Netflix Analysis

Netlix, Inc. is an American technology and media services provider and production company headquartered in **Los Gatos, California**. Netlix was founded in 1997 by **Reed Hastings** and **Marc Randolph** in Scotts Valley, California. The company's primary business is its subscription-based streaming service, which offers online streaming of a library of films and television series, including those produced in-house.

In [85]:
```
# Lets start by importing some basic libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


import warnings
warnings.filterwarnings('ignore')
```

In [86]:
```
# Read the csv file into a variable

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

---

1. Defining Problem Statement and Analysing basic metrics

Problem Statement : From the given data find out which type of shows to produce and how to grow the business.

Basic Metric : We need to set a metric which best captures how to measure if the business is growing or falling. Since we don't have a factual field, we use dimensions to create a metric. In this analysis I am using ***most repeated*** as the metric for evaluation.

---

2. 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 *(Pre-processing also included)*

In [87]:
```
# Lets see how the data looks like

df.head() # first 5 rows in the dataset
```

Out[87]:

	show_id	type	title	director	cast	country	date_added	release_year	rating
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13
1	s2	TV Show	Blood & Water	NaN	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...	NaN	September 24, 2021	2021	TV-MA
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA

In [88]:
```
# Lets look at the shape of the data

df.shape
# There are 8807 rows and 12 columns in this dataset
```

Out[88]:
```
(8807, 12)
```

In [89]:
```
# Lets look at the datatypes

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
```

All the columns except release_year is of object datatype, where release_year is of integer datatype

In [90]:
```
df.describe() # default behaviour of describe is only for numeric datatype
```

Out[90]:

	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

The data contains movies/TVshows which were released from 1925 to 2021

In [91]:
```
df.describe(include='O') # include='O' will describe the object datatype co
```

Out[91]:

	show_id	type	title	director	cast	country	date_added	rating	durati
count	8807	8807	8807	6173	7982	7976	8797	8803	88
unique	8807	2	8807	4528	7692	748	1767	17	2
top	s6715	Movie	America's Next Top Model	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	TV-MA	Seas
freq	1	6131	1	19	19	2818	109	3207	17

- We can see there are 2 netflix types, with 8807 titles which were directed by 4528 directors.
- Movies/TVshows of 748 countries are listed in this dataset.
- We can also see that 2818 movies/tvshows were made in United States.

In [92]:
```
# Lets check the percent of missing values

df.isnull().sum()/df.count() * 100
```

Out[92]:
```
show_id          0.000000
type             0.000000
title            0.000000
director        42.669691
cast            10.335755
country         10.418756
date_added       0.113675
release_year     0.000000
rating           0.045439
duration         0.034075
listed_in        0.000000
description      0.000000
dtype: float64
```

Director field has the most missing values - 42.6%

In [93]:
```
# Lets check what types of shows are in the dataset

df.type.unique()
```

Out[93]:
```
array(['Movie', 'TV Show'], dtype=object)
```

In [94]:
```
# How is these types distributed

df['type'].value_counts(normalize=True) * 100
```

Out[94]:
```
Movie      69.615079
TV Show    30.384921
Name: type, dtype: float64
```

> **69.6%** of shows are Movies and **30.3%** of shows are TV Shows.
>
> *We can see Most of the content in Netflix are **Movies***

Pre-processing

- We can see that the following columns are nested in the dataframe
  1. director
  2. cast
  3. country
  4. listed_in

In [95]:
```
# Function to split the values at comma

def split_nested(x):
    return str(x).split(',')
```

Un-nesting Director column

In [96]:
```
# Split the values in director column and convert it to a list
dir_split = df['director'].apply(split_nested).tolist()

