- Title: Netflix case study.
- Submitted by: Shivani Prajapati

```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns
```

Downloading the dataset

```
!gdown 12uE1LI_-ym7e1eOxcc3NBaRUTLUxMLPD
```

```
Downloading...
From: https://drive.google.com/uc?id=12uE1LI_-ym7e1eOxcc3NBaRUTLUxMLPD
To: /content/netflix_case_study.csv
100% 3.40M/3.40M [00:00<00:00, 19.6MB/s]
```

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

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

Next steps: View recommended plots New interactive sheet

```
netflix.columns
```

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

```
netflix.shape
```

```
(8807, 12)
```

The dataset has 12 features and 8807 rows.

```
netflix.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
```

```
netflix['rating'].value_counts()
```

	count
rating	
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

dtype: int64

Insights: We can see that last 3 values of the 'rating' column should be in 'duration' column.

```
netflix.loc[(netflix['rating'] == '74 min') | (netflix['rating'] == '84 min') | (netflix['rating'] == '66 min')]
netflix['duration'][[5541, 5794, 5813]] = netflix['rating'][[5541, 5794, 5813]]
netflix['rating'][[5541, 5794, 5813]] = 'NaN'
```

Show hidden output

```
netflix['rating'].value_counts()
```

rating	count
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
NaN	3
UR	3

dtype: int64

Convert date_added column to date_time format and type, rating column to categorical data type.

```
netflix['date_added'] = pd.to_datetime(netflix['date_added'], format = 'mixed')
```

```
netflix = netflix.astype({'type' : 'category', 'rating' : 'category'})
```

```
netflix.head(2)
```

	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	2021-09-25	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...

Next steps: View recommended plots New interactive sheet

```
netflix.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   category
 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   datetime64[ns]
 7   release_year  8807 non-null   int64
 8   rating        8803 non-null   category
 9   duration      8807 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: category(2), datetime64[ns](1), int64(1), object(8)
memory usage: 706.2+ KB
```

Exploratory Data Analysis

Non Graphical Analysis

Number of unique values in each column.

```
colm = netflix.columns
for col in colm:
  print('Unique values for column', col, '->', netflix[col].nunique())
  print()
```

```
Unique values for column show_id -> 8807

Unique values for column type -> 2

Unique values for column title -> 8807

Unique values for column director -> 4528

Unique values for column cast -> 7692

Unique values for column country -> 748

Unique values for column date_added -> 1714

Unique values for column release_year -> 74

Unique values for column rating -> 15

Unique values for column duration -> 220

Unique values for column listed_in -> 514

Unique values for column description -> 8775
```

```
netflix['release_year'].describe()
```

	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

dtype: float64

1. 25% of the total Tv shows and movies are from 1925 and 2013
2. 25% of the total Tv shows and movies are from 2019 and 2021

Conclusion - Netflix should add latest Movies and TV shows to attract more customers.

Null Values

```
netflix.isnull().sum()
```

	0
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

← Null values

dtype: int64

```
for col in netflix:
  na_count = (netflix[col].isnull().sum()*100) / (len(netflix))
  print(col, " ->", na_count, "%")
```

```
show_id  -> 0.0 %
type  -> 0.0 %
title  -> 0.0 %
director  -> 29.908027705234474 %
cast  -> 9.367548540933349 %
country  -> 9.43567616668559 %
date_added  -> 0.11354604292040422 %
release_year  -> 0.0 %
rating  -> 0.04541841716816169 %
duration  -> 0.0 %
listed_in  -> 0.0 %
description  -> 0.0 %
```

1. Approx 30% of director value is null in netflix dataframe.
2. Approx 9% of cast value are missing.
3. Approx 9% of country value are missing.

1. Imputing missing values of date_added column with minimum value.
2. Changing null values of rating column to zero.

```
min = netflix['date_added'].dropna().min()
netflix['date_added'].fillna(min, inplace = True)
```

```
<ipython-input-124-cbe2fb3f5604>:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained as
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col


  netflix['date_added'].fillna(min, inplace = True)
```

```
netflix['rating'] = netflix['rating'].cat.add_categories(['unknown_rating'])  # Add new category
netflix.loc[netflix['rating'].isna(), 'rating'] = 'unknown_rating'  # Now assign the value
```

```
netflix.isna().sum()
```

	0
show_id	0
type	0
title	0
director	2634
cast	825
country	831
date_added	0
release_year	0
rating	0
duration	0
listed_in	0
description	0

dtype: int64

Handling country column missing values.

For each genre, finding the country in which most TV show/ movies belong.

```
netflix_without_null_country = netflix.dropna(subset = ['country'])
country_per_genre = netflix_without_null_country.groupby('listed_in')['country'].value_counts().groupby(level=0).head(1)
country_per_genre = country_per_genre.reset_index()
country_per_genre
```

	listed_in	country	count
0	Action & Adventure	United States	64
1	Action & Adventure, Anime Features, Children &...	Japan	2
2	Action & Adventure, Anime Features, Classic Mo...	Japan	1
3	Action & Adventure, Anime Features, Horror Movies	Japan	1
4	Action & Adventure, Anime Features, Internatio...	Japan	32
...	...	...	...
493	TV Horror, TV Mysteries, Teen TV Shows	United States	1
494	TV Horror, Teen TV Shows	United States	2
495	TV Sci-Fi & Fantasy, TV Thrillers	Canada	1
496	TV Shows	United States	4
497	Thrillers	United States	43

498 rows × 3 columns

Next steps: View recommended plots New interactive sheet

Replacing null value of country column as per the genre.

```
# Impute missing countries
for index, row in netflix.iterrows():
  if pd.isnull(row['country']):
    for listed_in, country in zip(country_per_genre['listed_in'], country_per_genre['country']):
      if listed_in in row['listed_in']:
        netflix.loc[index, 'country'] = country
        break # Stop after finding the first match
```

```
netflix.isna().sum()
```

