

Business Case: Netflix - Data Exploration and Visualisation

```
In [11]: import pandas as pd
import numpy as np
```

```
df = pd.read_csv('netflix.csv')
df.head()
```

Out[11]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	See
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	15
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	15
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	See



Defining Problem Statement and Analysing basic metrics

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

- identify what kind of movies and TV shows are highly rated
- which are the countries where more highly rated content can be released
- see how user / content preferences have changed over the years

Observations on the shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required), missing value detection, statistical summary

overall shape of the data and datatype of all the attributes

In [12]: `df.shape`

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

In [13]: `# data-type of each column
df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype  
--- 
 0   show_id           8807 non-null    object  
 1   type              8807 non-null    object  
 2   title             8807 non-null    object  
 3   director          6173 non-null    object  
 4   cast               7982 non-null    object  
 5   country            7976 non-null    object  
 6   date_added        8797 non-null    object  
 7   release_year      8807 non-null    int64  
 8   rating             8803 non-null    object  
 9   duration           8804 non-null    object  
 10  listed_in          8807 non-null    object  
 11  description        8807 non-null    object  
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

In [14]: `df.columns`

Out[14]: `Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
 'release_year', 'rating', 'duration', 'listed_in', 'description'],
 dtype='object')`

Observation:

- The data is in the form of a 2D array containing 8807 rows and 12 columns
- The column names are: 'show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',

- 'release_year', 'rating', 'duration',
'listed_in', 'description'
- only the year column is of int64 data type
- **Missing value detection:** as can be seen columns
director, cast, country, date_added, rating and duration
have missing values since the non-null count is lower than
the number of 8807

categorical attributes analysis

```
In [15]: df['type'].unique()
```

```
Out[15]: array(['Movie', 'TV Show'], dtype=object)
```

Observation: there are 2 types of content - 'Movie' and 'TV Show'

```
In [16]: df['director'].unique().shape
```

```
Out[16]: (4529,)
```

Observation: There is content from 4529 directors on Netflix, but this is not accurate since
there are rows where there are multiple directors clubbed together and separated by
commas.

```
In [17]: df['country'].unique()
```

```
Out[17]: array(['United States', 'South Africa', 'nan', 'India',  
   'United States, Ghana, Burkina Faso, United Kingdom, Germany, Ethiopia',  
   'United Kingdom', 'Germany, Czech Republic', 'Mexico', 'Turkey',  
   'Australia', 'United States, India, France', 'Finland',  
   'China, Canada, United States',  
   'South Africa, United States, Japan', 'Nigeria', 'Japan',  
   'Spain, United States', 'France', 'Belgium',  
   'United Kingdom, United States', 'United States, United Kingdom',  
   'France, United States', 'South Korea', 'Spain',  
   'United States, Singapore', 'United Kingdom, Australia, France',  
   'United Kingdom, Australia, France, United States',  
   'United States, Canada', 'Germany, United States',  
   'South Africa, United States', 'United States, Mexico',  
   'United States, Italy, France, Japan',  
   'United States, Italy, Romania, United Kingdom',  
   'Australia, United States', 'Argentina, Venezuela',  
   'United States, United Kingdom, Canada', 'China, Hong Kong',  
   'Russia', 'Canada', 'Hong Kong', 'United States, China, Hong Kong',  
   'Italy, United States', 'United States, Germany',  
   'United Kingdom, Canada, United States', 'South Korea',  
   'Ireland', 'India, Nepal',  
   'New Zealand, Australia, France, United States', 'Italy',  
   'Italy, Brazil, Greece', 'Argentina', 'Jordan', 'Colombia',  
   'United States, Japan', 'Belgium, United Kingdom',  
   'Switzerland, United Kingdom, Australia', 'Israel, United States',  
   'Canada, United States', 'Brazil', 'Argentina, Spain', 'Taiwan',  
   'United States, Nigeria', 'Bulgaria, United States',  
   'Spain, United Kingdom, United States', 'United States, China',  
   'United States, France',  
   'Spain, France, United Kingdom, United States',  
   'France, Algeria', 'Poland', 'Germany',  
   'France, Israel, Germany, United States, United Kingdom',  
   'New Zealand', 'Saudi Arabia', 'Thailand', 'Indonesia',  
   'Egypt, Denmark, Germany', 'United States, Switzerland',  
   'Hong Kong, Canada, United States', 'Kuwait, United States',  
   'France, Canada, United States, Spain',  
   'France, Netherlands, Singapore', 'France, Belgium',  
   'Ireland, United States, United Kingdom', 'Egypt', 'Malaysia',  
   'Israel', 'Australia, New Zealand', 'United Kingdom, Germany',  
   'Belgium, Netherlands', 'South Korea, Czech Republic',  
   'Australia, Germany', 'Vietnam', 'United Kingdom, Belgium',  
   'United Kingdom, Australia, United States',  
   'France, Japan, United States',  
   'United Kingdom, Germany, Spain, United States',  
   'United Kingdom, United States, France, Italy',  
   'United States, Germany, Canada',  
   'United States, France, Italy, United Kingdom',  
   'United States, United Kingdom, Germany, Hungary',  
   'United States, New Zealand', 'Sweden', 'China', 'Lebanon',  
   'Romania', 'Finland, Germany', 'Lebanon, Syria', 'Philippines',  
   'Iceland', 'Denmark', 'United States, India',  
   'Philippines, Singapore, Indonesia',  
   'China, United States, Canada', 'Lebanon, United Arab Emirates',  
   'Canada, United States, Denmark', 'United Arab Emirates',  
   'Mexico, France, Colombia', 'Netherlands',  
   'Germany, United States, France', 'United States, Bulgaria',  
   'United Kingdom, France, Germany, United States',  
   'Norway, Denmark', 'Syria, France, Lebanon, Qatar',  
   'United States, Czech Republic', 'Mauritius',  
   'Canada, South Africa', 'Austria', 'Mexico, Brazil',  
   'Germany, France', 'Mexico, United States',  
   'United Kingdom, France, Spain, United States',  
   'United States, Australia',  
   'United States, United Kingdom, France', 'United States, Russia',
```

'United States, United Kingdom, New Zealand',
'Australia, United Kingdom', 'Canada, Nigeria, United States',
'France, United States, United Kingdom, Canada',
'France, United Kingdom', 'India, United Kingdom',
'Canada, United States, Mexico',
'United Kingdom, Germany, United States',
'Czech Republic, United Kingdom, United States',
'China, United Kingdom', 'Italy, United Kingdom', 'China, Taiwan',
'United States, Brazil, Japan, Spain, India',
'United States, China, United Kingdom', 'Cameroon',
'Lebanon, Palestine, Denmark, Qatar', 'Japan, United States',
'Uruguay, Germany', 'Egypt, Saudi Arabia',
'United Kingdom, France, Poland, Germany, United States',
'Ireland, Switzerland, United Kingdom, France, United States',
'United Kingdom, South Africa, France',
'Ireland, United Kingdom, France, Germany',
'Russia, United States', 'United Kingdom, United States, France',
'United Kingdom', 'United States, India, United Kingdom', 'Kenya',
'Spain, Argentina', 'India, United Kingdom, France, Qatar',
'Belgium, France', 'Argentina, Chile', 'United States, Thailand',
'Chile, Brazil', 'United States, Colombia',
'Canada, United States, United Kingdom', 'Uruguay', 'Luxembourg',
'United States, Cambodia, Romania', 'Bangladesh',
'Spain, Belgium, United States',
'United Kingdom, United States, Australia',
'Canada, United States, France', 'Portugal, United States',
'Portugal, Spain', 'India, United States',
'United Kingdom, Ireland', 'United Kingdom, Spain, United States',
'Hungary, United States', 'United States, South Korea',
'Canada, United States, Cayman Islands', 'India, France',
'France, Canada', 'Canada, Hungary, United States', 'Norway',
'Canada, United Kingdom, United States',
'United Kingdom, Germany, France, United States',
'Denmark, United States', 'Senegal', 'France, Algeria',
'United Kingdom, Finland, Germany, United States, Australia, Japan, France,
Ireland',
'Philippines, Canada, United Kingdom, United States',
'Ireland, France, Iceland, United States, Mexico, Belgium, United Kingdom,
Hong Kong',
'Singapore', 'Kuwait', 'United States, France, Serbia',
'United States, Italy', 'Spain, Italy',
'United States, Ireland, United Kingdom, India',
'United Kingdom, Singapore', 'Hong Kong, United States',
'United States, Malta, France, United Kingdom',
'United States, China, Canada', 'Canada, United States, Ireland',
'Lebanon, Canada, France', 'Japan, Canada, United States',
'Spain, France, Canada',
'Denmark, Singapore, Canada, United States',
'United States, France, Denmark', 'United States, China, Colombia',
'Spain, Thailand, United States', 'Mexico, Spain',
'Ireland, Luxembourg, Belgium', 'China, United States',
'Canada, Belgium', 'Canada, United Kingdom',
'Lebanon, United Arab Emirates, France, Switzerland, Germany',
'France, Belgium, Italy',
'Lebanon, United States, United Arab Emirates', 'Lebanon, France',
'France, Lebanon', 'France, Lebanon, United Kingdom',
'France, Norway, Lebanon, Belgium',
'Sweden, Czech Republic, United Kingdom, Denmark, Netherlands',
'United States, United Kingdom, India', 'Indonesia, Netherlands',
'Turkey, South Korea', 'Serbia, United States', 'Namibia',
'United Kingdom, Kenya', 'United Kingdom, France, Germany, Spain',
'United Kingdom, France, United States, Belgium, Luxembourg, China, German
y',
'Thailand, United States',