# Create a dataframe with the output and give the title as index
dir_df = pd.DataFrame(dir_split, index=df['title'])

# Stack them together to avoid extra unnecessary columns
dir_df = dir_df.stack()

# Reset the title index as column
dir_df = pd.DataFrame(dir_df.reset_index())

# Rename Director column
dir_df.rename(columns={0:'Director'}, inplace=True)

# Drop the level_1 column as it is not necessary
dir_df.drop(['level_1'], axis=1, inplace=True)

# Lets look at the new dataframe
dir_df.head(20)
```

Out[96]:

	title	Director
0	Dick Johnson Is Dead	Kirsten Johnson
1	Blood & Water	nan
2	Ganglands	Julien Leclercq
3	Jailbirds New Orleans	nan
4	Kota Factory	nan
5	Midnight Mass	Mike Flanagan
6	My Little Pony: A New Generation	Robert Cullen
7	My Little Pony: A New Generation	José Luis Ucha
8	Sankofa	Haile Gerima
9	The Great British Baking Show	Andy Devonshire
10	The Starling	Theodore Melfi
11	Vendetta: Truth, Lies and The Mafia	nan
12	Bangkok Breaking	Kongkiat Komesiri
13	Je Suis Karl	Christian Schwochow
14	Confessions of an Invisible Girl	Bruno Garotti
15	Crime Stories: India Detectives	nan
16	Dear White People	nan
17	Europe's Most Dangerous Man: Otto Skorzeny in ...	Pedro de Echave García
18	Europe's Most Dangerous Man: Otto Skorzeny in ...	Pablo Azorín Williams
19	Falsa identidad	nan

Un-nesting Cast column

In [97]:
```
cast_split = df['cast'].apply(split_nested).tolist()
cast_df = pd.DataFrame(cast_split, index=df['title'])
cast_df = cast_df.stack()
cast_df = pd.DataFrame(cast_df.reset_index())
cast_df.rename(columns={0:'Cast'}, inplace=True)
cast_df.drop(['level_1'], axis=1, inplace=True)
cast_df.head(10)
```

Out[97]:

	title	Cast
0	Dick Johnson Is Dead	nan
1	Blood & Water	Ama Qamata
2	Blood & Water	Khosi Ngema
3	Blood & Water	Gail Mabalane
4	Blood & Water	Thabang Molaba
5	Blood & Water	Dillon Windvogel
6	Blood & Water	Natasha Thahane
7	Blood & Water	Arno Greeff
8	Blood & Water	Xolile Tshabalala
9	Blood & Water	Getmore Sithole

Un-nesting listed_in column

In [98]:
```
listed_in_split = df['listed_in'].apply(split_nested).tolist()
listed_in_df = pd.DataFrame(listed_in_split, index=df['title'])
listed_in_df = listed_in_df.stack()
listed_in_df = pd.DataFrame(listed_in_df.reset_index())
listed_in_df.rename(columns={0:'Genre'}, inplace=True)
listed_in_df.drop(['level_1'], axis=1, inplace=True)
listed_in_df.head(10)
```

Out[98]:

	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
5	Ganglands	International TV Shows
6	Ganglands	TV Action & Adventure
7	Jailbirds New Orleans	Docuseries
8	Jailbirds New Orleans	Reality TV
9	Kota Factory	International TV Shows

Un-nesting countries

In [99]:
```
country_split = df['country'].apply(split_nested).tolist()
country_df = pd.DataFrame(country_split, index=df['title'])
country_df = country_df.stack()
country_df = pd.DataFrame(country_df.reset_index())
country_df.rename(columns={0:'Country'}, inplace=True)
country_df.drop(['level_1'], axis=1, inplace=True)
country_df.head(10)
```

Out[99]:

	title	Country
0	Dick Johnson Is Dead	United States
1	Blood & Water	South Africa
2	Ganglands	nan
3	Jailbirds New Orleans	nan
4	Kota Factory	India
5	Midnight Mass	nan
6	My Little Pony: A New Generation	nan
7	Sankofa	United States
8	Sankofa	Ghana
9	Sankofa	Burkina Faso

Merging all the data together

In [100]:
```
# We can use inner join to merge all the data together

df_new = dir_df.merge(cast_df,
             on=['title'],
             how='inner').merge(listed_in_df,
                                on=['title'],
                                how='inner').merge(country_df,
                                                   on=['title'],
                                                   how='inner')
df_new.head()
```

Out[100]:

	title	Director	Cast	Genre	Country
0	Dick Johnson Is Dead	Kirsten Johnson	nan	Documentaries	United States
1	Blood & Water	nan	Ama Qamata	International TV Shows	South Africa
2	Blood & Water	nan	Ama Qamata	TV Dramas	South Africa
3	Blood & Water	nan	Ama Qamata	TV Mysteries	South Africa
4	Blood & Water	nan	Khosi Ngema	International TV Shows	South Africa

In [101]:
```
df_new.shape
```

Out[101]:
```
(202065, 5)
```

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

How many TV shows and Movies are there in the dataset?

In [102]:
```
df.type.value_counts()
```

Out[102]:
```
Movie      6131
TV Show    2676
Name: type, dtype: int64
```

> There are **6131 Movies** and **2676 TV shows** available in the dataset

How many types of ratings are there and which one is the most common one?

In [103]:
```
df['rating'].value_counts()
```

Out[103]:
```
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
84 min         1
66 min         1
74 min         1
Name: rating, dtype: int64
```

> More than **3000** shows on Netflix are focused towards **Matured Adults**

Which countries produced the shows and which country produced the most shows?

In [104]:
```
# Total number of unique countries

