In [10]:

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

Problem Statement

Netflix is one of the largest OTT platform worldwide.It lets movies and shows accessible to its sunscribers without any language barrier. At the same time its a bussiness with crores of turnover. Hence it is important to find and forecast the type of content people are actually willing to watch. So we should do the analyse the data given and get insights that help Netflix to grow better.

In [11]:

```
!pip install gdown
```

```
Requirement already satisfied: gdown in c:\users\lenovo\anaconda3\lib\site-packages (4.6.0)
Requirement already satisfied: beautifulsoup4 in c:\users\lenovo\anaconda3\lib\site-packages
(from gdown) (4.11.1)
Requirement already satisfied: requests[socks] in c:\users\lenovo\anaconda3\lib\site-packages
(from gdown) (2.27.1)
Requirement already satisfied: tqdm in c:\users\lenovo\anaconda3\lib\site-packages (from gdow
n) (4.64.0)
Requirement already satisfied: filelock in c:\users\lenovo\anaconda3\lib\site-packages (from
gdown) (3.6.0)
Requirement already satisfied: six in c:\users\lenovo\anaconda3\lib\site-packages (from gdow
n) (1.16.0)
Requirement already satisfied: soupsieve>1.2 in c:\users\lenovo\anaconda3\lib\site-packages
(from beautifulsoup4->gdown) (2.3.1)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\lenovo\anaconda3\lib\site-pa
ckages (from requests[socks]->gdown) (1.26.9)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\lenovo\anaconda3\lib\site-packa
ges (from requests[socks]->gdown) (2021.10.8)
Requirement already satisfied: idna<4,>=2.5 in c:\users\lenovo\anaconda3\lib\site-packages (f
rom requests[socks]->gdown) (3.3)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\lenovo\anaconda3\lib\sit
e-packages (from requests[socks]->gdown) (2.0.4)
Requirement already satisfied: PySocks!=1.5.7,>=1.5.6 in c:\users\lenovo\anaconda3\lib\site-p
ackages (from requests[socks]->gdown) (1.7.1)
Requirement already satisfied: colorama in c:\users\lenovo\anaconda3\lib\site-packages (from
tqdm->gdown) (0.4.4)
```

In [12]:

```
url="https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/000/940/original/netflix.
```

In [5]:

```
df=pd.read_csv(url)
```

In [8]:

```
df
```

Out[8]:

	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	2 Seasons
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons
...	...	...	...	...	...	...	...	...	...	...
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007	R	158 min
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV-Y7	2 Seasons
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	R	88 min
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006	PG	88 min
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-	India	March 2, 2019	2015	TV-14	111 min

show_id	type	title	director	cast	country	date_added	release_year	rating	duration
				Jane Dias, Raaghav Chanan...					

8807 rows × 12 columns

```
In [9]: df.shape
```

```
Out[9]: (8807, 12)
```

There are 8807 unique movie/shows altogether

```
In [13]: 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
```

There are missing values in director,case,country,date_addded,release_year,rati and duration series.

```
In [103… df.isnull().sum()
```

```
Out[103]: show_id            0
type               0
title              0
director        2634
cast             825
country          831
date_added        10
release_year       0
rating             4
duration           3
listed_in          0
description        0
dtype: int64
```

Maximum missing data is found to be in director series which will hinder the ease of analysis

In [9]:

```
df["date_added"].head(1)
```

Out[9]:

```
0    September 25, 2021
Name: date_added, dtype: object
```

In [10]:

```
df["date_added"].tail(1)
```

Out[10]:

```
8806     March 2, 2019
Name: date_added, dtype: object
```

The first day and last day on which the movie/show is added is 02/03/2019 and 25/09/2021

In [44]:

```
country_counts=df["country"].value_counts()
print(country_counts)
```

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

the data includes details of movies/shows from 748 countries

In [27]:

```
total_count=df["type"].value_counts()
```

Out[27]:

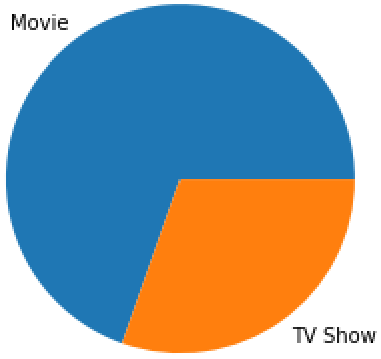
```
Movie      6131
TV Show    2676
Name: type, dtype: int64
```

In [ ]:

```
# Count of TV shows and movies
## distribution of movies and shows in the data given
```

In [50]:

```
plt.pie(total_count,labels=total_count.index)
plt.show()
```



In [49]:

```
unique_com=df[["country","type"]]
```

In [17]:

```
unique_com
```

Out[17]:

	country	type
0	United States	Movie
1	South Africa	TV Show
2	NaN	TV Show
3	NaN	TV Show
4	India	TV Show
...	...	...
8802	United States	Movie
8803	NaN	TV Show
8804	United States	Movie
8805	United States	Movie
8806	India	Movie

8807 rows × 2 columns

unique_com.value_counts().head(20)

Top 10 countries in terms of number os shows/movies

In [35]:

```
movie_list_top10=unique_com[unique_com["type"]=="Movie"].value_counts().head(10)
movie_list_top10
```

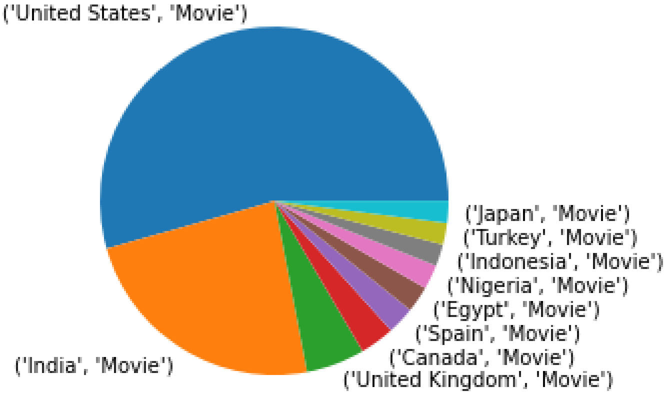
Out[35]:

```
country          type
United States    Movie    2058
India            Movie     893
United Kingdom   Movie     206
Canada           Movie     122
Spain            Movie      97
Egypt            Movie      92
Nigeria          Movie      86
Indonesia        Movie      77
Turkey           Movie      76
Japan            Movie      76
dtype: int64
```

Top 10 countries in terms of Number of movies are United States,India,United Kingdom,Canada,Spain,Egypt,Nigeria,Indonesia,Turkey,Japan

In [36]:

```
plt.pie(movie_list_top10,labels=movie_list_top10.index)
plt.show()
```



United States has the maximum movies produced

In [37]:

```
TVshow_list_top10=unique_com[unique_com["type"]=="TV Show"].value_counts().head(10)
TVshow_list_top10
```

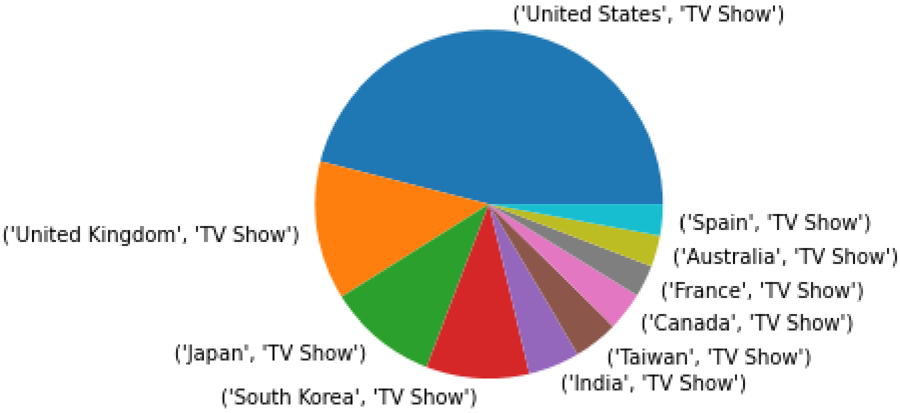
Out[37]:

```
country         type   
United States   TV Show    760
United Kingdom  TV Show    213
Japan           TV Show    169
South Korea     TV Show    158
India           TV Show     79
Taiwan          TV Show     68
Canada          TV Show     59
France          TV Show     49
Australia       TV Show     48
Spain           TV Show     48
dtype: int64
```

