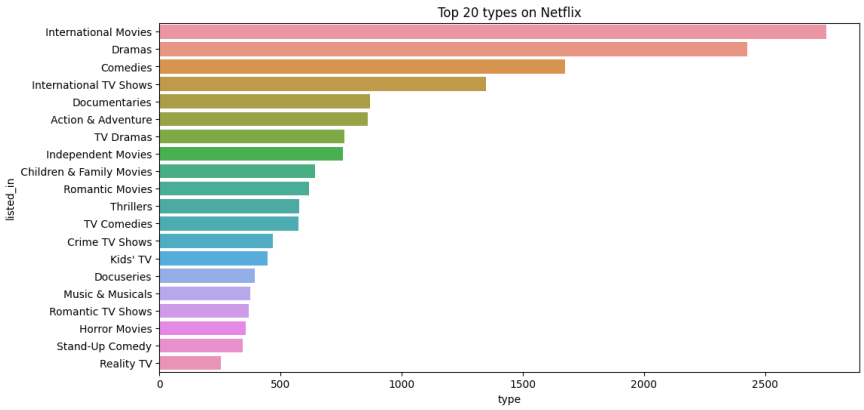
Top 20 Countries on Netflix

```
filtered_countries=df.set_index('type').country.str.split(', ',expand=True).stack().reset_index(level=1, drop=True)
filtered_countries=filtered_countries[filtered_countries !='Unknown']
filtered_countries.head()

plt.figure(figsize=(12,6))
sns.countplot(y=filtered_countries,order=filtered_countries.value_counts().index[:20])
plt.title('Top 20 Countries on Netflix')
plt.xlabel('type')
plt.ylabel('country')
plt.show()
```



Top 20 types on Netflix

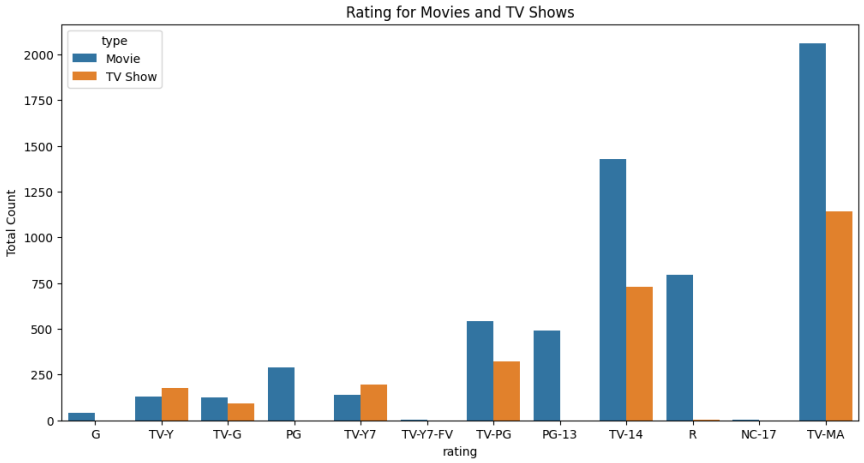
```
filtered_genre=df.set_index('type').listed_in.str.split(', ', expand=True).stack().reset_index(level=1 ,drop =True)

plt.figure(figsize=(12,6))
sns.countplot(y=filtered_genre,order=filtered_genre.value_counts().index[:20])
plt.title('Top 20 types on Netflix')
plt.xlabel('type')
plt.ylabel('listed_in')
plt.show()
```



Rating for Movies and TV Shows

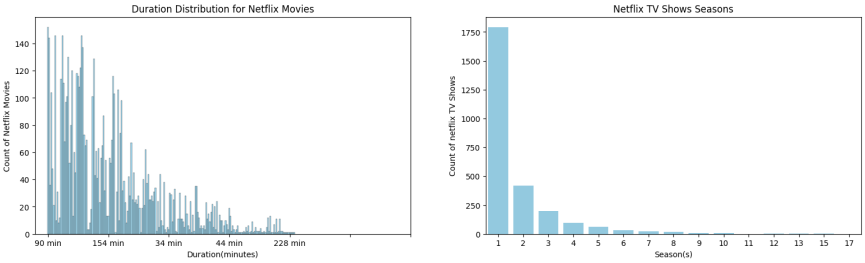
```
order=['G','TV-Y','TV-G','PG','TV-Y7','TV-Y7-FV','TV-PG','PG-13','TV-14','R','NC-17','TV-MA']
plt.figure(figsize=(12,6))
sns.countplot(x=df.rating,hue=df.type,order=order);
plt.title('Rating for Movies and TV Shows')
plt.xlabel('rating')
plt.ylabel('Total Count')
plt.show()
```

```
plt.show()
```



Netflix Movies And TV Shows Durations

```
fig,ax=plt.subplots(1,2,figsize=(19,5))
g1=sns.histplot(x=netflix_movies_df.duration,color='Skyblue',ax=ax[0])
g1.set_xticks(np.arange(0, 331, 50))
g1.set_title('Duration Distribution for Netflix Movies')
g1.set_ylabel('Count of Netflix Movies')
g1.set_xlabel('Duration(minutes)')
g2=sns.countplot(x=netflix_shows_df.Seasons,color='Skyblue',ax=ax[1])
g2.set_title('Netflix TV Shows Seasons')
g2.set_ylabel('Count of netflix TV Shows')
g2.set_xlabel('Season(s)')
fig.show()
```