df_new.Country.unique()
```

Out[104]:
```
array(['United States', 'South Africa', 'nan', 'India', ' Ghana',
       ' Burkina Faso', ' United Kingdom', ' Germany', ' Ethiopia',
       'United Kingdom', 'Germany', ' Czech Republic', 'Mexico', 'Turkey',
       'Australia', ' India', ' France', 'Finland', 'China', ' Canada',
       ' United States', ' Japan', 'Nigeria', 'Japan', 'Spain', 'France',
       'Belgium', 'South Korea', ' Singapore', ' Australia', ' Mexico',
       ' Italy', ' Romania', 'Argentina', ' Venezuela', ' Hong Kong',
       'Russia', 'Canada', 'Hong Kong', ' China', 'Italy', '',
       ' South Korea', 'Ireland', ' Nepal', 'New Zealand', ' Brazil',
       ' Greece', 'Jordan', 'Colombia', 'Switzerland', 'Israel', 'Brazil',
       ' Spain', 'Taiwan', ' Nigeria', 'Bulgaria', ' Algeria', 'Poland',
       ' Israel', 'Saudi Arabia', 'Thailand', 'Indonesia', 'Egypt',
       ' Denmark', ' Switzerland', 'Kuwait', ' Netherlands', ' Belgium',
       'Malaysia', ' New Zealand', 'Vietnam', ' Hungary', 'Sweden',
       'Lebanon', 'Romania', ' Syria', 'Philippines', 'Iceland',
       'Denmark', ' Indonesia', ' United Arab Emirates',
       'United Arab Emirates', ' Colombia', 'Netherlands', ' Bulgaria',
       'Norway', 'Syria', ' Lebanon', ' Qatar', 'Mauritius',
       ' South Africa', 'Austria', ' Russia', 'Czech Republic', ' Taiwan',
       'Cameroon', ' Palestine', 'Uruguay', ' Saudi Arabia', ' Poland',
       'Kenya', ' Argentina', ' Chile', ' Thailand', 'Chile',
       'Luxembourg', ' Cambodia', 'Bangladesh', 'Portugal', ' Ireland',
       'Hungary', ' Cayman Islands', 'Senegal', ' Finland', ' Iceland',
       'Singapore', ' Serbia', ' Malta', ' Luxembourg', ' Norway',
       'Serbia', 'Namibia', ' Kenya', ' Angola', ' Philippines', 'Peru',
       'Mozambique', 'Belarus', 'Ghana', ' Egypt', ' Jordan', 'Zimbabwe',
       ' Turkey', 'Puerto Rico', 'Pakistan', 'Cyprus', ' Malaysia',
       ' Sweden', ' Uruguay', ' Guatemala', ' Senegal', ' Portugal',
       ' Peru', ' Iraq', ' Malawi', 'Paraguay', ' Pakistan', 'Croatia',
       ' Iran', ' West Germany', ' Austria', ' Albania', 'Cambodia',
       ' Kuwait', 'Georgia', 'Soviet Union', ' Soviet Union', 'Greece',
       ' Morocco', ' Slovakia', 'West Germany', ' Ukraine', ' Bermuda',
       ' Ecuador', 'Iran', ' Armenia', ' Mongolia', ' Bahamas',
       ' Sri Lanka', ' Bangladesh', ' Zimbabwe', ' Latvia',
       ' Liechtenstein', 'Venezuela', ' Cuba', ' Nicaragua', ' Croatia',
       'Slovenia', ' Dominican Republic', ' Samoa', ' Azerbaijan',
       ' Botswana', ' Vatican City', 'Guatemala', 'Ukraine', 'Jamaica',
       ' Kazakhstan', ' Lithuania', ' Afghanistan', 'Somalia', ' Sudan',
       ' Panama', ' Slovenia', ' Namibia', ' Uganda', ' East Germany',
       ' Montenegro'], dtype=object)
```

In [105]:
```
df_new.Country.value_counts()
```

Out[105]:
```
United States     49868
India             22139
nan               11897
United Kingdom     9733
 United States     9482
                   ...
 Mongolia             2
 Ukraine              2
 Kazakhstan           1
 Nicaragua            1
 Uganda               1
Name: Country, Length: 198, dtype: int64
```

> **United States** produced the most number of shows which is streaming in Netflix

Which Genres are most popular & least popular in Netflix?

In [106]:
```
df_new.Genre.value_counts().head(5)
```

Out[106]:
```
 International Movies    27141
Dramas                   19657
Comedies                 13894
Action & Adventure       12216
 Dramas                  10149
Name: Genre, dtype: int64
```

In [107]:
```
df_new.Genre.value_counts(ascending=True).head()
```

Out[107]:
```
Sports Movies            3
LGBTQ Movies             5
TV Sci-Fi & Fantasy      7
Romantic Movies         20
 Stand-Up Comedy        24
Name: Genre, dtype: int64
```

> Most popular Genre in Netflix is **International Movies**
>
> Least popular Genre in Netflix is **Sports Movies**

Which director produced the most number of shows in Netflix?

In [108]:
```
df_new.Director.value_counts().head()
```

Out[108]:
```
nan                   50643
Martin Scorsese         419
Youssef Chahine         409
Cathy Garcia-Molina     356
Steven Spielberg        355
Name: Director, dtype: int64
```

> **Martin Scorsese** is attributed to most number of shows in Netflix

In [ ]:
```
```

Pre-processing(cont.)

In [109]:
```
# Lets join the main dataset with the un-nested dataset on title column