Top 10 countries in terms of Number of TV shows are United States,United Kingdom,Japan,South Korea,India,Taiwan,Canada,France,Australia,Spain

In [60]:

```
plt.pie(TVshow_list_top10,labels=TVshow_list_top10.index)
plt.show()
```



United States has the maximum TV shows

In [56]:

```
ratings=df["rating"].value_counts()
ratings
```

Out[56]:

```
TV-MA       3207
TV-14       2160
TV-PG        863
R            799
PG-13        490
TV-Y7        334
TV-Y         307
PG           287
TV-G         220
NR            80
G             41
TV-Y7-FV       6
NC-17          3
UR             3
74 min         1
84 min         1
66 min         1
Name: rating, dtype: int64
```

there are some durations in rating series and for some movies ratings are not available.So we can remove the last 3 rating rows

In [40]:

```
df["director"].unique()
```

Out[40]:

```
array(['Kirsten Johnson', nan, 'Julien Leclercq', ..., 'Majid Al Ansari',
       'Peter Hewitt', 'Mozez Singh'], dtype=object)
```

In [41]:

```
df["director"].nunique()
```

Out[41]:

```
4528
```

In [59]:

```
df["director"].value_counts()
```

Out[59]:

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

In [62]:

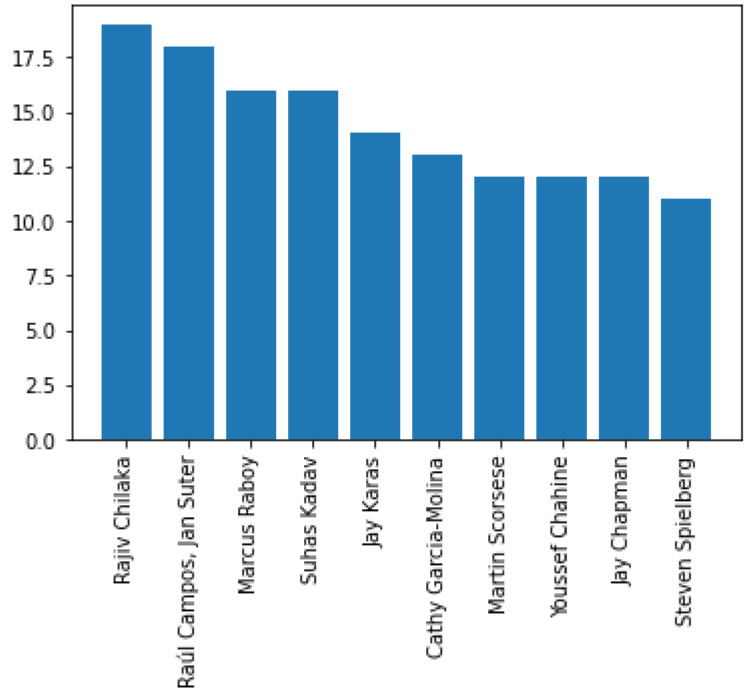
```
director_counts=df["director"].value_counts().head(10)
director_counts
```

Out[62]:

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

In [65]:

```
x=director_counts.index
y=director_counts
plt.bar(x,y)
plt.xticks(rotation=90)
plt.show()
```