'United States, France, Canada, Belgium', 'United Kingdom, China',
'Germany, China, United Kingdom',
'Australia, New Zealand, United States',
'Hong Kong, Iceland, United States', 'France, Australia, Germany',
'United States, Belgium, Canada, France', 'South Africa, Angola',
'United States, Philippines',
'United States, United Kingdom, Canada, China',
'United States, Canada, United Kingdom', 'Turkey, United States',
'Peru, Germany, Norway', 'Mozambique', 'Brazil, France',
'China, Spain, South Korea, United States', 'Spain, Germany',
'Hong Kong, China', 'France, Belgium, Luxembourg, Cambodia',
'United Kingdom, Australia', 'Belarus',
'Indonesia, United Kingdom',
'Switzerland, France, Belgium, United States', 'Ghana',
'Spain, France, Canada, United States', 'Chile, Italy',
'United Kingdom, Nigeria', 'Chile', 'France, Egypt',
'Egypt, France', 'France, Brazil, Spain, Belgium',
'Egypt, Algeria', 'Canada, South Korea, United States',
'Nigeria, United Kingdom', 'United States, France, Canada',
'Poland, United States',
'United Arab Emirates, Jordan, Lebanon, Saudi Arabia',
'United States, Mexico, Spain, Malta',
'Saudi Arabia, United Arab Emirates', 'Zimbabwe',
'United Kingdom, Germany, United Arab Emirates, New Zealand',
'Romania, United States', 'Canada, Nigeria',
'Saudi Arabia, Netherlands, Germany, Jordan, United Arab Emirates, United States',
'United Kingdom, Spain', 'Finland, France',
'United Kingdom, Germany, United States, France',
'India, United Kingdom, China, Canada, Japan, South Korea, United States',
'Italy, United Kingdom, France', 'United States, Mexico, Colombia',
'Turkey, India', 'Italy, Turkey',
'United Kingdom, United States, Japan',
'France, Belgium, United States',
'Puerto Rico, United States, Colombia', 'Uruguay, Argentina',
'United States, United Kingdom, Japan', 'United States, Argentina',
'United Kingdom, Italy', 'Ireland, United Kingdom',
'United Kingdom, France, Belgium, Canada, United States',
'Netherlands, Germany, Denmark, United Kingdom', 'Hungary',
'Austria, Germany', 'Taiwan, China',
'United Kingdom, United States, Ireland',
'South Korea, United States', 'Brazil, United Kingdom',
'Pakistan, United States', 'Romania, France, Switzerland, Germany',
'Romania, United Kingdom', 'France, Malta, United States',
'Cyprus',
'United Kingdom, France, Belgium, Ireland, United States',
'United States, Norway, Canada', 'Kenya, United States',
'France, South Korea, Japan, United States', 'Taiwan, Malaysia',
'Uruguay, Argentina, Germany, Spain',
'United States, United Kingdom, France, Germany, Japan',
'United States, France, Japan',
'United Kingdom, France, United States',
'Spain, France, United States',
'Indonesia, South Korea, Singapore', 'United States, Spain',
'Netherlands, Germany, Italy, Canada',
'Spain, Germany, Denmark, United States', 'Norway, Sweden',
'South Korea, Canada, United States, China',
'Argentina, Uruguay, Serbia', 'France, Japan',
'Mauritius, South Africa', 'United States, Poland',
'United Kingdom, United States, Germany, Denmark, Belgium, Japan',
'India, Germany', 'India, United Kingdom, Canada, United States',
'Philippines, United States', 'Romania, Bulgaria, Hungary',
'Uruguay, Guatemala', 'France, Senegal, Belgium',
'United Kingdom, Canada', 'Mexico, United States, Spain, Colombia',

'Canada, Norway', 'Singapore, United States',
'Finland, Germany, Belgium', 'United Kingdom, France',
'United States, Chile', 'United Kingdom, Japan, United States',
'Spain, United Kingdom', 'Argentina, United States, Mexico',
'United States, South Korea, Japan', 'Canada, Australia',
'United Kingdom, Hungary, Australia', 'Italy, Belgium',
'United States, United Kingdom, Germany', 'Switzerland',
'Singapore, Malaysia',
'France, Belgium, Luxembourg, Romania, Canada, United States',
'South Africa, Nigeria', 'Spain, France',
'United Kingdom, Hong Kong', 'Pakistan', 'Brazil, United States',
'Denmark, Brazil, France, Portugal, Sweden', 'India, Turkey',
'Malaysia, Singapore, Hong Kong', 'Philippines, Singapore',
'Australia, Canada', 'Taiwan, China, France, United States',
'Germany, Italy', 'Colombia, Peru, United Kingdom',
'Thailand, China, United States', 'Argentina, United States',
'Sweden, United States', 'Uruguay, Spain, Mexico',
'France, Luxembourg, Canada', 'Denmark, Spain', 'Chile, Argentina',
'United Kingdom, Belgium, Sweden', 'Canada, Brazil',
'Italy, France', 'Canada, Germany',
'Pakistan, United Arab Emirates', 'Ghana, United States',
'Mexico, Finland', 'United Arab Emirates, United Kingdom, India',
'Netherlands, Belgium', 'United States, Taiwan',
'Austria, Iraq, United States', 'United Kingdom, Malawi',
'Paraguay, Argentina', 'United Kingdom, Russia, United States',
'India, Pakistan', 'Indonesia, Singapore', 'Spain, Belgium',
'Iceland, Sweden, Belgium', 'Croatia', 'Uruguay, Argentina, Spain',
'United Kingdom, Ireland, United States',
'Canada, Germany, France, United States', 'United Kingdom, Japan',
'Norway, Denmark, Netherlands, Sweden',
'Hong Kong, China, United States', 'Ireland, Canada',
'Italy, Switzerland, France, Germany', 'Mexico, Netherlands',
'United States, Sweden', 'Germany, France, Russia',
'France, Iran, United States', 'United Kingdom, India',
'Russia, Poland, Serbia', 'Spain, Portugal', 'Peru',
'Mexico, Argentina',
'United Kingdom, Canada, United States, Cayman Islands',
'Indonesia, United States',
'United States, Israel, United Kingdom, Canada',
'Norway, Iceland, United States', 'Czech Republic, United States',
'United Kingdom, India, United States',
'United Kingdom, West Germany', 'India, Australia',
'United States, ', 'Belgium, United Kingdom, United States',
'India, Germany, Austria',
'United States, Brazil, South Korea, Mexico, Japan, Germany',
'Spain, Mexico', 'China, Japan', 'Argentina, France',
'China, United States, United Kingdom',
'France, Luxembourg, United States',
'China, United States, Australia', 'Colombia, Mexico',
'United States, Canada, Ireland', 'Chile, Peru',
'Argentina, Italy', 'Canada, Japan, United States',
'United Kingdom, Canada, United States, Germany',
'Italy, Switzerland, Albania, Poland',
'United States, Japan, Canada', 'Cambodia',
'Italy, United States, Argentina',
'Saudi Arabia, Syria, Egypt, Lebanon, Kuwait',
'United States, Canada, Indonesia, United Kingdom, China, Singapore',
'Spain, Colombia',
'United Kingdom, South Africa, Australia, United States',
'Bulgaria', 'Argentina, Brazil, France, Poland, Germany, Denmark',
'United Kingdom, Spain, United States, Germany',
'Philippines, Qatar', 'Netherlands, Belgium, Germany, Jordan',
'United Arab Emirates, United States', 'Norway, Germany, Sweden',
'South Korea, China', 'Georgia', 'Soviet Union, India',