df_all = df.merge(df_new, on=['title'], how='inner')
df_all.drop(['director', 'cast', 'country', 'listed_in'],axis=1, inplace=Tr
df_all.head(2)
```

Out[109]:

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

In [110]:
```
df_all.shape
```

Out[110]:
```
(202065, 12)
```

In [111]:
```
df_all.drop_duplicates(inplace=True)
```

In [112]:
```
df_all.shape
```

Out[112]:
```
(202058, 12)
```

In [113]:
```
date_added_split = df_all['date_added'].str.strip().apply(lambda x: str(x).
date_added_df = pd.DataFrame(date_added_split, index=df_all['title']).reset
date_added_df.rename(columns={
    0 : 'date_added_month',
    1 : 'date_added_day',
    2 : 'date_added_year'
}, inplace=True)
date_added_df['date_added_day'] = date_added_df.date_added_day.str[:-1]
date_added_df.head()
```

Out[113]:

	title	date_added_month	date_added_day	date_added_year
0	Dick Johnson Is Dead	September	25	2021
1	Blood & Water	September	24	2021
2	Blood & Water	September	24	2021
3	Blood & Water	September	24	2021
4	Blood & Water	September	24	2021

In [114]:
```
df_all = df_all.merge(date_added_df, on=['title'], how='inner')
df_all.drop('date_added', axis=1, inplace=True)
df_all.head()
```

Out[114]:

	show_id	type	title	release_year	rating	duration	description	Director	Cast	
0	s1	Movie	Dick Johnson Is Dead	2020	PG-13	90 min	As her father nears the end of his life, filmm...	Kirsten Johnson	nan	Dc
1	s2	TV Show	Blood & Water	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	nan	Ama Qamata	
2	s2	TV Show	Blood & Water	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	nan	Ama Qamata	
3	s2	TV Show	Blood & Water	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	nan	Ama Qamata	
4	s2	TV Show	Blood & Water	2021	TV-MA	2 Seasons	After crossing paths at a party, a Cape Town t...	nan	Ama Qamata	

In [115]:
```
df_all.shape
```

Out[115]:
```
(10956360, 14)
```

In [116]:
```
df_all.drop_duplicates(inplace=True)
```

In [117]:
```
df_all.shape
```

Out[117]:
```
(202058, 14)
```

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

In [118]:
```
# Lets split the data into TV and Movies based on type