	0
show_id	0
type	0
title	0
director	2634
cast	825
country	0
date_added	0
release_year	0
rating	0
duration	0
listed_in	0
description	0

dtype: int64

Imputing missing cast values

Imputing the missing director values:

replacing the director value -> In a particular type, in a particular country, in a particular genre.

```
netflix['country'].replace({', France, Algeria': 'France, Algeria'}, inplace=True)

#netflix.loc[365]
```

```
<ipython-input-130-957a68341a51>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained as
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col

  netflix['country'].replace({', France, Algeria': 'France, Algeria'}, inplace=True)
```

```
netflix_director = netflix.dropna(subset = ['director'])
netflix_director_df = netflix_director.groupby(['type', 'country', 'listed_in'], observed=False)[['type', 'country', 'listed_in', 'direct

netflix_director_df.head()
```

	type	country	listed_in	director
0	Movie	United States	Documentaries	Kirsten Johnson
2	TV Show	United States	Crime TV Shows, International TV Shows, TV Act...	Julien Leclercq
5	TV Show	United States	TV Dramas, TV Horror, TV Mysteries	Mike Flanagan
6	Movie	United States	Children & Family Movies	Robert Cullen, José Luis Ucha
7	Movie	United States, Ghana, Burkina Faso, United Kin...	Dramas, Independent Movies, International Movies	Haile Gerima
...	...	...	...	...
8786	Movie	United Kingdom	Children & Family Movies	James Brown
8788	Movie	Croatia, Slovenia, Serbia, Montenegro	Dramas, International Movies	Ivona Juka
8794	Movie	Egypt, France	Dramas, Independent Movies, International Movies	Mohamed Diab
8801	Movie	United Arab Emirates, Jordan	Dramas, International Movies, Thrillers	Majid Al Ansari

```
# Impute missing directors
for row in netflix.itertuples(index = True):
  if pd.isnull(row.director):
    for (group_type, group_country, group_listed_in), group_df in netflix_director_df:
      if group_type == row.type and group_country == row.country and group_listed_in == row.listed_in:
        netflix.at[row.Index, 'director'] = group_df['director'].iloc[0]  # Get first matching director
        break  # Stop after first match
```

```
netflix.isna().sum()
```

```
                  0
     show_id      0
        type      0
       title      0
    director   1268
        cast    825
     country      0
  date_added      0
release_year      0
      rating      0
    duration      0
   listed_in      0
 description      0

dtype: int64
```

Imputation has some limitations. The remaining null values of the 'director' column (not being imputed) will be set to 'Unknown_director'.

```
netflix['director'].fillna('Unknown_director', inplace = True)
```

```
<ipython-input-134-f261201bb0e5>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained as
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col


  netflix['director'].fillna('Unknown_director', inplace = True)
```

```
netflix.isna().sum()
```

```
                 0
     show_id     0
        type     0
       title     0
    director     0
        cast   825
     country     0
  date_added     0
release_year     0
      rating     0
    duration     0
   listed_in     0
 description     0

dtype: int64
```

Imputing missing cast values

```
temp = netflix[['country', 'type', 'director', 'cast']]
temp.head()
```

```
         country     type          director                                               cast
0  United States    Movie   Kirsten Johnson                                                NaN
1   South Africa  TV Show  Unknown_director  Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...
2  United States  TV Show   Julien Leclercq     Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...
3  United States  TV Show  Unknown_director                                                NaN
4          India  TV Show  Unknown_director      Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...
```

Next steps: View recommended plots New interactive sheet

```
temp['cast'] = temp['cast'].dropna().apply(lambda x: x.split(',')).copy()
temp
```

```
<ipython-input-137-560f5a53594a>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus
  temp['cast'] = temp['cast'].dropna().apply(lambda x: x.split(',')).copy()
```

```
            country     type          director                                               cast
   0  United States    Movie   Kirsten Johnson                                                NaN
   1   South Africa  TV Show  Unknown_director  [Ama Qamata, Khosi Ngema, Gail Mabalane, Th...
   2  United States  TV Show   Julien Leclercq     [Sami Bouajila, Tracy Gotoas, Samuel Jouy, ...
   3  United States  TV Show  Unknown_director                                                NaN
   4          India  TV Show  Unknown_director     [Mayur More, Jitendra Kumar, Ranjan Raj, Al...
 ...            ...      ...               ...                                                ...
8802  United States    Movie     David Fincher     [Mark Ruffalo, Jake Gyllenhaal, Robert Downe...
8803  United States  TV Show  Unknown_director                                                NaN
8804  United States    Movie    Ruben Fleischer  [Jesse Eisenberg, Woody Harrelson, Emma Ston...
8805  United States    Movie      Peter Hewitt      [Tim Allen, Courteney Cox, Chevy Chase, Kat...
8806          India    Movie       Mozez Singh   [Vicky Kaushal, Sarah-Jane Dias, Raaghav Cha...

8807 rows × 4 columns
```

Next steps: View recommended plots New interactive sheet

```
temp.head(10)
```

```
                                             country     type                    director                                                cast
0                                      United States    Movie             Kirsten Johnson                                                 NaN
1                                       South Africa  TV Show            Unknown_director     [Ama Qamata, Khosi Ngema, Gail Mabalane, Th...
2                                      United States  TV Show             Julien Leclercq        [Sami Bouajila, Tracy Gotoas, Samuel Jouy, ...
3                                      United States  TV Show            Unknown_director                                                 NaN
4                                              India  TV Show            Unknown_director        [Mayur More, Jitendra Kumar, Ranjan Raj, Al...
5                                      United States  TV Show               Mike Flanagan        [Kate Siegel, Zach Gilford, Hamish Linklater...
6                                      United States    Movie  Robert Cullen, José Luis Ucha  [Vanessa Hudgens, Kimiko Glenn, James Marsde...
7  United States, Ghana, Burkina Faso, United Kin...    Movie                Haile Gerima    [Kofi Ghanaba, Oyafunmike Ogunlano, Alexandr...
8                                     United Kingdom  TV Show              Andy Devonshire       [Mel Giedroyc, Sue Perkins, Mary Berry, Pau...
9                                      United States    Movie              Theodore Melfi      [Melissa McCarthy, Chris O'Dowd, Kevin Kline...
```