'Australia, United Arab Emirates', 'Canada, Germany, South Africa',
'South Korea, China, United States', 'India, Soviet Union',
'India, Mexico', 'Georgia, Germany, France',
'United Arab Emirates, Romania', 'India, Malaysia',
'Germany, Jordan, Netherlands', 'Turkey, France, Germany, Poland',
'Greece, United States', 'France, United Kingdom, United States',
'Norway, Germany', 'France, Morocco', 'Cambodia, United States',
'United States, Denmark', 'United States, Colombia, Mexico',
'United Kingdom, Italy, Israel, Peru, United States',
'Argentina, Uruguay, Spain, France',
'United Kingdom, France, United States, Belgium',
'France, Canada, China, Cambodia',
'United Kingdom, France, Belgium, United States', 'Chile, France',
'Netherlands, United States', 'France, United Kingdom, India',
'Czech Republic, Slovakia', 'Singapore, France',
'Spain, Switzerland', 'United States, Australia, China',
'South Africa, United States, Germany',
'United States, United Kingdom, Australia',
'Spain, Italy, Argentina', 'Chile, Spain, Argentina, Germany',
'West Germany', 'Austria, Czech Republic', 'Lebanon, Qatar',
'United Kingdom, Jordan, Qatar, Iran',
'France, South Korea, Japan', 'Israel, Germany, France',
'Canada, Japan, Netherlands', 'United States, Hungary',
'France, Germany', 'France, Qatar',
'United Kingdom, Germany, Canada', 'Ireland, South Africa',
'Chile, United States, France', 'Belgium, France, Netherlands',
'United Kingdom, Ukraine, United States',
'Germany, Australia, France, China', 'Norway, United States',
'United States, Bermuda, Ecuador',
'United States, Hungary, Ireland, Canada',
'United Kingdom, Egypt, United States',
'United States, France, United Kingdom', 'Spain, Mexico, France',
'United States, South Africa', 'Hong Kong, China, Singapore',
'South Africa, China, United States', 'Denmark, France, Poland',
'New Zealand, United Kingdom',
'Netherlands, Denmark, South Africa', 'Iran, France',
'United Kingdom, United States, France, Germany',
'Australia, France', 'Ireland, United Kingdom, United States',
'United Kingdom, France, Germany', 'Canada, Luxembourg',
'Brazil, Netherlands, United States, Colombia, Austria, Germany',
'France, Canada, Belgium', 'Canada, France',
'Bulgaria, United States, Spain, Canada', 'Sweden, Netherlands',
'France, United States, Mexico',
'Australia, United Kingdom, United Arab Emirates, Canada',
'Australia, Armenia, Japan, Jordan, Mexico, Mongolia, New Zealand, Philippines, South Africa, Sweden, United States, Uruguay',
'India, Iran', 'France, Belgium, Spain',
'Denmark, Sweden, Israel, United States', 'United States, Iceland',
'United Kingdom, Russia',
'United States, Israel, Italy, South Africa',
'Netherlands, Denmark, France, Germany', 'South Korea, Japan',
'United Kingdom, Pakistan', 'France, New Zealand',
'United Kingdom, Czech Republic, United States, Germany, Bahamas',
'China, Germany, India, United States', 'Germany, Sri Lanka',
'United States, India, Bangladesh',
'United States, Canada, France', 'Brazil, France, Germany',
'Germany, United States, Hong Kong, Singapore',
'France, Germany, Switzerland',
'Germany, France, Luxembourg, United Kingdom, United States',
'United Kingdom, Canada, Italy', 'Czech Republic, France',
'Taiwan, Hong Kong, United States, China', 'Germany, Australia',
'United Kingdom, Poland, United States', 'Denmark, Zimbabwe',
'United Kingdom, South Africa',
'Finland, Sweden, Norway, Latvia, Germany',

'South Africa, United States, New Zealand, Canada',
'United States, Italy, United Kingdom, Liechtenstein',
'Denmark, France, Belgium, Italy, Netherlands, United States, United Kingdom',
'United States, Australia, Mexico',
'United Kingdom, Czech Republic, Germany, United States',
'France, China, Japan, United States',
'United States, South Korea, China', 'Germany, Belgium',
'Pakistan, Norway, United States',
'United States, Canada, Belgium, United Kingdom', 'Venezuela',
'Canada, France, Italy, Morocco, United States',
'Canada, Spain, France', 'United States, Indonesia',
'Spain, France, Italy',
'United Arab Emirates, United States, United Kingdom',
'United Kingdom, Israel, Russia', 'Spain, Cuba',
'United States, Brazil', 'United States, France, Mexico',
'United States, Nicaragua',
'United Kingdom, United States, Spain, Germany, Greece, Canada',
'Italy, Canada, France',
'United Kingdom, Denmark, Canada, Croatia', 'Italy, Germany',
'United States, France, United Kingdom, Japan',
'United States, United Kingdom, Denmark, Sweden',
'United States, United Kingdom, Italy',
'United States, France, Canada, Spain',
'Russia, United States, China', 'United States, Canada, Germany',
'Ireland, United States', 'United States, United Arab Emirates',
'United States, Ireland',
'Ireland, United Kingdom, Italy, United States', 'Poland',
'Slovenia, Croatia, Germany, Czech Republic, Qatar',
'Canada, United Kingdom, Netherlands',
'United States, Spain, Germany', 'India, Japan',
'China, South Korea, United States',
'United Kingdom, France, Belgium',
'Canada, Ireland, United States',
'United Kingdom, United States, Dominican Republic',
'United States, Senegal', 'Germany, United Kingdom, United States',
'South Africa, Germany, Netherlands, France',
'Canada, United States, United Kingdom, France, Luxembourg',
'Ireland, United States, France', 'Germany, United States, Canada',
'United Kingdom, Germany, Canada, United States',
'United States, France, Canada, Lebanon, Qatar',
'Netherlands, Belgium, United Kingdom, United States',
'France, Belgium, China, United States',
'United States, Chile, Israel',
'United Kingdom, Norway, Denmark, Germany, Sweden',
'Norway, Denmark, Sweden', 'China, India, Nepal',
'Colombia, Mexico, United States', 'United Kingdom, South Korea',
'Denmark, China', 'United States, Greece, Brazil',
'South Korea, France',
'United States, Australia, Samoa, United Kingdom',
'Germany, United Kingdom', 'Argentina, Chile, Peru',
'Turkey, Azerbaijan', 'Poland, West Germany',
'Germany, United States, Sweden', 'Canada, Spain',
'United States, Cambodia', 'United States, Greece',
'Norway, United Kingdom, France, Ireland',
'United Kingdom, Poland', 'Israel, Sweden, Germany, Netherlands',
'Switzerland, France', 'Italy, India', 'United States, Botswana',
'Chile, Argentina, France, Spain, United States',
'United States, India, South Korea, China',
'Denmark, Germany, Belgium, United Kingdom, France',
'Denmark, Germany, Belgium, United Kingdom, France, Sweden',
'France, Switzerland, Spain, United States, United Arab Emirates',
'Brazil, India, China, United States',
'Denmark, France, United States, Sweden', 'Australia, Iraq',

'China, Morocco, Hong Kong', 'Canada, United States, Germany',
'United Kingdom, Thailand', 'Venezuela, Colombia',
'Colombia, United States',
'France, Germany, Czech Republic, Belgium',
'Switzerland, Vatican City, Italy, Germany, France',
'Portugal, France, Poland, United States',
'United States, New Zealand, Japan',
'United States, Netherlands, Japan, France', 'India, Switzerland',
'Canada, India', 'United States, Morocco',
'Singapore, Japan, France',
'Canada, Mexico, Germany, South Africa',
'United Kingdom, United States, Canada',
'Germany, France, United States, Canada, United Kingdom',
'United States, Uruguay', 'India, Canada',
'Ireland, Canada, United Kingdom, United States',
'United States, Germany, Australia', 'Australia, France, Ireland',
'Australia, India', 'United States, United Kingdom, Canada, Japan',
'Sweden, United Kingdom, Finland', 'Hong Kong, Taiwan',
'United States, United Kingdom, Spain, South Korea', 'Guatemala',
'Ukraine',
'Italy, South Africa, West Germany, Australia, United States',
'United States, Germany, United Kingdom, Australia',
'Italy, France, Switzerland', 'Canada, France, United States',
'Switzerland, United States', 'Thailand, Canada, United States',
'China, Hong Kong, United States', 'United Kingdom, New Zealand',
'Czech Republic, United Kingdom, France',
'Australia, United Kingdom, Canada', 'Jamaica, United States',
'Australia, United Kingdom, United States, New Zealand, Italy, France',
'France, United States, Canada',
'United Kingdom, France, Canada, Belgium, United States',
'Denmark, United Kingdom, Sweden', 'United States, Hong Kong',
'United States, Kazakhstan',
'Argentina, France, United States, Germany, Qatar',
'United States, Germany, United Kingdom',
'United States, Germany, United Kingdom, Italy',
'United States, New Zealand, United Kingdom',
'Finland, United States', 'Spain, France, Uruguay',
'France, Canada, United States', 'United States, Canada, China',
'Ireland, Canada, Luxembourg, United States, United Kingdom, Philippines, India',
'United States, Czech Republic, United Kingdom', 'Israel, Germany',
'Mexico, France',
'Israel, Germany, Poland, Luxembourg, Belgium, France, United States',
'Austria, United States', 'United Kingdom, Lithuania',
'United States, Greece, United Kingdom',
'United Kingdom, China, United States, India',
'United States, Sweden, Norway',
'United Kingdom, United States, Morocco',
'United States, United Kingdom, Morocco',
'Spain, Canada, United States',
'United States, India, United Arab Emirates',
'United Kingdom, Canada, France, United States',
'India, Germany, France',
'Belgium, Ireland, Netherlands, Germany, Afghanistan',
'France, Canada, Italy, United States, China',
'Ireland, United Kingdom, Greece, France, Netherlands',
'Denmark, Indonesia, Finland, Norway, United Kingdom, Israel, France, United States, Germany, Netherlands',
'New Zealand, United States',
'United States, Australia, South Africa, United Kingdom',
'United States, Germany, Mexico',
'Somalia, Kenya, Sudan, South Africa, United States',
'United States, Canada, Japan, Panama',
'United Kingdom, Spain, Belgium', 'Serbia, South Korea, Slovenia',

```
'Denmark, United Kingdom, South Africa, Sweden, Belgium',
'Germany, Canada, United States',
'Ireland, Canada, United States, United Kingdom',
'New Zealand, United Kingdom, Australia',
'United Kingdom, Australia, Canada, United States',
'Germany, United States, Italy', 'United States, Venezuela',
'United Kingdom, Canada, Japan',
'United Kingdom, United States, Czech Republic',
'United Kingdom, China, United States',
'United Kingdom, Brazil, Germany',
'United Kingdom, Namibia, South Africa, Zimbabwe, United States',
'Canada, United States, India, United Kingdom',
'Switzerland, United Kingdom, United States',
'United Kingdom, India, Sweden',
'United States, Brazil, India, Uganda, China',
'Peru, United States, United Kingdom',
'Germany, United States, United Kingdom, Canada',
'Canada, India, Thailand, United States, United Arab Emirates',
'United States, East Germany, West Germany',
'France, Netherlands, South Africa, Finland',
'Egypt, Austria, United States', 'Russia, Spain',
'Croatia, Slovenia, Serbia, Montenegro', 'Japan, Canada',
'United States, France, South Korea, Indonesia',
'United Arab Emirates, Jordan'], dtype=object)
```

Observation: There are movies and TV shows that released in multiple countries. But this is not accurate, since we can see that there are movies and TV shows released in multiple countries which are clubbed together separated by commas.