tv = df_all[df_all['type']=='TV Show']
movie = df_all[df_all['type']=='Movie']
```

In [119]:
```
movie.head()
```

Out[119]:

	show_id	type	title	release_year	rating	duration	description	Director	Ca
0	s1	Movie	Dick Johnson Is Dead	2020	PG-13	90 min	As her father nears the end of his life, filmm...	Kirsten Johnson	n
6863	s7	Movie	My Little Pony: A New Generation	2021	PG	91 min	Equestria's divided. But a bright-eyed hero be...	Robert Cullen	Vanes Hudge
6883	s7	Movie	My Little Pony: A New Generation	2021	PG	91 min	Equestria's divided. But a bright-eyed hero be...	Robert Cullen	Kim Gle
6903	s7	Movie	My Little Pony: A New Generation	2021	PG	91 min	Equestria's divided. But a bright-eyed hero be...	Robert Cullen	Jam Marsd
6923	s7	Movie	My Little Pony: A New Generation	2021	PG	91 min	Equestria's divided. But a bright-eyed hero be...	Robert Cullen	Sc Cars

In [120]:
```
tv.type.value_counts()
```

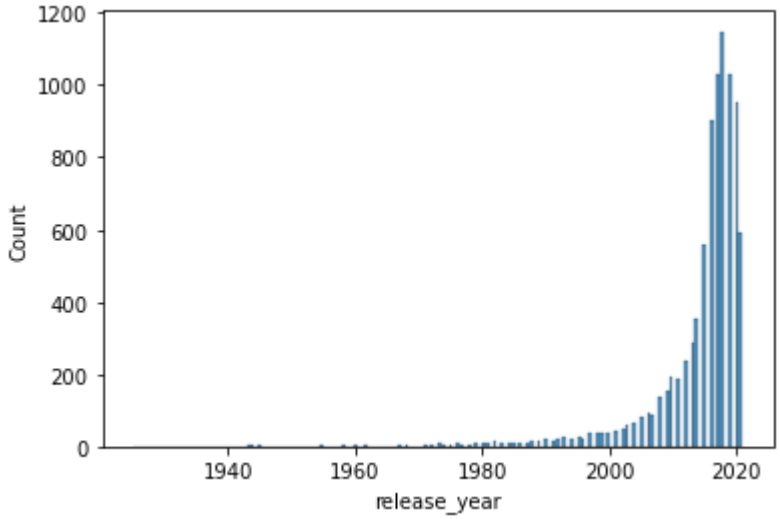
Out[120]:
```
TV Show    56148
Name: type, dtype: int64
```

In [121]:
```
movie.type.value_counts()
```

Out[121]:
```
Movie    145910
Name: type, dtype: int64
```

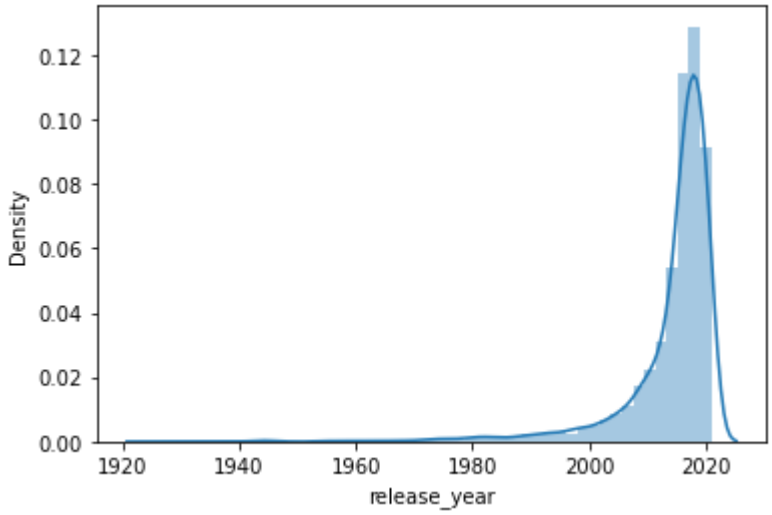
In [122]:
```
sns.histplot(df['release_year'])
```

Out[122]:
```
<AxesSubplot:xlabel='release_year', ylabel='Count'>
```



In [123]:
```
sns.distplot(df['release_year'])
```

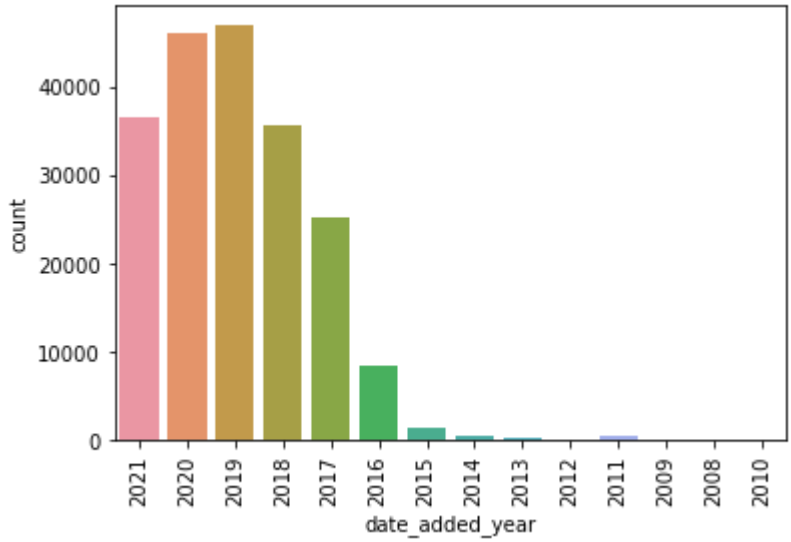
Out[123]:
```
<AxesSubplot:xlabel='release_year', ylabel='Density'>
```



Most number of shows in Netflix are released between year **2000 & 2020**

In [124]:
```
sns.countplot(df_all['date_added_year'])
plt.xticks(rotation=90)
```

Out[124]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13]),
 [Text(0, 0, '2021'),
  Text(1, 0, '2020'),
  Text(2, 0, '2019'),
  Text(3, 0, '2018'),
  Text(4, 0, '2017'),
  Text(5, 0, '2016'),
  Text(6, 0, '2015'),
  Text(7, 0, '2014'),
  Text(8, 0, '2013'),
  Text(9, 0, '2012'),
  Text(10, 0, '2011'),
  Text(11, 0, '2009'),
  Text(12, 0, '2008'),
  Text(13, 0, '2010')])
```