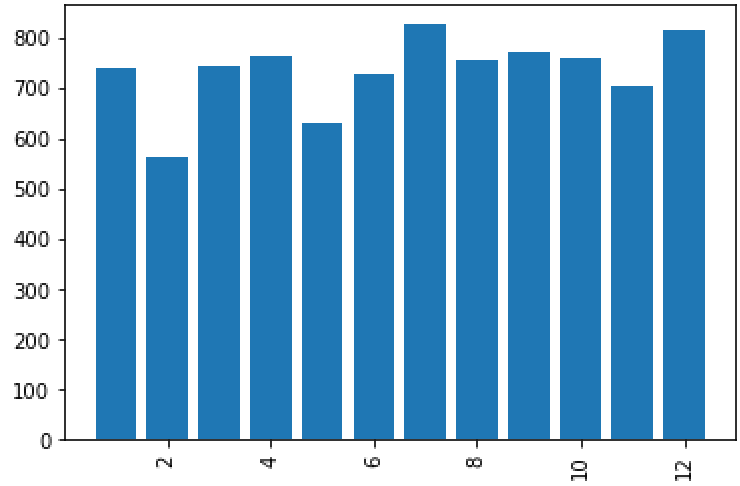
Top 10 directors and their count of movies directed

In [93]:

```
dates=df["date_added"]
dates_array=pd.to_datetime(dates)
month_added=dates_array.dt.month.value_counts()
```

In [94]:

```
x=month_added.index
y=month_added
plt.bar(x,y)
plt.xticks(rotation=90)
plt.show()
```



Most number of movies were added in August Month

In [95]:

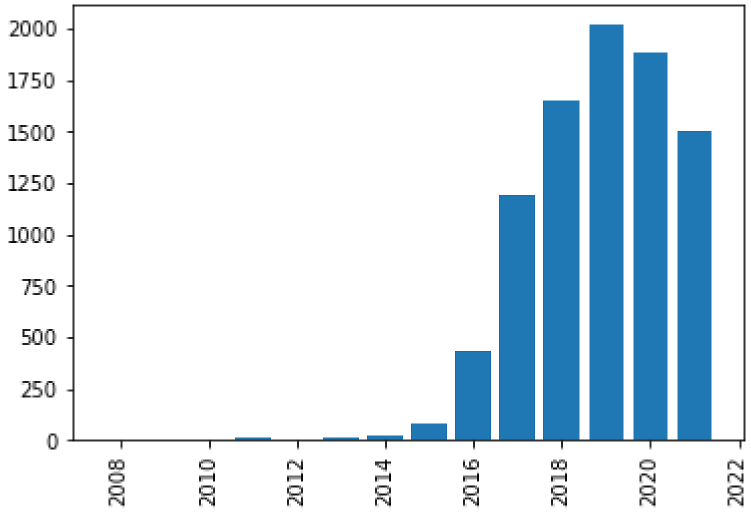
```
dates=df["date_added"]
dates_array=pd.to_datetime(dates)
year_added=dates_array.dt.year.value_counts()
year_added
```

Out[95]:

```
2019.0    2016
2020.0    1879
2018.0    1649
2021.0    1498
2017.0    1188
2016.0     429
2015.0      82
2014.0      24
2011.0      13
2013.0      11
2012.0       3
2009.0       2
2008.0       2
2010.0       1
Name: date_added, dtype: int64
```

In [96]:

```
x=year_added.index
y=year_added
plt.bar(x,y)
plt.xticks(rotation=90)
plt.show()
```



Most number of movies were added in 2019.The reason may be due to the Covis-19 outbreak and Lockdown

In [99]:

```
release_yr=df["release_year"]
release_yr
```

Out[99]:

```
0       2020
1       2021
2       2021
3       2021
4       2021
        ... 
8802    2007
8803    2018
8804    2009
8805    2006
8806    2015
Name: release_year, Length: 8807, dtype: int64
```