Next steps: View recommended plots New interactive sheet

```
cast_list = temp.dropna(subset = ['cast']).explode('cast')
cast_list
```

```
      country     type    director               cast
1     South Africa TV Show Unknown_director    Ama Qamata
1     South Africa TV Show Unknown_director    Khosi Ngema
1     South Africa TV Show Unknown_director    Gail Mabalane
1     South Africa TV Show Unknown_director    Thabang Molaba
1     South Africa TV Show Unknown_director    Dillon Windvogel
...        ...       ...        ...                 ...
8806       India   Movie     Mozez Singh    Manish Chaudhary
8806       India   Movie     Mozez Singh    Meghna Malik
8806       India   Movie     Mozez Singh    Malkeet Rauni
8806       India   Movie     Mozez Singh    Anita Shabdish
8806       India   Movie     Mozez Singh  Chittaranjan Tripathy

64126 rows × 4 columns
```

Next steps: View recommended plots New interactive sheet

```
cast_groupby = cast_list.groupby(['country', 'director'], observed=False)['cast'].agg(list).reset_index()
cast_groupby.head(10)
```

```
   country                         director                                           cast
0  , South Korea           Unknown_director  [Jung Hae-in, Koo Kyo-hwan, Kim Sung-kyun, ...
1  Argentina               Alejandro Doria   [Luis Brandoni, China Zorrilla, Antonio Gasa...
2  Argentina             Alejandro Montiel   [Luisana Lopilato, Joaquín Furriel, Rafael F...
3  Argentina                   Ana Quiroga                                  [Luciana Aymar]
4  Argentina  Andy Caballero, Diego Corsini  [Franco Masini, Yamila Saud, Victorio D'Ales...
5  Argentina                  Carlos Sorín   [Valeria Bertuccelli, Esteban Lamothe, Juliá...
6  Argentina                 Daniel Burman   [Alan Sabbagh, Julieta Zylberberg, Usher Bar...
7  Argentina                 Daniela Goggi   [Eugenia Suárez, Esteban Lamothe, Gloria Car...
8  Argentina                  Diego Kaplan   [Carolina Ardohain, Mónica Antonópulos, Guil...
9  Argentina                 Eduardo Pinto   [Brian Maya, Matías Desiderio, Manuela Pal, ...
```

Next steps: View recommended plots New interactive sheet

```
for row in netflix.itertuples(index = True):
  if pd.isnull(row.cast):
    for col in cast_groupby.itertuples(index = False):
      if col.country == row.country and col.director == row.director:
        netflix.at[row.Index, 'cast'] = col.cast # Get first matching director
        break
```

Show hidden output

```
netflix.isna().sum()
```

	0
show_id	0
type	0
title	0
director	0
cast	429
country	0
date_added	0
release_year	0
rating	0
duration	0
listed_in	0
description	0

dtype: int64

Imputation method has some limitations. The remaining null cast values (not imputed) will be set to 'Unknown_cast'

```
netflix['cast'].fillna('Unknown_cast', inplace = True)
```

```
<ipython-input-144-1f14a7978c0b>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained as
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col


  netflix['cast'].fillna('Unknown_cast', inplace = True)
```

```
#convert list value to comma separated string value in a cast column
netflix['cast'] = netflix['cast'].apply(lambda x: ', '.join(x) if isinstance(x, list) else x)
```

```
netflix.isna().sum()
```

	0
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
description	0

dtype: int64

```
netflix
```

The data now has no null values in any of the column. Hence successfully imputed.

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	des
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Andy Puddicombe, Evelyn Lewis Prieto, Ginger...	United States	2021-09-25	2020	PG-13	90 min	Documentaries	As li
1	s2	TV Show	Blood & Water	Unknown_director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	par
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To far
3	s4	TV Show	Jailbirds New Orleans	Unknown_director	Blanca Suárez, Iván Marcos, Óscar Casas, Ad...	United States	2021-09-24	2021	TV-MA	1 Season	Docuseries, Reality TV	flirt to dc
4	s5	TV Show	Kota Factory	Unknown_director	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	
...	...	...	...	...	...	...	...	...	...	...	...	
					Mark Ruffalo, Jake						Cult Movies,	ca