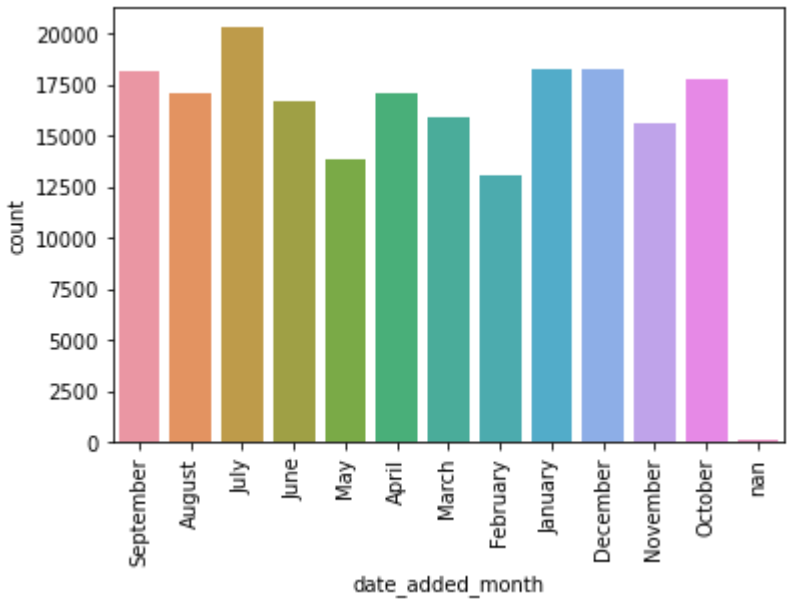


Most shows are added to Netflix in the year **2020**

Very minimal shows are added to Netflix before 2015

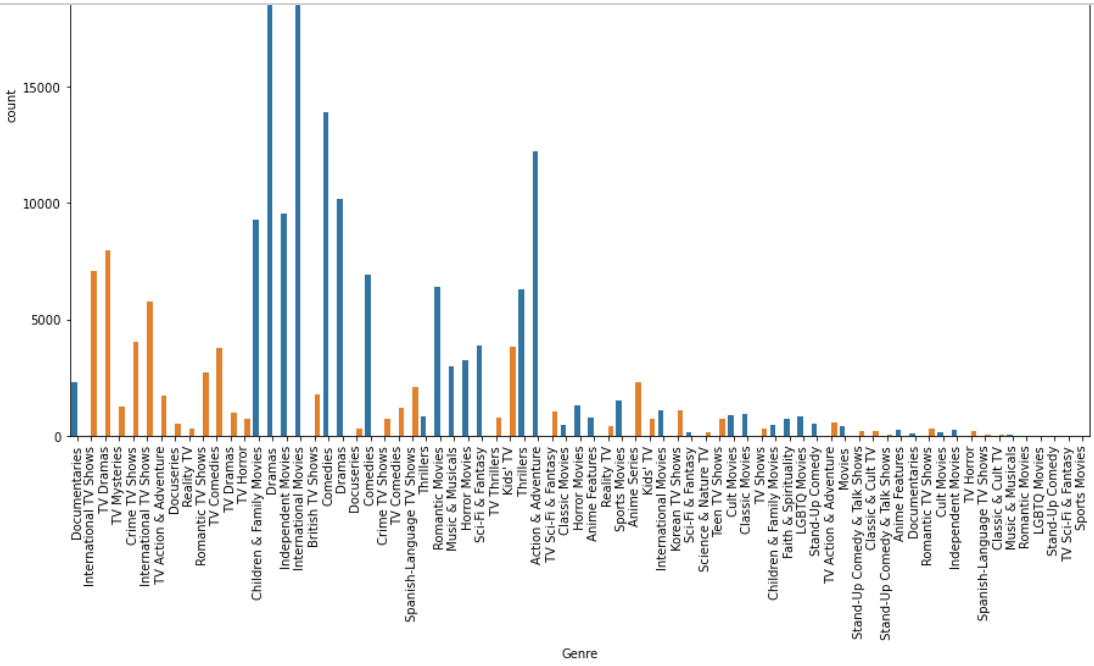
In [125]:
```
sns.countplot(df_all['date_added_month'])
plt.xticks(rotation=90)
```