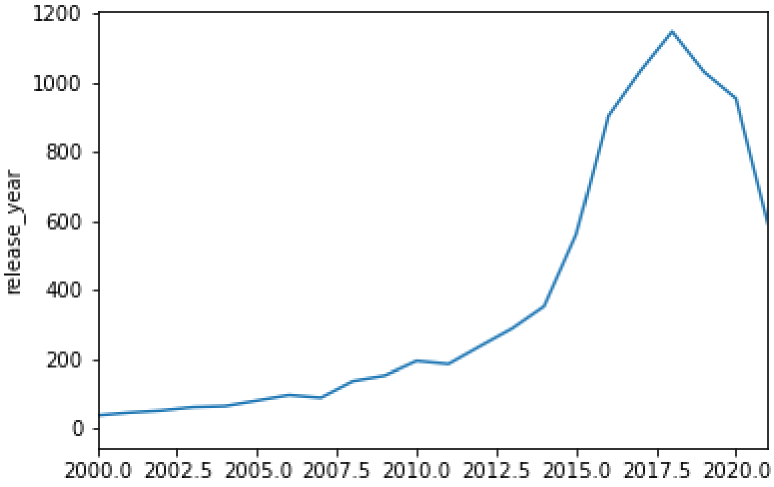
In [100…

```
release_yr.value_counts()
```

```
Out[100]: 2018    1147
          2017    1032
          2019    1030
          2020     953
          2016     902
                  ...
          1959       1
          1925       1
          1961       1
          1947       1
          1966       1
          Name: release_year, Length: 74, dtype: int64
```

```
In [102…  sns.lineplot(data=release_yr,
                       x=release_yr.value_counts().index,
                       y=release_yr.value_counts())
          plt.xlim(left=2000,right=2021)
```

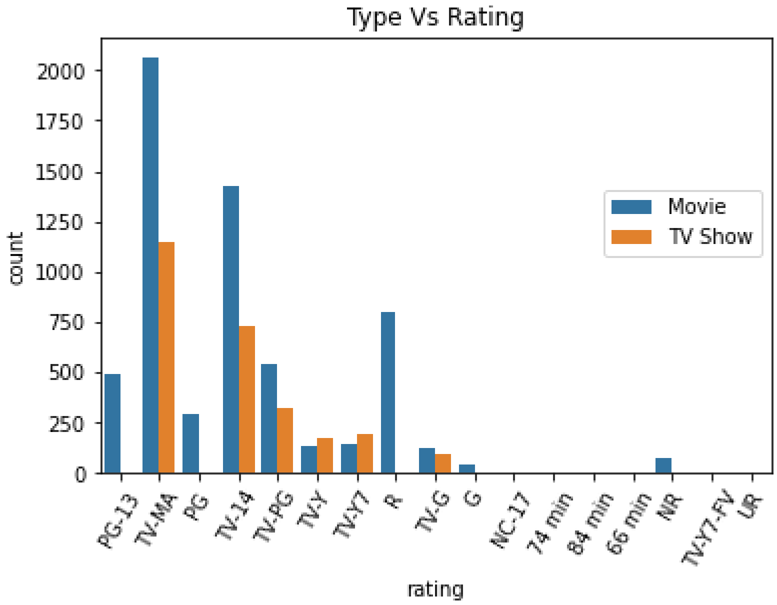
```
Out[102]: (2000.0, 2021.0)
```



since the number of movies released before 2000 were insignificant compared to movies released after 2000, we can trim the graph.

maximum movies were released in 2018

```
In [112…  sns.countplot(data=df,
                        x="rating",
                        hue="type",)
          plt.title('Type Vs Rating')
          plt.xticks(rotation=60)
          plt.legend(loc=(0.75,0.5))
          plt.show()
```



TV-MA Mature Audience content is rated maximum among movies and shows. The rating series contains some entries from duration.