In [18]: # Write code to get unique countries

In []:

In [19]: df['release_year'].sort_values().unique()

Out[19]: array([1925, 1942, 1943, 1944, 1945, 1946, 1947, 1954, 1955, 1956, 1958,
 1959, 1960, 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969,
 1970, 1971, 1972, 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980,
 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990, 1991,
 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002,
 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013,
 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021], dtype=int64)

In [20]: df['release_year'].max()

Out[20]: 2021

In [21]: df['release_year'].min()

Out[21]: 1925

Observation: Netflix has content released between 1925 and 2021

In [22]: df['rating'].unique()

Out[22]: array(['PG-13', 'TV-MA', 'PG', 'TV-14', 'TV-PG', 'TV-Y', 'TV-Y7', 'R',
 'TV-G', 'G', 'NC-17', '74 min', '84 min', '66 min', 'NR', nan,
 'TV-Y7-FV', 'UR'], dtype=object)

In [23]: df[(df['rating'] == '74 min') | (df['rating'] == '84 min') | (df['rating'] == '66 m...')]

Out[23]:		show_id	type	title	director	cast	country	date_added	release_year	rating	duration
5541	s5542	Movie	Louis C.K. 2017	Louis C.K.	Louis C.K.	Louis C.K.	United States	April 4, 2017	2017	74 min	N
5794	s5795	Movie	Louis C.K.: Hilarious	Louis C.K.	Louis C.K.	Louis C.K.	United States	September 16, 2016	2010	84 min	N
5813	s5814	Movie	Louis C.K.: Live at the Comedy Store	Louis C.K.	Louis C.K.	Louis C.K.	United States	August 15, 2016	2015	66 min	N



Observation: These are the ratings given on Netflix

'PG-13', 'TV-MA', 'PG', 'TV-14', 'TV-PG', 'TV-Y', 'TV-Y7', 'R', 'TV-G', 'G', 'NC-17',
'74 min', '84 min', '66 min', 'NR', 'TV-Y7-FV', 'UR'

out of which we can clearly see that 74 min, 84 min, 66 min are durations
which by mistake has been entered in rating column

In []:

In []:

In [24]: `df['duration'].unique()`

```
Out[24]: array(['90 min', '2 Seasons', '1 Season', '91 min', '125 min',
   '9 Seasons', '104 min', '127 min', '4 Seasons', '67 min', '94 min',
   '5 Seasons', '161 min', '61 min', '166 min', '147 min', '103 min',
   '97 min', '106 min', '111 min', '3 Seasons', '110 min', '105 min',
   '96 min', '124 min', '116 min', '98 min', '23 min', '115 min',
   '122 min', '99 min', '88 min', '100 min', '6 Seasons', '102 min',
   '93 min', '95 min', '85 min', '83 min', '113 min', '13 min',
   '182 min', '48 min', '145 min', '87 min', '92 min', '80 min',
   '117 min', '128 min', '119 min', '143 min', '114 min', '118 min',
   '108 min', '63 min', '121 min', '142 min', '154 min', '120 min',
   '82 min', '109 min', '101 min', '86 min', '229 min', '76 min',
   '89 min', '156 min', '112 min', '107 min', '129 min', '135 min',
   '136 min', '165 min', '150 min', '133 min', '70 min', '84 min',
   '140 min', '78 min', '7 Seasons', '64 min', '59 min', '139 min',
   '69 min', '148 min', '189 min', '141 min', '130 min', '138 min',
   '81 min', '132 min', '10 Seasons', '123 min', '65 min', '68 min',
   '66 min', '62 min', '74 min', '131 min', '39 min', '46 min',
   '38 min', '8 Seasons', '17 Seasons', '126 min', '155 min',
   '159 min', '137 min', '12 min', '273 min', '36 min', '34 min',
   '77 min', '60 min', '49 min', '58 min', '72 min', '204 min',
   '212 min', '25 min', '73 min', '29 min', '47 min', '32 min',
   '35 min', '71 min', '149 min', '33 min', '15 min', '54 min',
   '224 min', '162 min', '37 min', '75 min', '79 min', '55 min',
   '158 min', '164 min', '173 min', '181 min', '185 min', '21 min',
   '24 min', '51 min', '151 min', '42 min', '22 min', '134 min',
   '177 min', '13 Seasons', '52 min', '14 min', '53 min', '8 min',
   '57 min', '28 min', '50 min', '9 min', '26 min', '45 min',
   '171 min', '27 min', '44 min', '146 min', '20 min', '157 min',
   '17 min', '203 min', '41 min', '30 min', '194 min', '15 Seasons',
   '233 min', '237 min', '230 min', '195 min', '253 min', '152 min',
   '190 min', '160 min', '208 min', '180 min', '144 min', '5 min',
   '174 min', '170 min', '192 min', '209 min', '187 min', '172 min',
   '16 min', '186 min', '11 min', '193 min', '176 min', '56 min',
   '169 min', '40 min', '10 min', '3 min', '168 min', '312 min',
   '153 min', '214 min', '31 min', '163 min', '19 min', '12 Seasons',
   nan, '179 min', '11 Seasons', '43 min', '200 min', '196 min',
   '167 min', '178 min', '228 min', '18 min', '205 min', '201 min',
   '191 min'], dtype=object)
```

Observation: as can be seen different content has different duration. So they need to be standardized and converted to minutes of overall content duration.

Statistical Summary

```
In [25]: df.head(3)
```

Out[25]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Nan	United States	September 25, 2021	2020	PG-13	90
1	s2	TV Show	Blood & Water	Nan	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	See
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	Nan	September 24, 2021	2021	TV-MA	1 S

In [26]: # max, min, mean, std

In [27]: df.describe(include=object)

Out[27]:

	show_id	type	title	director	cast	country	date_added	rating	duration
count	8807	8807	8807	6173	7982	7976	8797	8803	8804
unique	8807	2	8807	4528	7692	748	1767	17	220
top	s1	Movie	Dick Johnson Is Dead	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	TV-MA	1 Season
freq	1	6131	1	19	19	2818	109	3207	1793

In [28]: df[df['date_added']=='January 1, 2020'].count()

Out[28]:

show_id	109
type	109
title	109
director	98
cast	107
country	103
date_added	109
release_year	109
rating	109
duration	109
listed_in	109
description	109
dtype: int64	

In [29]: df[df['rating']=='TV-MA'].count()

```
Out[29]: show_id      3207  
          type        3207  
          title       3207  
          director    2117  
          cast         2881  
          country      2931  
          date_added   3205  
          release_year 3207  
          rating        3207  
          duration      3207  
          listed_in     3207  
          description    3207  
          dtype: int64
```

Observations:

- out of 8807 rows movies occur 6131 times
- January 1, 2020 maximum number of content was added on the same date.
- Out of 8803 ratings 3207 got a rating of 'TV-MA'

In []:

In [30]: df.describe()

Out[30]: release_year

	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 []:

In []:

Missing Value & Outlier check

Missing value detection

In [31]: 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
```

Observation: as can be seen columns **director, cast, country, date_added, rating and duration** have missing values since the non-null count is lower than the number of rows (8807 rows)

```
In [32]: temp = pd.read_csv('netflix.csv')
temp.head()
```

Out[32]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Nan	United States	September 25, 2021	2020	PG-13	50 min
1	s2	TV Show	Blood & Water	Nan	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	See notes
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	Nan	September 24, 2021	2021	TV-MA	15 min
3	s4	TV Show	Jailbirds	Nan	Nan	Nan	September 24, 2021	2021	TV-MA	15 min
4	s5	TV Show	New Orleans Factory	Nan	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	See notes

In [33]: `temp[temp['duration'].isna()]`

Out[33]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
5541	s5542	Movie	Louis C.K. 2017	Louis C.K.	Louis C.K.	United States	April 4, 2017	2017	74 min	N
5794	s5795	Movie	Louis C.K.: Hilarious	Louis C.K.	Louis C.K.	United States	September 16, 2016	2010	84 min	N
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	N

In [34]: `temp[temp['rating'].isna()].count()`

```
Out[34]: show_id      4  
          type        4  
          title       4  
          director    1  
          cast        4  
          country     3  
          date_added  4  
          release_year 4  
          rating      0  
          duration    4  
          listed_in   4  
          description  4  
          dtype: int64
```

```
In [35]: def swap(x):  
    x['duration'] = x['rating']  
    x['rating'] = np.nan  
    return x  
  
temp[temp['duration'].isna()] = temp[temp['duration'].isna()].apply(swap, axis=1)
```