Next steps: View recommended plots New interactive sheet

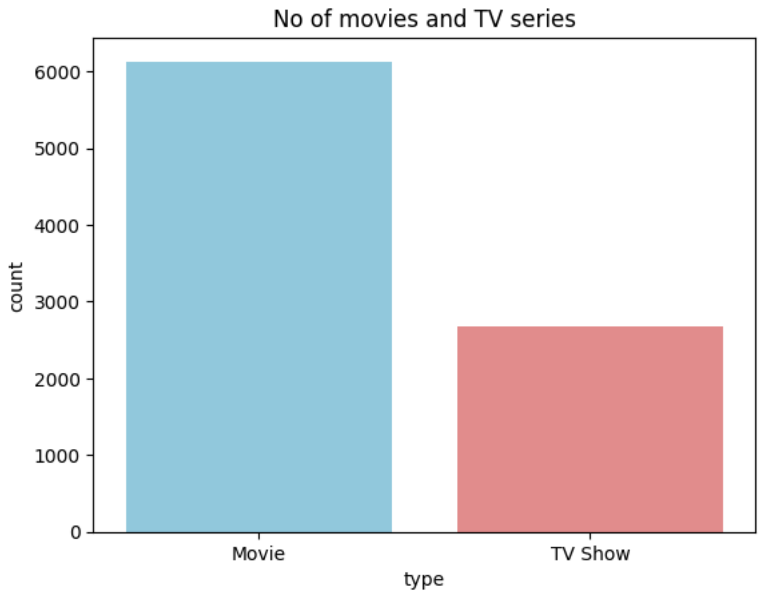
Graphical Analysis

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

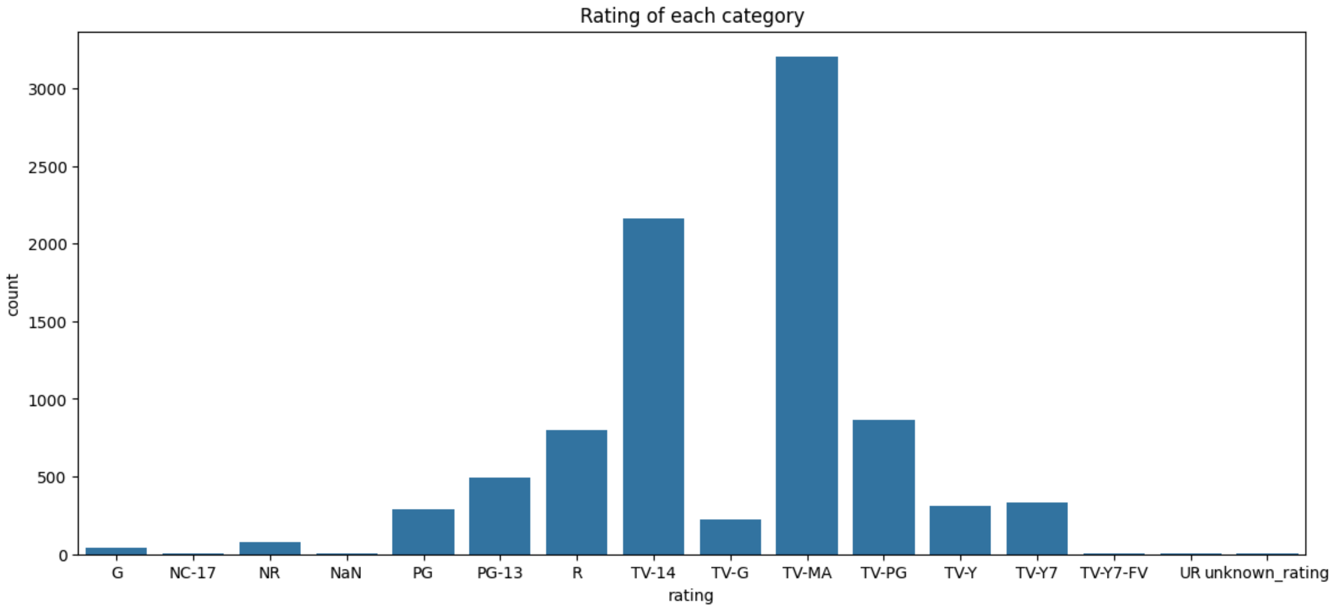
```
for col in netflix.select_dtypes(include=['category']).columns:
  print(netflix[col].value_counts(), "\n")
```

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

rating
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
unknown_rating       4
NC-17                3
NaN                  3
UR                   3
Name: count, dtype: int64
```

```
sns.countplot(data = netflix, x = 'type', hue = 'type', palette=['skyblue', 'lightcoral'], legend = False)
plt.title('No of movies and TV series')
plt.show()
```



```
plt.figure(figsize=(14, 6))
sns.countplot(data = netflix, x = 'rating')
plt.title('Rating of each category')
plt.show()
```



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

```
# number of movies produced in each country
only_movies = netflix[netflix['type'] == 'Movie']                              # filtering out only movies
movies_per_country = only_movies.groupby('country')['title'].count().reset_index()
movies_per_country.sort_values(by='title', ascending=False).iloc[0:10]        #pick the top 10 countries.
```

```
            country  title
524   United States   2482
217           India    893
439  United Kingdom    220
49           Canada    122
383           Spain     97
127           Egypt     92
318         Nigeria     86
277           Japan     78
237       Indonesia     77
427          Turkey     76
```

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

```
only_tvshow = netflix[netflix['type'] == 'TV Show']                              # filtering out only TV Shows.
tvshow_per_country = only_tvshow.groupby('country')['title'].count().reset_index()
tvshow_per_country.sort_values(by='title', ascending=False).iloc[0:10]        #pick the top 10 countries.
```

```
            country  title
160   United States   1068
140  United Kingdom    229
83            Japan    190
120     South Korea    158
66            India    106
132          Taiwan     68
47           France     59
17           Canada     59
4         Australia     48
94           Mexico     48
```

Find which is the best week to release the movie.

```
# Creating a new column containing week
netflix['week_of_year'] = netflix['date_added'].dt.isocalendar().week
only_movies = netflix[netflix['type'] == 'Movie']
movies_per_week = only_movies.groupby('week_of_year')['title'].count().reset_index()
movies_per_week.sort_values(by='title', ascending=False).iloc[0:10]
```

```
    week_of_year  title
0              1    316
43            44    243
39            40    215
8              9    207
25            26    195
34            35    189
30            31    185
12            13    174
17            18    173
26            27    154
```

Insights : Best week to release a movie is first week of the year. :

Find which is the best week to release the TV-show.