Out[125]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12]),
 [Text(0, 0, 'September'),
  Text(1, 0, 'August'),
  Text(2, 0, 'July'),
  Text(3, 0, 'June'),
  Text(4, 0, 'May'),
  Text(5, 0, 'April'),
  Text(6, 0, 'March'),
  Text(7, 0, 'February'),
  Text(8, 0, 'January'),
  Text(9, 0, 'December'),
  Text(10, 0, 'November'),
  Text(11, 0, 'October'),
  Text(12, 0, 'nan')])
```



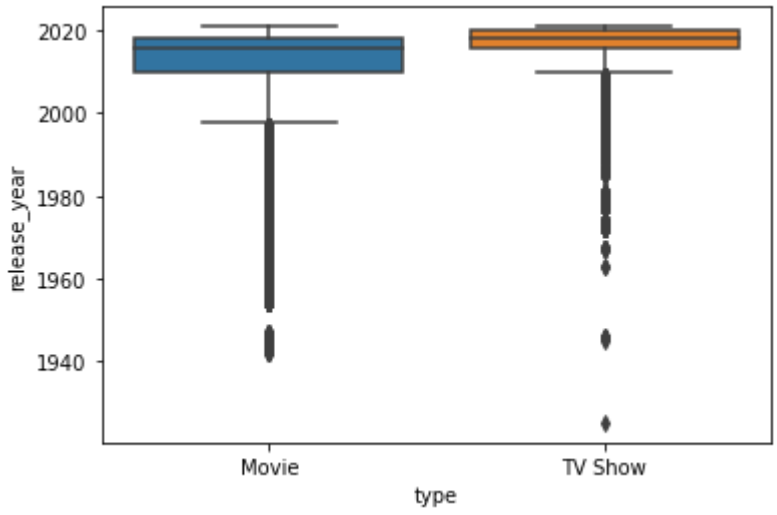
There is almost equal distribution of movies added throught the year with February being the lowest

In [126]:
```
plt.figure(figsize=(15,10))
sns.countplot(df_all['Genre'], hue=df_all['type'])
plt.xticks(rotation=90)
```



In [127]:
```
sns.boxplot(df_all['type'], df_all['release_year'])
```

Out[127]: <AxesSubplot:xlabel='type', ylabel='release_year'>



Most Tv shows and Movies are released recently

5. Missing Value & Outlier check (Treatment optional)

In [128]:
```
df_all.replace({'nan':np.nan}, inplace=True)
```

In [129]:
```
df_all.isnull().sum()
```

Out[129]:
```
show_id               0
type                  0
title                 0
release_year          0
rating               67
duration              3
description           0
Director          50643
Cast               2149
Genre                 0
Country           11897
date_added_month    158
date_added_day      158
date_added_year     158
dtype: int64
```

In [130]:
```
# Treating missing values