```
In [36]: temp[temp['duration'].isna()]
```

```
Out[36]: show_id  type  title  director  cast  country  date_added  release_year  rating  duration  listed_i
```



Observation: missing values in duration is resolved

```
In [37]: temp[temp['rating'].isna()].count()
```

```
Out[37]: show_id      7  
          type        7  
          title       7  
          director    4  
          cast        7  
          country     6  
          date_added  7  
          release_year 7  
          rating      0  
          duration    7  
          listed_in   7  
          description  7  
          dtype: int64
```

```
In [38]: temp[temp['rating'].isna()]
```

Out[38]:	show_id	type	title	director	cast	country	date_added	release_year	ra
5541	s5542	Movie	Louis C.K. 2017	Louis C.K.	Louis C.K.	United States	April 4, 2017	2017	
5794	s5795	Movie	Louis C.K.: Hilarious	Louis C.K.	Louis C.K.	United States	September 16, 2016	2010	
5813	s5814	Movie	Louis C.K.: Live at the Comedy Store	Louis C.K.	Louis C.K.	United States	August 15, 2016	2015	
5989	s5990	Movie	13TH: A Conversation with Oprah Winfrey & Ava ...	NaN	Oprah Winfrey, Ava DuVernay	NaN	January 26, 2017	2017	
6827	s6828	TV Show	Gargantia on the Verdurous Planet	NaN	Kaito Ishikawa, Hisako Kanemoto, Ai Kayano, Ka...	Japan	December 1, 2016	2013	
7312	s7313	TV Show	Little Lunch	NaN	Flynn Curry, Olivia Deeble, Madison Lu, Oisín ...	Australia	February 1, 2018	2015	
7537	s7538	Movie	My Honor Was Loyalty	Alessandro Pepe	Leone Frisa, Paolo Vaccarino, Francesco Miglio...	Italy	March 1, 2017	2015	

In [39]: `temp['rating'].value_counts()`

Out[39]:

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

Name: rating, dtype: int64

Since TV-MA is the most common rating. SO we will fill up the missing ratings with TV-MA

```
In [40]: temp['rating'].fillna('TV-MA', inplace=True)
```

```
In [41]: temp[temp['rating'].isna()].count()
```

```
Out[41]: 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
```

Observation: Missing ratings are now filled in and there are no further missing rating as seen above.

```
In [42]: temp[temp['show_id']=='s5990']
```

	show_id	type	title	director	cast	country	date_added	release_year	rating
5989	s5990	Movie	13TH: A Conversation with Oprah Winfrey & Ava ...	NaN	Oprah Winfrey, Ava DuVernay	NaN	January 26, 2017	2017	TV-MA

```
In [43]: temp.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            8807 non-null   object 
 9   duration          8807 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
```

Observation: date_added column has some missing values which we can fill in with a approximate value of 'January 1,' + release_year

```
In [44]: temp['date_added'].fillna('January 1, ' + str(temp['release_year']), inplace=True)
```

```
In [45]: temp.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        8807 non-null    object  
 7   release_year      8807 non-null    int64  
 8   rating             8807 non-null    object  
 9   duration           8807 non-null    object  
 10  listed_in          8807 non-null    object  
 11  description         8807 non-null    object  
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

```
In [46]: temp.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        8807 non-null    object  
 7   release_year      8807 non-null    int64  
 8   rating             8807 non-null    object  
 9   duration           8807 non-null    object  
 10  listed_in          8807 non-null    object  
 11  description         8807 non-null    object  
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

```
In [47]: temp['country'].value_counts()
```

```
Out[47]: 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
```

```
In [48]: temp[temp['country'].isna()]
```

Out[48]:		show_id	type	title	director	cast	country	date_added	release_year	ratio
	2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	T M
	3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	T M
	5	s6	TV Show	Midnight Mass	Mike Flanagan	Kate Siegel, Zach Gilford, Hamish Linklater, H...	NaN	September 24, 2021	2021	T M
	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	
	10	s11	TV Show	Vendetta: Truth, Lies and The Mafia	NaN	NaN	NaN	September 24, 2021	2021	T M

	8718	s8719	Movie	Westside vs. the World	Michael Fahey	Ron Perlman, Louie Simmons	NaN	August 9, 2019	2019	T M
	8759	s8760	Movie	World's Weirdest Homes	Storm Theunissen	Charlie Luxton	NaN	February 1, 2019	2015	TV-
	8783	s8784	TV Show	Yoko	NaN	Eileen Stevens, Alyson Leigh Rosenfeld, Sarah ...	NaN	June 23, 2018	2016	TV
	8785	s8786	TV Show	YOM	NaN	Sairaj, Devyani Dagaonkar, Ketan Singh, Mayur ...	NaN	June 7, 2018	2016	TV-

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV-	
In [49]:	<code>temp[temp['director']=='Louis C.K.']['country'].value_counts()</code>									
Out[49]:	United States 3 Name: country, dtype: int64									
In [50]:	<code>temp[temp['director']=='Louis C.K.']['country'].value_counts().keys()[0]</code>									
Out[50]:	'United States'									
In [51]:	<code>temp.head()</code>									
Out[51]:	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	See more
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 S
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 S
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	See more

In [52]: `temp['country'].describe()[2]`

```
Out[52]: 'United States'
```

```
In [53]: temp['country'].fillna(temp['country'].describe()[2], inplace=True)
```

```
In [54]: def getCountry(x):
    if len(x["country"].value_counts()) == 0:
        pass
    else:
        x['country_top'] = x["country"].value_counts().keys()[0] # gives the country with max count
    return x
```

```
In [55]: #temp = temp.groupby(["director"]).apply(getCountry)
```

```
In [56]: temp.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	50 min
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	See more
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA	1 hr 15 min
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	United States	September 24, 2021	2021	TV-MA	1 hr 15 min
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	See more

```
In [57]: temp.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     8807 non-null   object  
 6   date_added  8807 non-null   object  
 7   release_year 8807 non-null   int64  
 8   rating      8807 non-null   object  
 9   duration    8807 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
```

As can be seen country column now has no missing values

```
In [58]: temp['director'].fillna(temp['director'].describe()[2], inplace=True)
```

```
In [59]: temp['cast'].fillna(temp['cast'].describe()[2], inplace=True)
```

```
In [60]: temp.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    8807 non-null   object  
 4   cast         8807 non-null   object  
 5   country     8807 non-null   object  
 6   date_added  8807 non-null   object  
 7   release_year 8807 non-null   int64  
 8   rating      8807 non-null   object  
 9   duration    8807 non-null   object  
 10  listed_in   8807 non-null   object  
 11  description 8807 non-null   object  
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

```
In [61]: df[df['duration'].isna()] = df[df['duration'].isna()].apply(swap, axis=1)
df['rating'].fillna('TV-MA',inplace=True)
df['date_added'].fillna('January 1, '+ str(df['release_year']),inplace=True)
df['country'].fillna(df['country'].describe()[2], inplace=True)
df['director'].fillna(df['director'].describe()[2], inplace=True)
df['cast'].fillna(df['cast'].describe()[2], inplace=True)
```

```
In [62]: 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    8807 non-null   object  
 4   cast         8807 non-null   object  
 5   country     8807 non-null   object  
 6   date_added  8807 non-null   object  
 7   release_year 8807 non-null   int64  
 8   rating      8807 non-null   object  
 9   duration    8807 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
```

Observation: as can be seen there are no more missing values in any of the columns.

In []:

unnesting of the data in columns like Cast, Director, Country, listed_in

In [63]: `temp.head()`

Out[63]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

In [64]: `temp['director_list'] = temp['director'].str.split(", ")`

In [65]: `temp.head()`

Out[65]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

In [66]: `temp['cast_list'] = temp['cast'].str.split(", ")`

In [67]: `temp.head()`

Out[67]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thabang Molaba, Dillon Windvogel, Natasha Thahane, Arno Greeff, Xolile Tshabalala, Getmore Sithole, Cindy Mahlangu, Ryle De Morny, Greteli Fincham, Sello Maake Ka-Ncube, Odwa Gwanya, Mekaila Mathys, Sandi Schultz, Duane Williams, Shamilla Miller, Patrick Mofokeng	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

In [68]: `list(temp[temp['show_id']=='s2']['cast_list'])# just to show the parsed list`

Out[68]:

```
[['Ama Qamata',
 'Khosi Ngema',
 'Gail Mabalane',
 'Thabang Molaba',
 'Dillon Windvogel',
 'Natasha Thahane',
 'Arno Greeff',
 'Xolile Tshabalala',
 'Getmore Sithole',
 'Cindy Mahlangu',
 'Ryle De Morny',
 'Greteli Fincham',
 'Sello Maake Ka-Ncube',
 'Odwa Gwanya',
 'Mekaila Mathys',
 'Sandi Schultz',
 'Duane Williams',
 'Shamilla Miller',
 'Patrick Mofokeng']]
```

In [69]: `temp['country_list'] = temp['country'].str.split(", ")`

In [70]: `temp['category'] = temp['listed_in'].str.split(", ")`

In [71]: `temp.head()`

Out[71]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

In [72]: `temp=temp.explode('category').reset_index().drop('index',axis=1)`

In [73]: `temp.head()`

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
3	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
4	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA

In [74]: `temp=temp.explode('country_list').reset_index().drop('index',axis=1)`
`temp=temp.explode('cast_list').reset_index().drop('index',axis=1)`
`temp=temp.explode('director_list').reset_index().drop('index',axis=1)`

In [75]: `temp.head()`

Out[75]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13	90 min
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	50 min
2	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	50 min
3	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	50 min
4	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	50 min

In [76]: `temp.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 201991 entries, 0 to 201990
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   show_id          201991 non-null   object 
 1   type              201991 non-null   object 
 2   title             201991 non-null   object 
 3   director          201991 non-null   object 
 4   cast               201991 non-null   object 
 5   country            201991 non-null   object 
 6   date_added        201991 non-null   object 
 7   release_year      201991 non-null   int64  
 8   rating             201991 non-null   object 
 9   duration           201991 non-null   object 
 10  listed_in          201991 non-null   object 
 11  description         201991 non-null   object 
 12  director_list      201991 non-null   object 
 13  cast_list           201991 non-null   object 
 14  country_list        201991 non-null   object 
 15  category            201991 non-null   object 
dtypes: int64(1), object(15)
memory usage: 24.7+ MB
```