```
only_tvshow = netflix[netflix['type'] == 'TV Show']
only_tvshow = netflix[netflix['type'] == 'TV Show']
tvshow_per_week = only_tvshow.groupby('week_of_year')['title'].count().reset_index()   #finding week-wise count of tv shows released.
tvshow_per_week.sort_values(by='title', ascending=False).iloc[0:10]                   #top 10 output
```

	week_of_year	title
26	27	86
30	31	83
12	13	76
43	44	75
23	24	75
34	35	74
4	5	73
25	26	73
39	40	72
49	50	70

Insights : Best week to release a TV Show is 27th week of the year. :

Find which is the best month to release the movie

```
# creating a new month column
netflix['month_of_year'] = netflix['date_added'].dt.strftime('%B')
only_movies = netflix[netflix['type'] == 'Movie']         #filtering out only movies
movies_per_month = only_movies.groupby('month_of_year')['title'].count().reset_index()   #finding month-wise count of movies released
movies_per_month.sort_values(by='title', ascending=False)     # filtering the data in descending order
```

	month_of_year	title
5	July	565
0	April	550
2	December	547
4	January	546
10	October	545
7	March	529
1	August	519
11	September	519
9	November	498
6	June	492
8	May	439
3	February	382

Insights: Best month to release a movie is July.

Find which is the best month to release the TV Show

```
only_tvshow = netflix[netflix['type'] == 'TV Show']
tvshow_per_month = only_tvshow.groupby('month_of_year')['title'].count().reset_index()     #finding month-wise count of tv shows released
tvshow_per_month.sort_values(by='title', ascending=False)
```

	month_of_year	title
2	December	266
5	July	262
11	September	251
1	August	236
6	June	236
10	October	215
0	April	214
7	March	213
9	November	207
4	January	202
8	May	193
3	February	181

Insights: Best month to release a TV Show is December.

Identify the top 10 directors who have appeared in most movies.

```
only_movies = netflix[netflix['type']== 'Movie'] # filtering just movies
only_movies.groupby('director')['title'].count().reset_index().sort_values(by='title', ascending=False).iloc[0:10]
```

	director	title
4136	Unknown_director	24
3795	Spike Lee	20
2202	Kirsten Johnson	20
3907	Susan Lacy	20
3252	Rajiv Chilaka	19
3303	Raúl Campos, Jan Suter	18
3885	Suhas Kadav	16
3400	Robert Cullen, José Luis Ucha	16
2492	Marcus Raboy	15
1716	Jay Karas	14

Identify the top 10 directors who have appeared in most TV Shows.

```
only_tvshow = netflix[netflix['type']== 'TV Show'] # filtering just movies
only_tvshow.groupby('director')['title'].count().reset_index().sort_values(by='title', ascending=False).iloc[0:10]
```

	director	title
214	Unknown_director	1244
218	Vijay S. Bhanushali	121
141	Michael Simon	85
213	Tsutomu Mizushima	78
58	Garrett Bradley	62
180	Ryan Polito	62
162	Park Joon-hwa	59
72	Hsu Fu-chun	58
110	Kenny Ortega	55
20	Billy Corben	39

Identify the top 10 actors who have appeared in most movies.

```
temp1 = netflix[netflix['type'] == 'Movie'][['title', 'cast']]
temp1['cast'] = temp1['cast'].dropna().apply(lambda x: x.split(','))
```

```
exploded_cast = temp1.explode('cast')

exploded_cast.groupby('cast')['title'].count().reset_index().sort_values('title', ascending = False).iloc[1:11]
# Since cast is unknown in maximum cases. we are excluding that case.
```

	cast	title
3114	Anupam Kher	38
405	Jigna Bhardwaj	31
439	Julie Tejwani	28
813	Rajesh Kava	28
866	Rupa Bhimani	28
18003	Om Puri	27
20465	Rupa Bhimani	27
28543	Shah Rukh Khan	26
18247	Paresh Rawal	25
4284	Boman Irani	25

Identify the top 10 actors who have appeared in most TV Shows.

```
temp2 = netflix[netflix['type'] == 'TV Show'][['title', 'cast']]

temp2['cast'] = temp2['cast'].dropna().apply(lambda x: x.split(','))
exploded_cast_tv = temp2.explode('cast') #unnesting cast column

exploded_cast_tv.groupby('cast')['title'].count().reset_index().sort_values('title', ascending = False).iloc[0:10]
```

	cast	title
5692	Sean Astin	136
6038	Steven Yeun	123
1971	Fred Tatasciore	123
3372	Kevin Michael Richardson	114
3193	Kari Wahlgren	110
2159	Grey Griffin	107
6475	Um Sang-hyun	105
2357	Hong Bum-ki	103
6348	Tom Kenny	99
1469	David Harbour	99

```
netflix
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	des
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Andy Puddicombe, Evelyn Lewis Prieto, Ginger...	United States	2021-09-25	2020	PG-13	90 min	Documentaries	As li
1	s2	TV Show	Blood & Water	Unknown_director	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	pai
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	2021-09-24	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To far
3	s4	TV Show	Jailbirds New Orleans	Unknown_director	Blanca Suárez, Iván Marcos, Óscar Casas, Ad...	United States	2021-09-24	2021	TV-MA	1 Season	Docuseries, Reality TV	flirt to do
4	s5	TV Show	Kota Factory	Unknown_director	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	
...	...	...	...	...	...	...	...	...	...	...	...	
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	2019-11-20	2007	R	158 min	Cult Movies, Dramas, Thrillers	ca re
8803	s8804	TV Show	Zombie Dumb	Unknown_director	Blanca Suárez, Iván Marcos, Óscar Casas, Ad...	United States	2019-07-01	2018	TV-Y7	2 Seasons	Kids' TV, Korean TV Shows, TV Comedies	V spo a
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	2019-11-01	2009	R	88 min	Comedies, Horror Movies	s w ov
					Tim Allen,							