df_all['rating'] = df_all['rating'].fillna('TV-MA')
```

In [131]:
```
df_all['duration'].dropna(inplace=True)
```

In [132]:
```
df_all['Director'] = df_all['Director'].fillna('Not Available')
df_all['Cast'] = df_all['Cast'].fillna('Not Available')
df_all['Country'] = df_all['Country'].fillna('Not Available')
```

In [133]:
```
df_all.isnull().sum()
```

Out[133]:
```
show_id             0
type                0
title               0
release_year        0
rating              0
duration            3
description         0
Director            0
Cast                0
Genre               0
Country             0
date_added_month    158
date_added_day      158
date_added_year     158
dtype: int64
```

In [134]:
```
df_all.dropna(inplace=True)
```

In [135]:
```
df_all.isnull().sum()
```

Out[135]:
```
show_id             0
type                0
title               0
release_year        0
rating              0
duration            0
description         0
Director            0
Cast                0
Genre               0
Country             0
date_added_month    0
date_added_day      0
date_added_year     0
dtype: int64
```

In [136]:
```
df_all.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 201897 entries, 0 to 10956336
Data columns (total 14 columns):
 #   Column            Non-Null Count   Dtype 
---  ------            --------------   ----- 
 0   show_id           201897 non-null  object
 1   type              201897 non-null  object
 2   title             201897 non-null  object
 3   release_year      201897 non-null  int64 
 4   rating            201897 non-null  object
 5   duration          201897 non-null  object
 6   description       201897 non-null  object
 7   Director          201897 non-null  object
 8   Cast              201897 non-null  object
 9   Genre             201897 non-null  object
 10  Country           201897 non-null  object
 11  date_added_month  201897 non-null  object
 12  date_added_day    201897 non-null  object
 13  date_added_year   201897 non-null  object
dtypes: int64(1), object(13)
memory usage: 23.1+ MB
```

6. Insights based on Non-Graphical and Visual Analysis

All the columns except release_year is of object datatype, where release_year is of integer datatype

The data contains movies/TVshows which were released from 1925 to 2021

- We can see there are 2 netflix types, with 8807 titles which were directed by 4528 directors.
- Movies/TVshows of 748 countries are listed in this dataset.
- We can also see that 2818 movies/tvshows were made in United States.

Director field has the most missing values - 42.6%

69.6% of shows are Movies and **30.3%** of shows are TV Shows.

*We can see Most of the content in Netflix are **Movies***

There are **6131 Movies** and **2676 TV shows** available in the dataset

More than **3000** shows on Netflix are focused towards **Matured Adults**

United States produced the most number of shows which is streaming in Netflix

Most popular Genre in Netflix is **International Movies**

Least popular Genre in Netflix is **Sports Movies**

Martin Scorsese is attributed to most number of shows in Netflix

Most number of shows in Netflix are released between year **2000 & 2020**

Most shows are added to Netflix in the year **2020**

Very minimal shows are added to Netflix before 2015

There is almost equal distribution of movies added throught the year with February being the lowest

7. Business Insights

I. **Quantity Analysis**

- A. We looked at the stuff on Netflix, and it turns out they have more movies than TV shows. Seems like people like movies >more.

II. **Content Addition Trends**

- A. Netflix adds most of its stuff in July, and then a bit in December. They're probably doing it on purpose to keep us >entertained all year round.

III. **Genre Correlations**

- A. We found out that certain types of shows and movies often go hand in hand. Like, if you like dramas, you might also >like international shows. It's all about what people enjoy watching together.

IV. **Movie Characteristics**

- A. Movies used to be longer back in the 1960s, but now they're usually around 100 minutes. Things have changed over time.

V. **TV Show Insights**

- A. Most TV shows on Netflix only have one season. People seem to prefer shorter series.

VI. **Common Themes**

- A. We noticed words like love, life, family, and adventure pop up a lot in titles and descriptions. These are the themes >that Netflix likes to stick with.

VII. **Rating Distribution**

- A. People's ratings on Netflix have changed over the years. It tells us what people like and don't like.

VIII. **Data-Driven Insights**

- A. Our journey of looking at the data showed us how powerful information can be in understanding what's going on with >Netflix. It's useful for both people who watch and those who make the shows and movies.

IX. **Continued Relevance**

- A. As Netflix keeps growing, it's important to understand what people like to watch. Knowing the patterns helps everyone >navigate this big world of Netflix.

X. **Conclusion**

- A. We hope you had fun reading about Netflix! Go explore the cool stories on there. Let the numbers guide your binge->watching!

8. Recommendations

Recommendations:

I. **Diversification of Content:**

- A. Netflix should add more TV shows because some people like watching those more than movies.

II. **Strategic Collaboration:**

- A. Netflix could talk to famous directors to make more stuff and get more people interested. Even new directors with good >ratings could be a good idea.

III. **Genre Prioritization:**

- A. They should not only focus on international movies. Adding different types like horror and comedy could be cool.

IV. **TV Show Strategy:**

- A. Making more thriller TV shows could be a good idea. People seem to like those for longer.

V. **Content Release Timing:**

- A. Netflix should release stuff on holidays, at the end of the year, and on weekends. That way, more people might watch.

VI. **Direct-to-OTT Release:**

- A. If a movie or show gets good reviews, Netflix should release it directly on their platform. It might get more people to >sign up.

VII. **Celebrity Collaborations:**

- A. Netflix can work with actors who have lots of fans. They could make TV shows or series and more people might watch.

VIII. **Geographical Advertising:**

- A. Netflix should advertise more in places with fewer movies. People there might like their own TV shows, so Netflix can >tell them about it.