In [152…

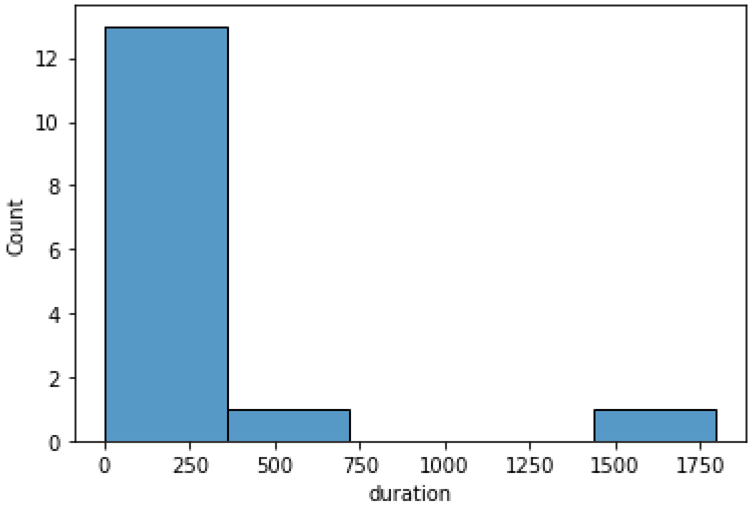
```
Tv_show_duration=df[df["type"]=="TV Show"]["duration"]
Tv_show_duration_20=Tv_show_duration.value_counts()
```

In [153…

```
sns.histplot(Tv_show_duration_20,bins=5)
plt.figure(figsize=(25,20))
```

Out[153]:

```
<Figure size 1800x1440 with 0 Axes>
```



```
<Figure size 1800x1440 with 0 Axes>
```

In [150…

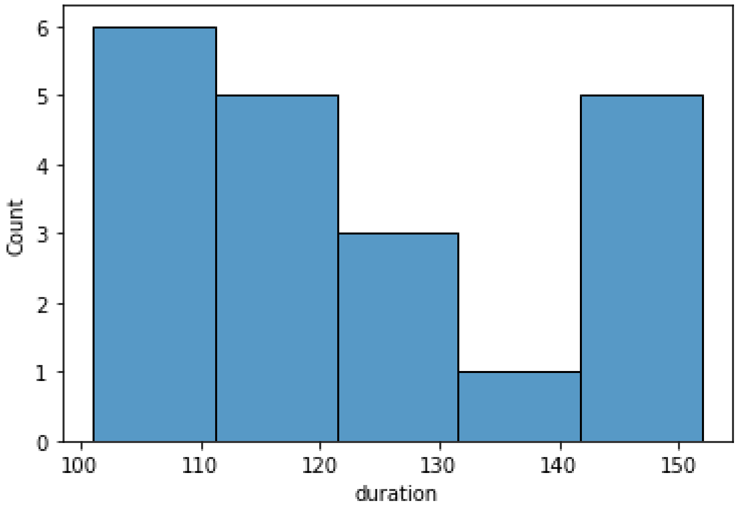
```
movie_duration=df[df["type"]=="Movie"]["duration"]
movie_duration_20=movie_duration.value_counts().head(20)
```

In [151…

```
sns.histplot(movie_duration_20,bins=5)
plt.figure(figsize=(25,20))
```

Out[151]:

```
<Figure size 1800x1440 with 0 Axes>
```



```
<Figure size 1800x1440 with 0 Axes>
```

```
In [ ]:
```

Recommendations

-Top countries in terms of Number of movies are United States,India,United Kingdom,Canada,Spain,Egypt,Nigeria,Indonesia,Turkey,Japan

-Top countries in terms of Number of TV shows are United States,United Kingdom,Japan,South Korea,India,Taiwan,Canada,France,Australia,Spain

-Most number of movies were added in August Month

-TV-MA Mature Audience content is rated maximum among movies and shows. The rating series contains some entries from duration.

-Most number of movies were added in 2019.The reason may be due to the Covis-19 outbreak and Lockdown

Insights

- Produce and release more content by directors of highly rated shows/movies

- Relese movies during holidays or breaks

- most rated genre is

- most of the TV shows were terminated after 1 season. Either the shows should be carefully chosen or make sure the coming seasons have a strong background

- The favourite genre can vary according to majority of people of county. So try get produce and release more of those contents.

```
In [ ]:
```