Next steps: View recommended plots New interactive sheet

Which genre movies are more popular or produced more

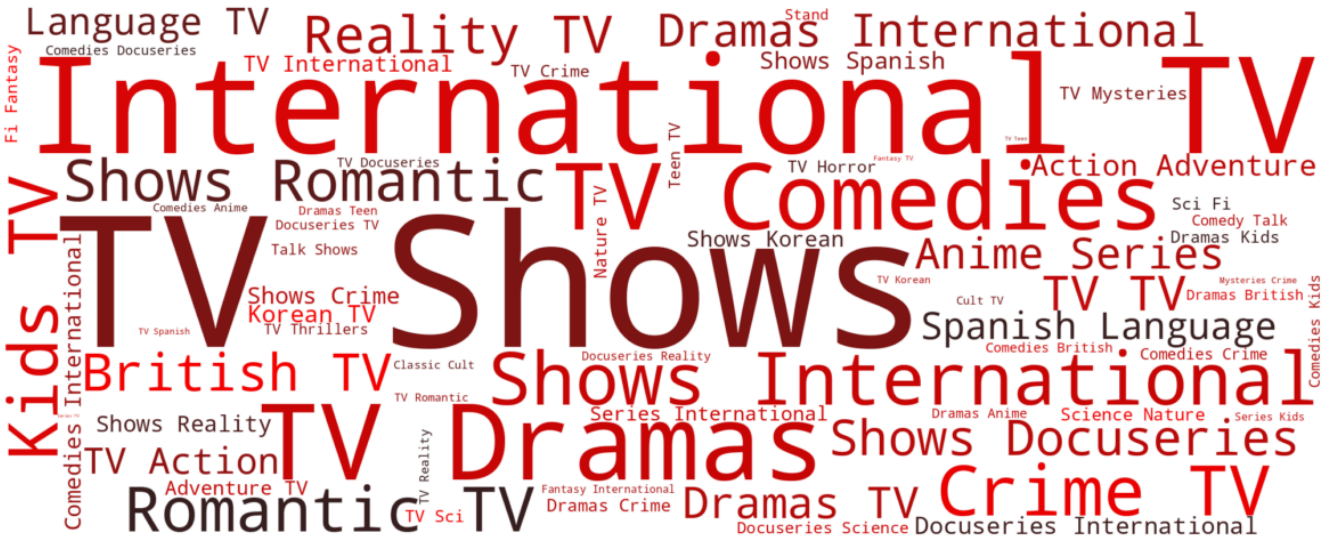
```
pip install wordcloud matplotlib
```

```
Requirement already satisfied: wordcloud in /usr/local/lib/python3.11/dist-packages (1.9.4)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (3.10.0)
Requirement already satisfied: numpy>=1.6.1 in /usr/local/lib/python3.11/dist-packages (from wordcloud) (1.26.4)
Requirement already satisfied: pillow in /usr/local/lib/python3.11/dist-packages (from wordcloud) (11.1.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.3.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (4.56.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (24.2)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (3.2.1)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.7->matplotlib) (1.17.0)
```

```
from wordcloud import WordCloud
```

```
tv_genre = netflix[netflix['type'] == 'TV Show']

text = str(list(tv_genre['listed_in'])).replace(',','').replace("'",'').replace('"','').replace('[','').replace(']','')

color = sns.color_palette("dark:red", as_cmap=True)

wordcld = WordCloud(max_words = 150, width = 2000,  height = 800,background_color = 'white',colormap = color).generate(text)
```

```
plt.figure(figsize=(15, 7))
plt.imshow(wordcld,interpolation = 'bilinear')
plt.axis('off')
plt.show()
```



Insights

1. Popular Movie genres on Netflix include International Movies, Comedies, Dramas, Action, and Romantic films.
2. Among TV Shows on Netflix, popular genres encompass Drama, Crime, Romance, Kids' content, Comedies, and International series.

Find After how many days the movie will be added to Netflix after the release of the movie (you can consider the recent past data)

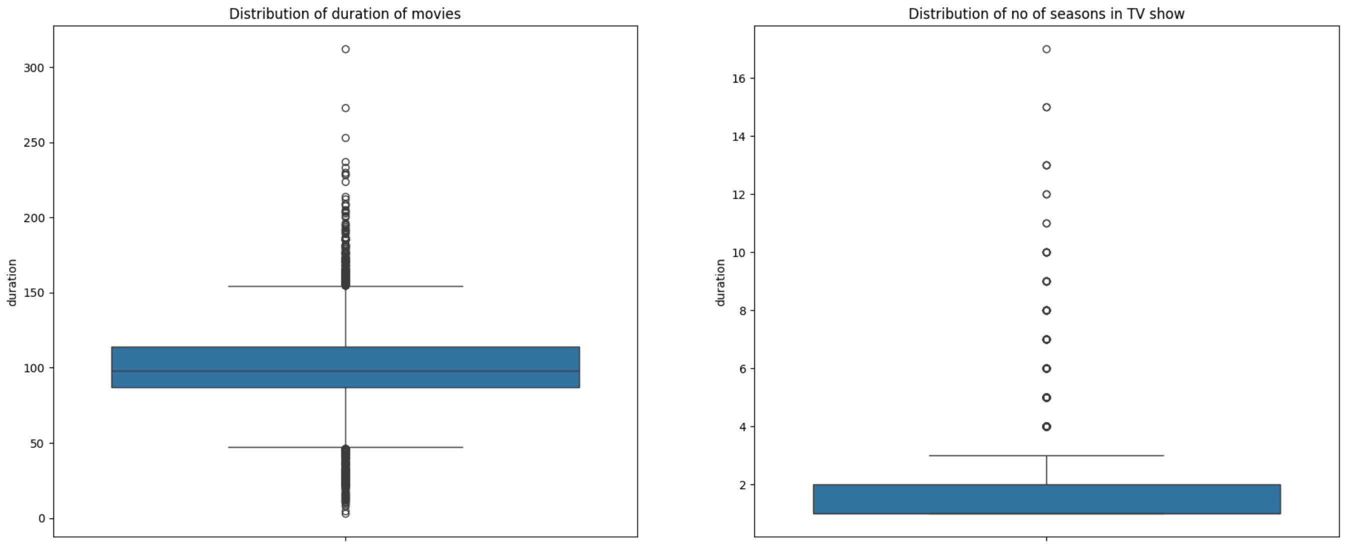
```
netflix["added_year"] = netflix["date_added"].dt.year    # extracting year from date_added column and creating new column out of it.
netflix["year_difference"] = netflix["added_year"] - netflix["release_year"]    # finding difference of two columns.
mode_difference = netflix["year_difference"].mode()[0]                          # finding mode


print("Mode of the year difference:", mode_difference)
```

```
Mode of the year difference: 0
```

Since mode turns out to be 0, the best time to add a movie/ TV show in Netflix is as soon as the movie is released.