Observation: As can be seen there are no more missing values and the nested columns are unnested to new columns. but the older columns director, cast , country and listed_in needs to be dropped.

```
In [77]: temp = temp.drop('cast', axis=1)
temp = temp.drop('country', axis=1)
temp = temp.drop('listed_in', axis=1)
```

```
In [78]: temp.head()
```

		show_id	type	title	director	date_added	release_year	rating	duration	description	dir
0	s1	Movie	Dick Johnson Is Dead	Rajiv Chilaka	Kirsten Johnson	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...	
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
2	s2	TV Show	Blood & Water	Rajiv Chilaka	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
3	s2	TV Show	Blood & Water	Rajiv Chilaka	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
4	s2	TV Show	Blood & Water	Rajiv Chilaka	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	

```
In [79]: temp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 201991 entries, 0 to 201990
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   show_id          201991 non-null   object  
 1   type              201991 non-null   object  
 2   title             201991 non-null   object  
 3   director          201991 non-null   object  
 4   date_added        201991 non-null   object  
 5   release_year      201991 non-null   int64  
 6   rating            201991 non-null   object  
 7   duration          201991 non-null   object  
 8   description        201991 non-null   object  
 9   director_list     201991 non-null   object  
 10  cast_list         201991 non-null   object  
 11  country_list      201991 non-null   object  
 12  category          201991 non-null   object  
dtypes: int64(1), object(12)
memory usage: 20.0+ MB
```

In []:

Observation: cleaned data is ready for further analysis

Non-Graphical Analysis: Value counts and unique attributes

Assumptions

- There is no data that directly represent the user response , for example like, dislikes, 5 star rating or revenue. So the natural trends in the data will be treated as positive / negative affirmation from users
- among TV shows 1 Season is considered to have 10 episodes and each episode is of 30 min each. which means each season is of 300 min each
- Will be using 2 kinds of DataFrames df for original data with missing values imputed and temp for exploded data and missing values imputed

Comparison of tv shows vs. movies

In [80]: `df.head()`

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

[◀](#) [▶](#)

In [81]: `temp.head(10)`

	show_id	type	title	director	date_added	release_year	rating	duration	description	dir
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...	
1	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
2	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
3	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
4	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
5	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
6	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
7	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
8	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
9	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a	

show_id	type	title	director	date_added	release_year	rating	duration	description	dir
								party, a	Cape Town

```
In [82]: df['type'].value_counts()
```

```
Out[82]: Movie      6131
          TV Show    2676
          Name: type, dtype: int64
```

```
In [83]: temp[temp['type']=='Movie']['category'].unique()
```

```
Out[83]: array(['Documentaries', 'Children & Family Movies', 'Dramas',
       'Independent Movies', 'International Movies', 'Comedies',
       'Thrillers', 'Romantic Movies', 'Music & Musicals',
       'Horror Movies', 'Sci-Fi & Fantasy', 'Action & Adventure',
       'Classic Movies', 'Anime Features', 'Sports Movies', 'Cult Movies',
       'Faith & Spirituality', 'LGBTQ Movies', 'Stand-Up Comedy',
       'Movies'], dtype=object)
```

```
In [84]: temp[temp['type']=='Movie']['category'].nunique()
```

```
Out[84]: 20
```

```
In [85]: temp[temp['type']=='Movie']["country_list"].nunique()
```

```
Out[85]: 122
```

```
In [86]: temp[temp['type']=='Movie']["rating"].nunique()
```

```
Out[86]: 14
```

```
In [87]: temp[temp['type']=='TV Show']['category'].unique()
```

```
Out[87]: array(['International TV Shows', 'TV Dramas', 'TV Mysteries',
       'Crime TV Shows', 'TV Action & Adventure', 'Docuseries',
       'Reality TV', 'Romantic TV Shows', 'TV Comedies', 'TV Horror',
       'British TV Shows', 'Spanish-Language TV Shows', 'TV Thrillers',
       "Kids' TV", 'TV Sci-Fi & Fantasy', 'Anime Series',
       'Korean TV Shows', 'Science & Nature TV', 'Teen TV Shows',
       'TV Shows', 'Stand-Up Comedy & Talk Shows', 'Classic & Cult TV'],
       dtype=object)
```

```
In [88]: temp[temp['type']=='TV Show']['category'].nunique()
```

```
Out[88]: 22
```

```
In [89]: temp[temp['type']=='TV Show']["country_list"].nunique()
```

```
Out[89]: 66
```

```
In [90]: temp[temp['type']=='TV Show']["rating"].nunique()
```

```
Out[90]: 9
```

Observations:

- As can be seen there are more movies on Netflix than TV shows

- The number and variation of categories under both movies and TV Shows are comparable
- Movies are available almost in twice the number of countries as that of TV Shows.
- So we can conclude that Movies have a considerably larger outreach as that of TV Shows
- Movies are available across 14 different ratings whereas TV Shows are available only in 9 ratings

In []:

What is the best time to launch a TV show

In [91]: `df[df['type'] == 'TV Show']['release_year'].value_counts()`

```
Out[91]:
```

2020	436
2019	397
2018	380
2021	315
2017	265
2016	244
2015	162
2014	88
2012	64
2013	63
2010	40
2011	40
2009	34
2008	23
2006	14
2007	14
2005	13
2003	10
2004	9
1999	7
2002	7
2001	5
1993	4
2000	4
1997	4
1998	4
1990	3
1996	3
1992	3
1995	2
1994	2
1988	2
1986	2
1989	1
1967	1
1985	1
1946	1
1981	1
1972	1
1979	1
1977	1
1991	1
1974	1
1925	1
1945	1
1963	1

Name: release_year, dtype: int64

```
In [92]: df[df['type'] == 'TV Show']['date_added'].value_counts().head(20)
```

```
Out[92]: July 6, 2021      31
          January 1, 2016    23
          August 1, 2017    22
          June 19, 2021     21
          July 1, 2017      19
          February 1, 2019   17
          November 1, 2019   14
          December 15, 2016  14
          November 15, 2019  13
          July 12, 2019      13
          October 1, 2017    13
          May 15, 2020       13
          November 1, 2020   12
          January 1, 2020    12
          March 31, 2017     12
          October 1, 2016    12
          April 14, 2021     12
          September 1, 2016  11
          December 1, 2020   11
          October 1, 2020    11
          Name: date_added, dtype: int64
```

In []:

In []:

Does Netflix has more focus on TV Shows than movies in recent years

```
In [93]: df[df['type'] == 'TV Show']['release_year'].value_counts()
```

```
Out[93]: 2020    436  
2019    397  
2018    380  
2021    315  
2017    265  
2016    244  
2015    162  
2014     88  
2012     64  
2013     63  
2010     40  
2011     40  
2009     34  
2008     23  
2006     14  
2007     14  
2005     13  
2003     10  
2004      9  
1999      7  
2002      7  
2001      5  
1993      4  
2000      4  
1997      4  
1998      4  
1990      3  
1996      3  
1992      3  
1995      2  
1994      2  
1988      2  
1986      2  
1989      1  
1967      1  
1985      1  
1946      1  
1981      1  
1972      1  
1979      1  
1977      1  
1991      1  
1974      1  
1925      1  
1945      1  
1963      1  
Name: release_year, dtype: int64
```

```
In [94]: df[df['type'] == 'Movie']['release_year'].value_counts()
```

```
Out[94]: 2017    767  
2018    767  
2016    658  
2019    633  
2020    517  
...  
1966      1  
1961      1  
1946      1  
1963      1  
1947      1  
Name: release_year, Length: 73, dtype: int64
```

Observation: As can be seen there is more number of movies released in the recent years than TV Shows. SO we cannot conclude that Netflix has more focus on TV Shows than movies in recent years

In []:

Understanding what content is available in different countries

In [95]: `temp.groupby("country_list")["category"].unique()`

Out[95]:

```
country_list
Afghanistan      [International TV Shows, TV Dramas, Dramas, In...
Albania          [Documentaries, International Movies]
Algeria          [Dramas, International Movies]
Angola           [Dramas, Independent Movies, International Mov...
Vatican City    [Documentaries, Faith & Spirituality, Internat...
Venezuela        [Dramas, International Movies, Thrillers, Docu...
Vietnam          [Dramas, International Movies, Romantic Movies...
West Germany     [International TV Shows, TV Comedies, Document...
Zimbabwe         [Comedies, International Movies, Romantic Movi...
Name: category, Length: 127, dtype: object
```