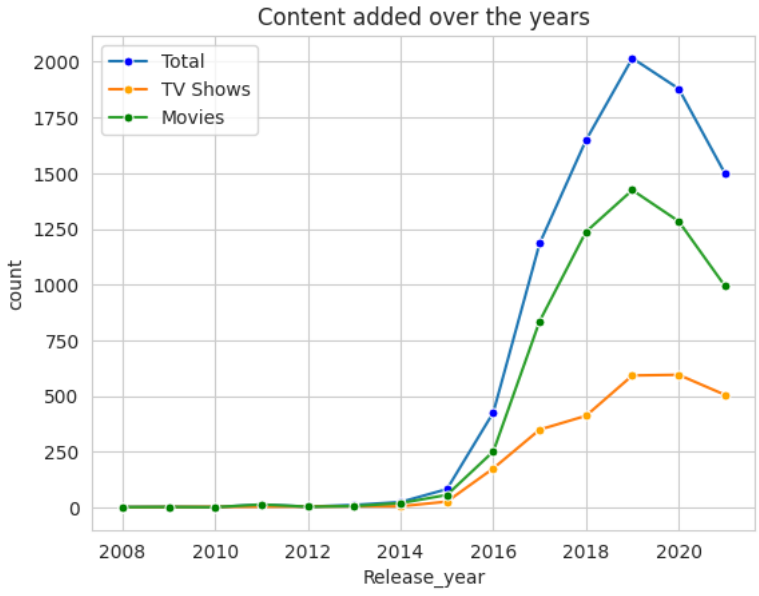


Content added over the years

```
d1 = df[df["type"] == "TV Show"]
d2 = df[df["type"] == "Movie"]

col = "Release_year"

vc1 = d1[col].value_counts().reset_index()

vc1 = vc1.rename(columns = {col : "count", "index" : col})
vc1['percent'] = vc1['count'].apply(lambda x : 100*x/sum(vc1['count']))
vc1 = vc1.sort_values(col)

vc2 = d2[col].value_counts().reset_index()
vc2 = vc2.rename(columns = {col : "count", "index" : col})
vc2['percent'] = vc2['count'].apply(lambda x : 100*x/sum(vc2['count']))
vc2 = vc2.sort_values(col)

vc3 = df[col].value_counts().reset_index()
vc3 = vc3.rename(columns = {col : "count", "index" : col})
vc3['percent'] = vc3['count'].apply(lambda x : 100*x/sum(vc3['count']))
vc3 = vc3.sort_values(col)

print(vc3)
```

```
    Release_year  count    percent
12          2008      2   0.022753
11          2009      2   0.022753
13          2010      1   0.011377
8           2011     13   0.147895
10          2012      3   0.034130
9           2013     11   0.125142
7           2014     24   0.273038
6           2015     82   0.932878
5           2016    426   4.846416
4           2017   1185  13.481229
2           2018   1648  18.748578
0           2019   2016  22.935154
1           2020   1879  21.376564
3           2021   1498  17.042093
```

```
sns.set_style("whitegrid")
plt.title("Content added over the years")
ax=sns.lineplot(x="Release_year",y="count",data=vc3,label="Total",marker='o', markerfacecolor='blue', markersize=5)
ax=sns.lineplot(x="Release_year",y="count",data=vc1,label="TV Shows",marker='o', markerfacecolor='orange', markersize=5)
ax=sns.lineplot(x="Release_year",y="count",data=vc2,label="Movies",marker='o', markerfacecolor='green', markersize=5)
plt.show()
```



Conclusions

• Country that produces the largest number of content titles on Netflix is the United States with 2,000++ content titles production.

• The genre with the largest number of content titles is International Movies with 1,700++ content.

• Rating with the largest number of Movies content is TV-MA with 1,400++ content titles and the rating with the largest number of TV Shows content is also TV-MA with 800++ content titles.

• The number of content titles on Netflix continued to increase from 2014 to 2019.

• The actor with the largest number of content titles on Netflix is Anupam Kher.

• The percentage of movies is 66.3% of the total content, while the percentage of TV shows is 33.7% of the total content.

• The Director with the largest number of content titles on Netflix is Rajiv Chilaka who has directed 20++ number of content titles on Netflix.

Recommendations:

Content Strategy:

Given the popularity of certain genres, Netflix should continue investing in and producing content that aligns with user preferences. This strategy can enhance user satisfaction and retention.

Release Calendar Optimization:

Understanding temporal trends in content additions can help Netflix optimize its release calendar. Strategic planning around peak user engagement periods can maximize the impact of new releases.

Promotion of High-Rated Content:

Netflix can capitalize on the identified top-rated shows and movies by featuring them prominently on the platform, tailoring promotional campaigns, and leveraging user reviews in marketing materials.

User Engagement Initiatives:

Consider implementing user engagement initiatives, such as personalized recommendations based on user preferences and viewing history. This can enhance the overall user experience and keep subscribers engaged.

Collaborations and Original Content:

Exploring collaborations with content creators and investing in original content within popular genres can contribute to the platform's uniqueness and competitiveness in the streaming market.

Continuous Monitoring:

Regularly monitoring user ratings, genre preferences, and other key metrics is crucial for staying agile and responsive to evolving user trends. This can inform ongoing content strategy adjustments.

In summary, the Netflix data analysis not only provided insights into current user behavior and content preferences but also offers strategic recommendations for content curation, promotional activities, and user engagement initiatives. By implementing these recommendations, Netflix can further solidify its position as a leading streaming platform in the competitive entertainment industry.