Graphical analysis

```
plt.figure(figsize=(20,8))
duration_df = netflix.loc[netflix["duration"].str.contains("min")== True]["duration"].apply(lambda x: x.split()[0]).astype(int)  # splt:
plt.subplot(1,2,1) #subplots to make the data look easy for comparison.
sns.boxplot(duration_df)
plt.title("Distribution of duration of movies")
duration_seson_df = netflix.loc[netflix["duration"].str.contains("Season")== True]["duration"].apply(lambda x: x.split()[0]).astype(int
plt.subplot(1,2,2)
sns.boxplot(duration_seson_df)
plt.title("Distribution of no of seasons in TV show")
plt.show()
```



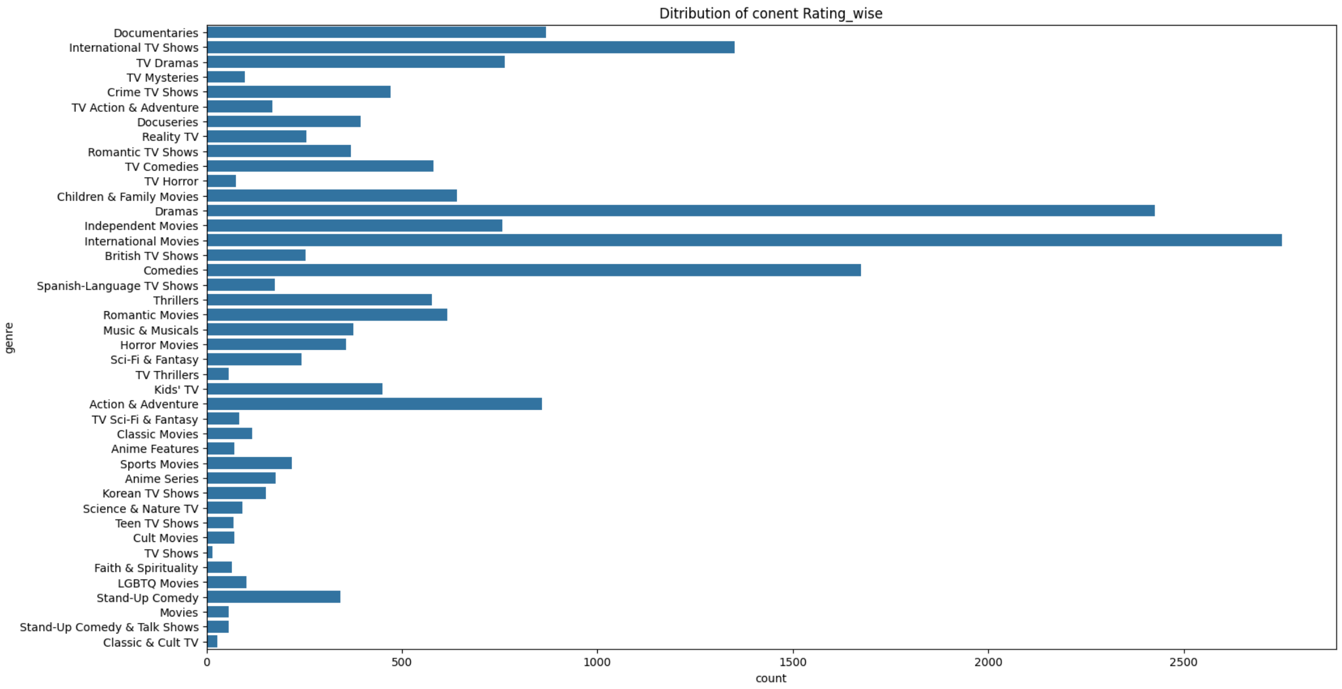
1. Average duration of movies are around 100 min
2. TV shows mostly are having 1 or 2 seasons.
3. There are lot of outliers present in movies as compare to TV shows

```
#exploding listed_in column
listed_in = netflix["listed_in"].apply(lambda x: str(x).split(", ")).tolist()
df_genre = pd.DataFrame(listed_in, index = netflix["title"])
df_genre = df_genre.stack()
df_genre = df_genre.reset_index()
df_genre.drop(columns = "level_1" , inplace = True)
df_genre.columns = ["title" , "genre"]
df_genre.head()
```

	title	genre
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

Next steps: View recommended plots New interactive sheet

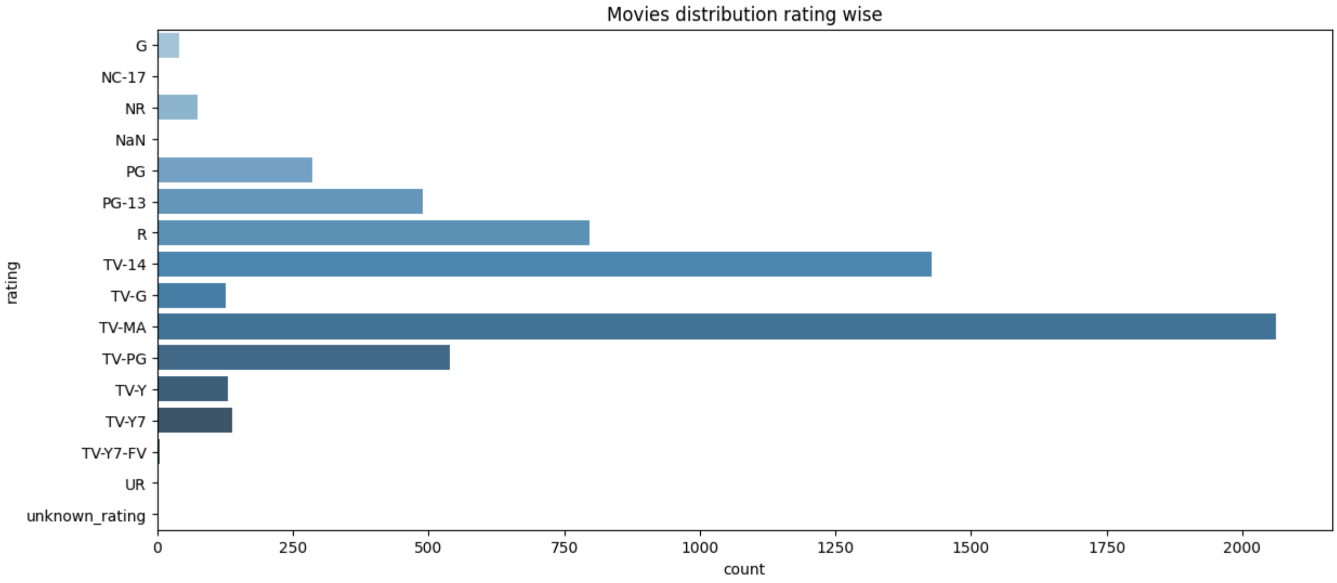
```
plt.figure(figsize = (18,10))
sns.countplot(y = "genre" , data =df_genre )
plt.title("Ditribution of conent Rating_wise")
plt.show()
```