In [96]: `temp.groupby("country_list")["rating"].unique()`

Out[96]:

```
country_list
Afghanistan      [TV-MA, TV-14]
Albania          [TV-MA]
Algeria          [TV-MA]
Angola           [TV-MA]
Vatican City    [PG]
Venezuela        [TV-MA, TV-14, NR]
Vietnam          [TV-14, TV-MA, TV-G]
West Germany     [TV-MA, TV-14, G]
Zimbabwe         [TV-G, NR, TV-MA]
Name: rating, Length: 127, dtype: object
```

In [97]: `temp.groupby(["country_list", "release_year"])["title"].nunique().head(20)`

```
Out[97]:   country_list  release_year
              2014          1
              2021          1
Afghanistan  2016          1
Albania      2018          1
Algeria       1979          1
              2014          1
              2020          1
Angola        2020          1
Argentina     1985          2
              1993          1
              1998          1
              2004          2
              2006          3
              2007          1
              2013          1
              2014          1
              2015          7
              2016         11
              2017         14
              2018         17
Name: title, dtype: int64
```

```
In [88]: temp.groupby(["country_list","type"])["title"].nunique()
```

```
Out[88]:   country_list  type
              Movie        1
              TV Show      1
Afghanistan  Movie        1
Albania      Movie        1
Algeria       Movie        3
              ..
Venezuela    Movie        4
Vietnam      Movie        7
West Germany Movie        3
              TV Show      2
Zimbabwe     Movie        3
Name: title, Length: 188, dtype: int64
```

Observation: looking at the above analysis we can see what type, category, rating content available in different countries.

Visual Analysis - Univariate, Bivariate after pre-processing of the data

For continuous variable(s): Distplot, countplot, histogram for univariate analysis

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

Out[100]: `show_id type title director cast country date_added release_year rating`

0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	David Attenborough	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	Rajiv Chilaka	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	United States	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	Rajiv Chilaka	David Attenborough	United States	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	Rajiv Chilaka	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

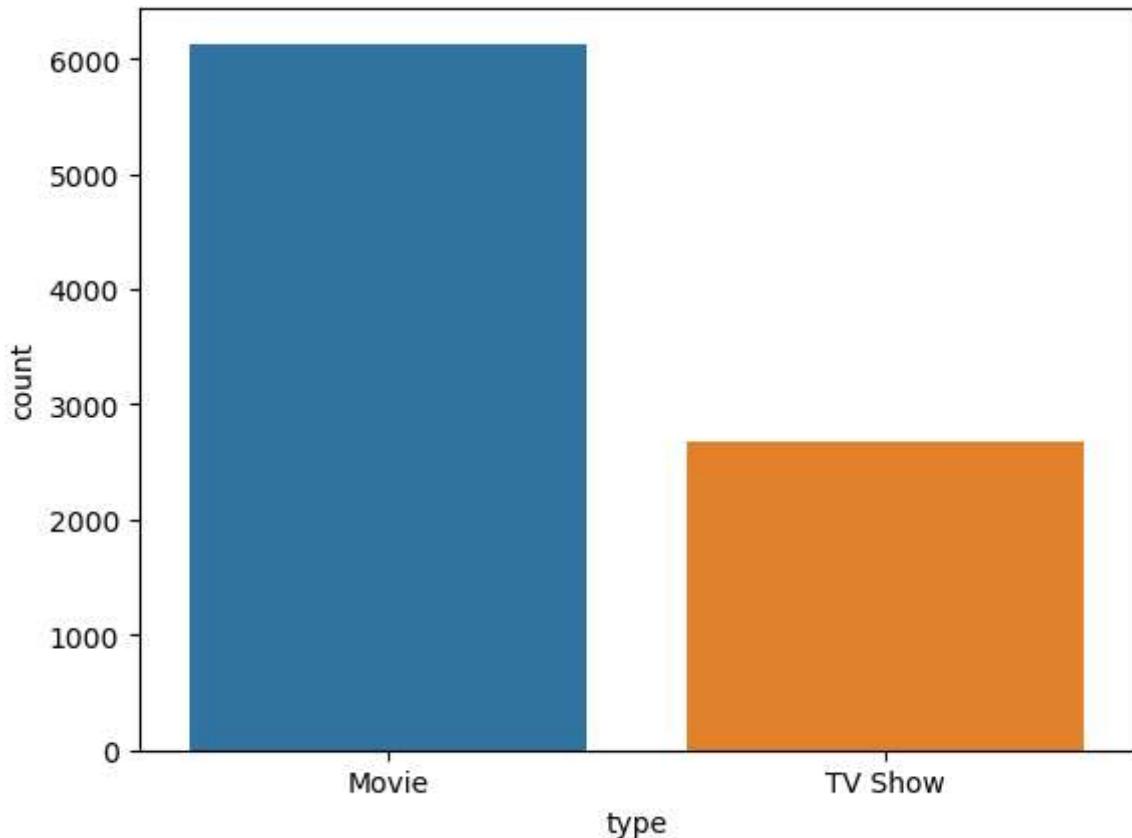
In [101... `temp.head()`

	show_id	type	title	director	date_added	release_year	rating	duration	description	dir
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	September 25, 2021	2020	PG-13	90 min	As her father nears the end of his life, filmm...	
1	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
2	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
3	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	
4	s2	TV Show	Blood & Water	Rajiv Chilaka	September 24, 2021	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	



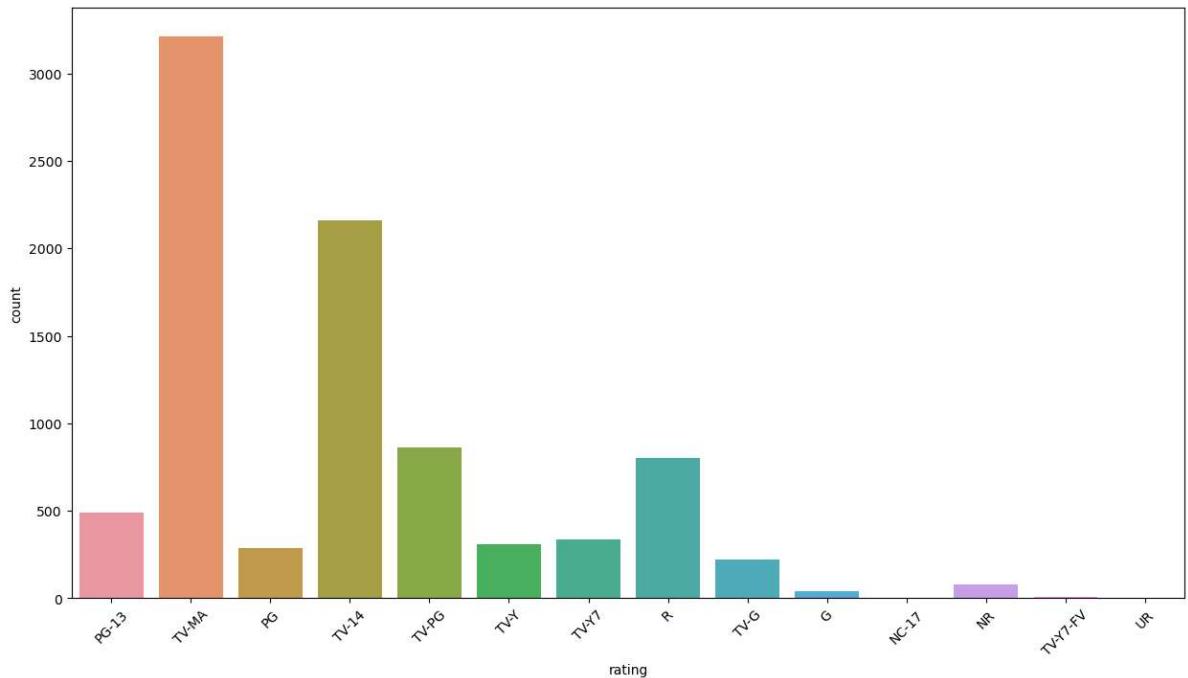
```
In [9]: import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [99]: sns.countplot(x ='type', data = df)
plt.show()
```



Observation: As can be seen there is almost twice the number of movies than TV shows on Netflix

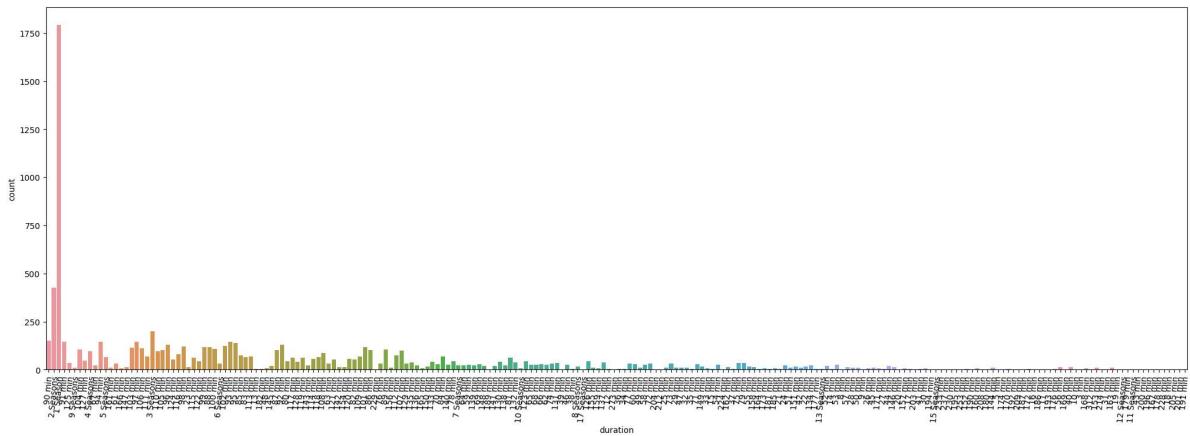
```
In [106]: plt.figure(figsize=(15,8))
plt.xticks(rotation=45)
sns.countplot(x ='rating', data = df)
plt.show()
```