Most appearing category in netflix movies and TV shows are:-

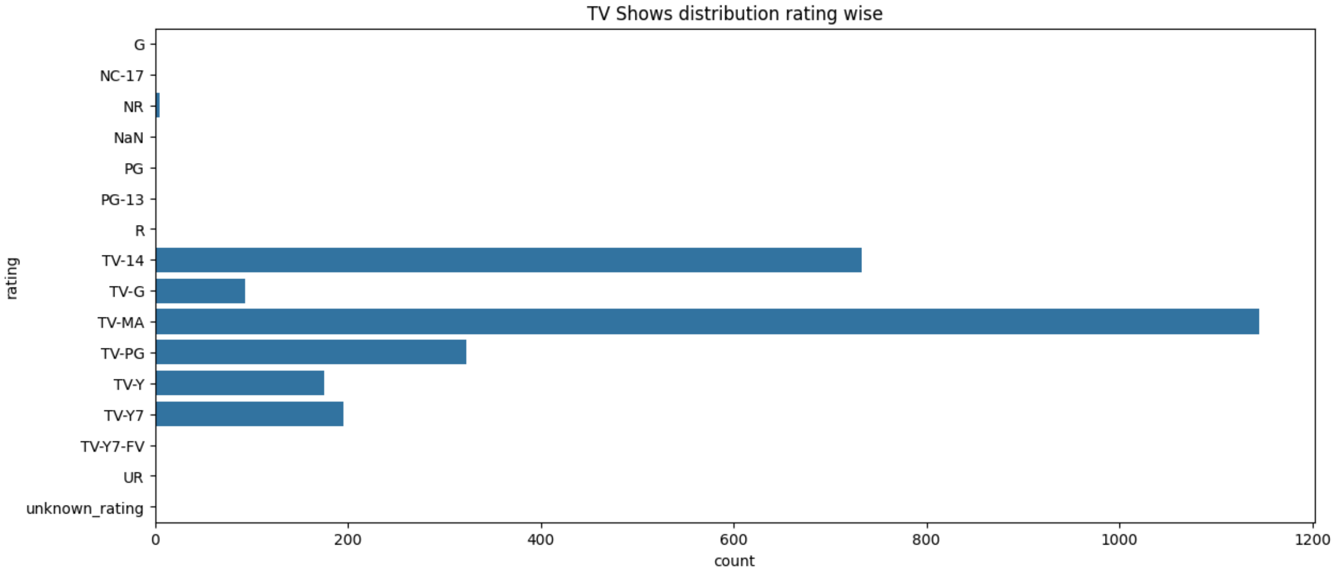
1. International Movies
2. Dramas
3. Comedies
4. International TV show

```
plt.figure(figsize=(14,6))
movies_ratingwise = netflix.loc[netflix["type"] == "Movie" , ["type" , "rating"]]
sns.countplot( y="rating" ,hue = 'rating', data =movies_ratingwise,  palette="Blues_d" )
plt.title("Movies distribution rating wise")
plt.show()
```



Mostly movies are belongs to TV-MA & TV-14 rating.

```
plt.figure(figsize=(14,6))
movies_ratingwise = netflix.loc[netflix["type"] == "TV Show" , ["type" , "rating"]]
sns.countplot( y="rating" , data =movies_ratingwise)
plt.title("TV Shows distribution rating wise")
plt.show()
```



Mostly TV Shows are belongs to TV-MA & TV-14 rating.

Summary :-

1. Netflix added more movies as compare to TV shows
2. Content for United States on netflix is maximum as compare to other countries.
3. Netflix content is mostly availabe for adults only
4. Most popular genres in recent years are International movies, Dramas, Comedies, International TV Shows and Action & Adventure.
5. In 2021 , there is significant amount of drop in content added due to COVID pandemic.
6. Most of viewers of Netflix is from United States followed by India & United Kingdom.

Recommendations :

Movies :-

1. Preferred movies duration is between 90-100 minutes.
2. Netflix should add more movies for United States and India falling in category of Internation movies and comedies
3. Netflix should add more movies for United States and India having rating of TV-MA & TV-14.
4. Top three countries where movies added are United States, India & United Kingdom.
5. Netflix shoud add TV Show on Friday than any other weekday.

TV Show:-

1. Preferred movies duration is 1-2 seeasons.
2. Netflix should focus on countries like Japan, South Korea and France in TV shows, as they prefer TV shows over movies.
3. Netflix should add TV Show on Friday than other weekday.
4. As per 2021 data, count of TV shows are more than movies, this means people want more web-series as they have for leisure time may be due to work from home scenario.

Double-click (or enter) to edit

Double-click (or enter) to edit

Double-click (or enter) to edit