Observation: Most of the content is rated as TV-MA, TV-14, TV-PG, R

```
In [110]: plt.figure(figsize=(25,8))
plt.xticks(rotation=85)
```

```
sns.countplot(x = 'duration', data = df)
plt.show()
```

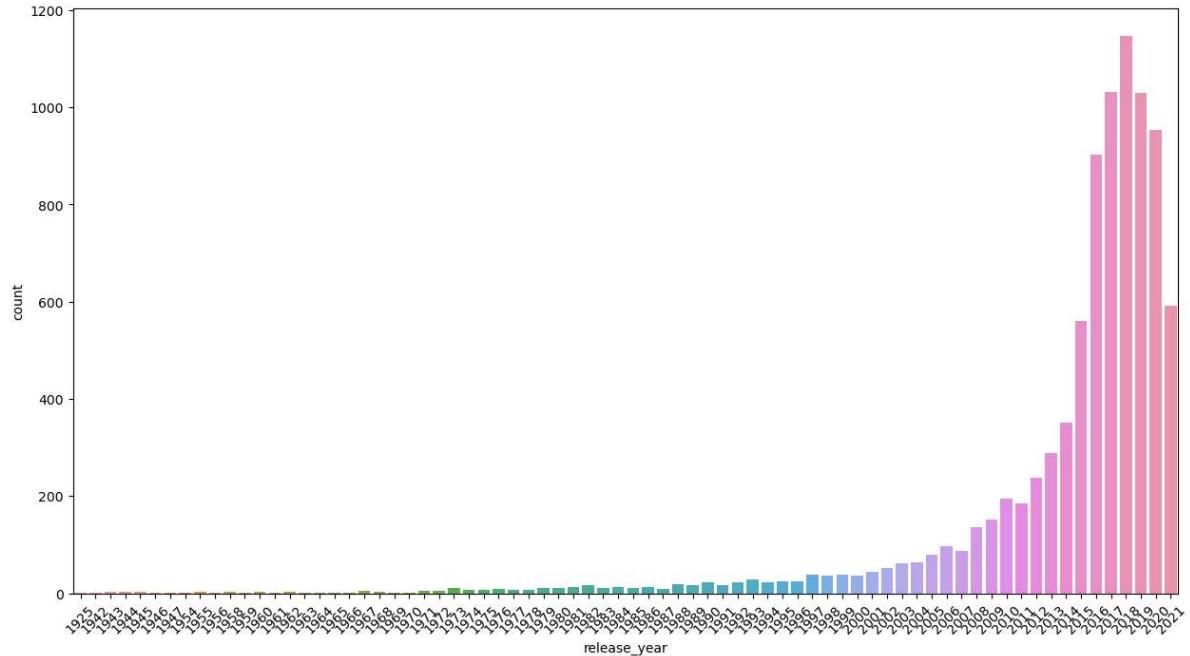


Observation: As can be seen on the left side most of the content is of length 90 min, 2 Season and 1 Season

In []:

How has the number of movies released per year changed over the last 20-30 years?

```
plt.figure(figsize=(15,8))
plt.xticks(rotation=45)
sns.countplot(x ='release_year', data = df)
plt.show()
```



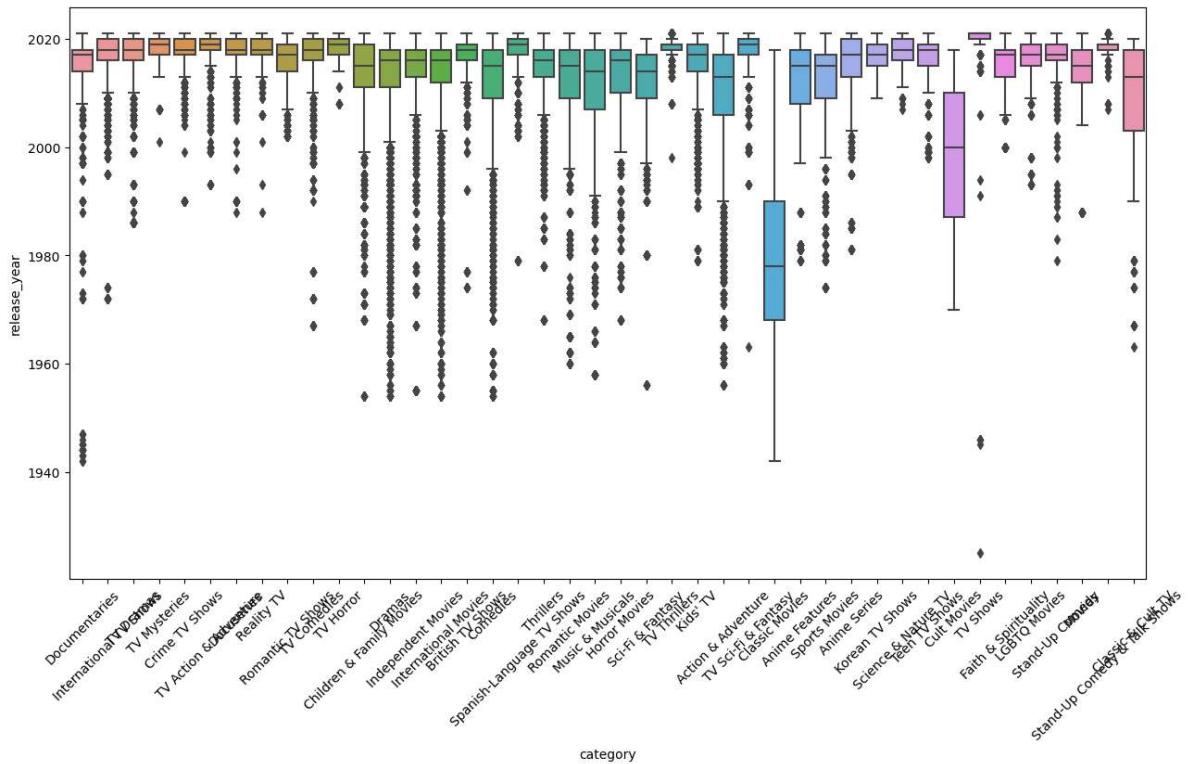
Observation: as can be seen from the right side of the plot maximum number of content was published in 2018 after which there is a steady decline each year

In []:

For categorical variable(s): Boxplot

```
In [117]: plt.figure(figsize=(15,8))
plt.xticks(rotation=45)
sns.boxplot(x = 'category', y= "release_year", data = temp)
```

Out[117]: <AxesSubplot:xlabel='category', ylabel='release_year'>



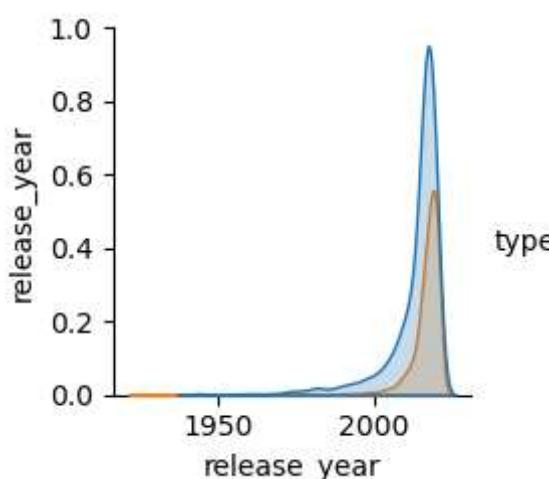
Observation: As can be seen Classic and cult movies are not that common in the 2020s, whereas new categories such as Standup comedy, Anime and TV-scifi are emerging

In []:

For correlation: Heatmaps, Pairplots

```
In [118]: sns.pairplot(df, hue='type')
```

Out[118]: <seaborn.axisgrid.PairGrid at 0x1c83ae83d30>



Observation: The amount of content being consumed is at an all time high in the recent 2020s for both TV shows and Movies

Insights based on Non-Graphical and Visual Analysis

All the insights and observations are mentioned next to each cell after both graphical and non-graphical analysis is done. Marked as "Observation"

Business Insights

All the insights and observations are mentioned next to each cell after both graphical and non-graphical analysis is done. Marked as "Observation"

In []:

Recommendations

Recommendations:

1. More TV shows needs to be released
2. TV Shows should be available in more genres / categories like Movies
3. Some countries have only 1 or 2 categories of content available so more Movies and TV-shows under different categories should be release in those countries, since that would appeal a wider audience.
4. Some countries have content only available under 1 or 2 ratings, which might make it age restrictive. So content from a wider range of ratings needs to be release in those under-served countries.
5. More attention needs to be paid to the new and emerging genres / categories
6. there has been a dip in the number of movies and TV-Shows released each year since 2019. So more movies and TV-shows needs to be release each year

In []:

