



NETFLIX BUSINESS CASE ANALYSIS

This Netflix case study 🏠 pertains to the collection of data from all over the world 🌎 with the primary selection of the data along with the cast and director contributed with the most top rated rating categories 📈 with the genere type to fetch the analysis with individuals around the globe who are interested in which of the categories they are fall in throughout. A portion of the data activity has also been assessed by assessing personal perspective and conduct, but all testing and analysis in this data set are only enrolled with the questions listed on the scaler platform when executing the questions attributes. I played a significant role in this endeavour by providing a personal perspective while conducting analysis.

```
In [3]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

We must first import the Python libraries into our work notebook in order to access the options as functions and methods to deal with the NETFLIX data situations that we must analyse, interpretate, and visualise.

READING DATA FILE

```
In [11]: df = pd.read_csv('Netflix_case_studies.csv')
```

The code above is for reading a CSV (comma-separated values) data file, using an approach from the Pandas library to read a specific file.

Here, the variable known (df) as a dataframe is only used to force us to assess all of the data into one variable before the ensuing operations.

SHAPE OF DATA

```
In [359]: df.shape
```

```
Out[359]: (8807, 12)
```

The shape method retrieves the numerical values from the data, which indicates that the data is available in the form of rows and columns with a specific number of rows and columns.

SIZE OF DATA

```
In [360]: df.size
```

```
Out[360]: 105684
```

The size method returns the total number of data from the data set which represents the whole number of row and column with the dot product

INFORMATION OF DATA

In [361]: 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
```

The info() method is a part of the Pandas library, and it returns both the data type of the column and the entire data column into the frame.

Here, the object data type indicates that the major values are categorical, and the "releasing_year" enters the int data type just to obtain the values in integer format.

DATA TOP SIGNIFIES

In [362]: df.head()

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

The above statement signifies a portion of the data, just to clarify the data set and the value of the data set, like a glimpse of the dataset. 🌟

The head function comes from the pandas libraries which returns the top 5 elements from the data set, the default value for the head is 5! else we can change the values from any range as per the requirement needs.

DATA BOTTOM SIGNIFIES

In [363]: df.tail()

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey Jr...	United States	November 20, 2019	2007	R	158 min	Cult Movies, Dramas, Thrillers	A political cartoonist, a crime reporter and a...
8803	s8804	TV Show	Zombie Dumb	Nan	Nan	Nan	July 1, 2019	2018	TV-Y7	2 Seasons	Kids' TV, Korean TV Shows, TV Comedies	While living alone in a spooky town, a young g...
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	R	88 min	Comedies, Horror Movies	Looking to survive in a world taken over by zo...
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006	PG	88 min	Children & Family Movies, Comedies	Dragged from civilian life, a former superhero...
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chan...	India	March 2, 2019	2015	TV-14	111 min	Dramas, International Movies, Music & Musicals	A scrappy but poor boy worms his way into a ty...

The above statement signifies a portion of the data, just to clarify the data set and the value of the data set, like a glimpse of the dataset.

The tail function comes from the pandas libraries which returns the bottom 5 elements from the data set, the default value for the tail is 5! else we can change the values from any range as per the requirement needs.

STATISTICAL DESCRIPTION OF DATA

In [364]: df.describe()

```
Out[364]: 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 describe function name from the pandas library return the glimpse of the data with the statistical values from all over the data just to predict the normal ranges and average ranges to the particular elements.

Note: it will display only the numerical values and return from the numerical values.

STATISTICAL DESCRIPTION OF DATA WITH STRING/ OBJECT FORMAT CONCLUSION

In [365]: df.describe(include=object)

	show_id	type	title	director	cast	country	date_added	rating	duration	listed_in	description
count	8807	8807	8807	6173	7982	7976	8797	8803	8804	8807	8807
unique	8807	2	8807	4528	7692	748	1767	17	220	514	8775
top	s1	Movie	Dick Johnson Is Dead	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	TV-MA	1 Season	Dramas, International Movies	Paranormal activity at a lush, abandoned propo...
freq	1	6131	1	19	19	2818	109	3207	1793	362	4

The describe method with the object data type can also be called and returned from the data frame, with the key including the object as a parameter that decribes shows the statistical measures with the possible attribute with the columns name.

DATA CLEANING AND FEATURING

In [366]: ⏷ df.isnull()

Out[366]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	False	False	False	False	True	False	False	False	False	False	False	False
1	False	False	False	True	False	False	False	False	False	False	False	False
2	False	False	False	False	False	True	False	False	False	False	False	False
3	False	False	False	True	True	True	False	False	False	False	False	False
4	False	False	False	True	False	False	False	False	False	False	False	False
...
8802	False	False	False	False	False	False	False	False	False	False	False	False
8803	False	False	False	True	True	True	False	False	False	False	False	False
8804	False	False	False	False	False	False	False	False	False	False	False	False
8805	False	False	False	False	False	False	False	False	False	False	False	False
8806	False	False	False	False	False	False	False	False	False	False	False	False

8807 rows × 12 columns

isnull will return the data as a boolean values! ✨ where the values with the NA or null comes as True and rest comes with the False category lets see the total number of values which are Nulls on the data set

In [367]: ⏷ df.isna().sum()

Out[367]:

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	

The above code returns the number of NA value from the data set, which shows the elements in the dataset having a lot of null values in it.

In [368]: ⏷ df.fillna("UNKWN", inplace=True)

We cannot drop the Na values from the data as, the data will be impacted from the original data, so to overcome on this, we have to fill the NA values as passing with some of the object name as "UNKWN"

In [369]: ⏷ df.isna().sum()

Out[369]:

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	

Now, the every column and rows are not return any NA values from the dataset.
it shows in above analysis -> the whole column does not contain any NA/ NULL values 🎉

In [370]: df.head()

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

The show_id does not give any values insight with the DataFrame so we should remove this for a better and understandable insight with the Data, 📈 Below is the code for dropping the certain column

SPLITTING COLUMN WITH MULTIPLE NAMES INTO SINGLE NAME

In [371]: df['cast'].str.split(", ", expand = True).head()

	0	1	2	3	4	5	6	7	8	9	...	40	41	42	43	44	...
0	UNKWN	None	None	None	None	None	None	None	None	None	...	None	None	None	None	None	None
1	Ama Qamata	Khosi Ngema	Mabalane	Gail Molaba	Thabang Molaba	Dillon Windvogel	Natasha Thahane	Arno Greeff	Xolile Tshabalala	Getmore Sithole	Cindy Mahlangu	...	None	None	None	None	None
2	Sami Bouajila	Tracy Gotoas	Samuel Jouy	Nabiba Akkari	Sofia Lesaffre	Salim Kechiouche	Noureddine Farahi	Geert Van Rampelberg	Bakary Diombera	None	...	None	None	None	None	None	None
3	UNKWN	None	None	None	None	None	None	None	None	None	...	None	None	None	None	None	None
4	Mayur More	Jitendra Kumar	Ranjan Raj	Alam Khan	Ahsaas Channa	Revathi Pillai	Urvi Singh	Arun Kumar	None	None	...	None	None	None	None	None	None

5 rows × 50 columns

The above scenario for the un-nesting the DataFrame while we have a cast column where there are multiple number of names of the cast which is separated by the comma, the above code for the un-nesting the multiple values for the cast name which has been divided into the number of frames! 🔍

In [372]: df['cast'].str.split(", ",expand= True).stack()

```
Out[372]: 0      0          UNKWN
           1      0          Ama Qamata
           1          Khosi Ngema
           2          Gail Mabalane
           3          Thabang Molaba
           ...
8806  3          Manish Chaudhary
     4          Meghna Malik
     5          Malkeet Rauni
     6          Anita Shabdish
     7          Chittaranjan Tripathy
Length: 64951, dtype: object
```

The casted categories for the multiple values into the single values, and stacked over the by over for the names formed into the series, so that can be extract easily for the rest either to be change into the Frame or by series. 🔍

DROPPING

```
In [373]: df.drop(columns=['show_id'], inplace=True)
```

The above code snippet will drop the column with the specific name categorised. Here the drop feature is shown for just selecting the feature of pandas. The rest show_id can be used later if analysis required. ☺

COLUMN NAMES

```
In [374]: df.keys()
```

```
Out[374]: Index(['type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
      dtype='object')
```

Keys 🔑 are the method which returns all the col name from the dataset

RETURNING COUNT OF TYPES

```
In [375]: df['type'].value_counts()
```

```
Out[375]: Movie      6131
TV Show    2676
Name: type, dtype: int64
```

Statement describes the Number of movies in the Dataset is 6131 and TV shows conclude 2676, well it returns the features as we say the HUE from the dataset which plays important role while Framing

DISPLAYING THE NUMBER OF MOVIES AND SHOW BASED ON RATINGS

```
In [376]: df[['type', 'rating']].value_counts()
```

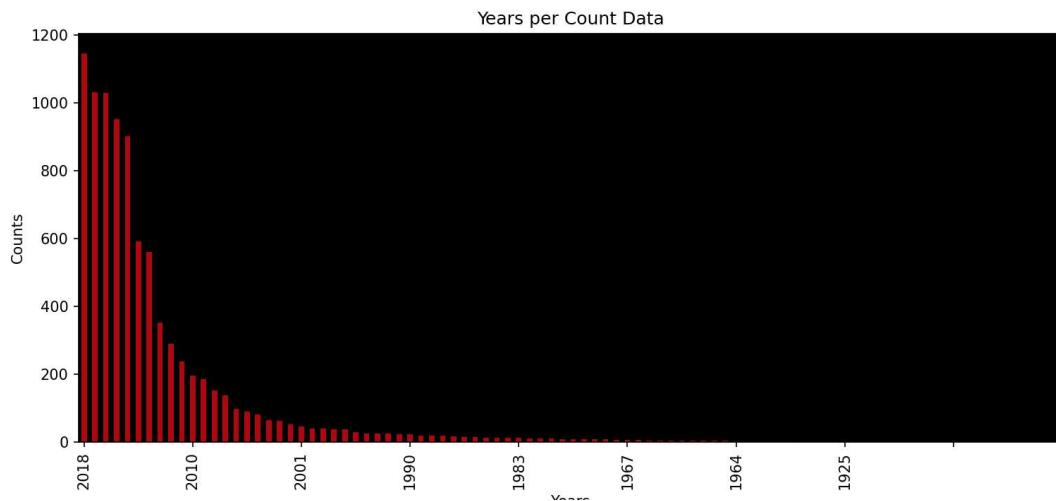
```
Out[376]: type      rating
Movie     TV-MA        2062
          TV-14        1427
TV Show   TV-MA       1145
Movie     R            797
TV Show   TV-14       733
Movie     TV-PG        540
          PG-13        490
TV Show   TV-PG        323
Movie     PG           287
TV Show   TV-Y7        195
          TV-Y          176
Movie     TV-Y7        139
          TV-Y          131
          TV-G          126
TV Show   TV-G          94
Movie     NR           75
          G             41
          TV-Y7-FV       5
TV Show   NR           5
Movie     NC-17         3
          UR            3
TV Show   UNKWN        2
          R             2
Movie     UNKWN        2
          74 min         1
          84 min         1
TV Show   TV-Y7-FV       1
Movie     66 min         1
dtype: int64
```

The above non graphical solution for the particular Movie 🎬 and TV show 📺 for each rating industries, just to signify that every rating publisher having what terms of ratings has been given by the people.

DISPLAYING HOW MANY MOVIES AND SHOWS PRODUCED IN YEAR

```
In [377]: plt.figure(figsize=(12,5))
df['release_year'].value_counts().plot(kind='bar',color="#b20710")
plt.xticks(range(0,100,10))
plt.xlabel("Years")
plt.ylabel("Counts")
plt.title("Years per Count Data")
ax = plt.gca()
ax.set_facecolor("black")
plt.show()
```

<IPython.core.display.Javascript object>

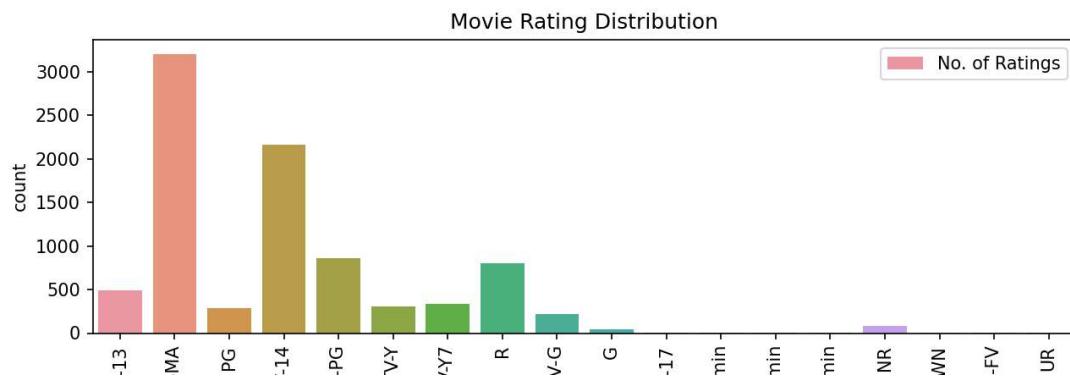


The Above scenario shows the year wise the count of movies and TV shows has been made and the tendency of making movie is bright higher in the 2018 as compare to the year 1925
 Business View: The increasing segments of the movies and shows in the coming years will be have great impact while coming in the new era, so presenting movies and shows in coming year will grow!

HOW MANY RATINGS GATHERED FOR EACH RATING COMPANY

```
In [379]: plt.figure(figsize=(10,3))
sns.countplot(data=df, x= 'rating', label = "No. of Ratings")
plt.xticks(rotation = 90)
plt.title("Movie Rating Distribution")
plt.legend()
plt.show()
```

<IPython.core.display.Javascript object>



The above scenario of the Graph displays the rating categories with the count of every categories followed, We here simple standarise the concept for the graph shows the "TV-MA" 🌟 rating have the maximum number of count ratings approx more than 3000

Business View: The rating companies given the best majority role for the people or the movie makers to enhance the role and segment of the movies which been categorised as the highest rated and lowest rated into the frame! While business will see the behaviour for the ratings while conducting.

HOW MANY MOVIES AND SHOWS ARE THERE

```
In [380]: plt.figure(figsize=(10,4))
plt.hist(data=df, x ="type", label="Movie || TV Shows", color='Darkred')
plt.xlabel("Type Of shows")
plt.ylabel("No. of Shows")
ax = plt.gca()
ax.set_facecolor("Black")
plt.title("Distribution Among types")
plt.legend()
plt.grid()
plt.show()
```

<IPython.core.display.Javascript object>



The above graph shows the number of Tv_shows and movies present the data, Here we can judge he Movie are higher watching instances from the audience as compare to the Tv_shows 🤝
Movies are Higher 🤝
Tv_Shows are lower 🤝

Business view: The making of the Tv shows as compare to the Movie is less, so people are more interacted and archived towards movie segments, the growth can be seen in movies as compare to the TV shows

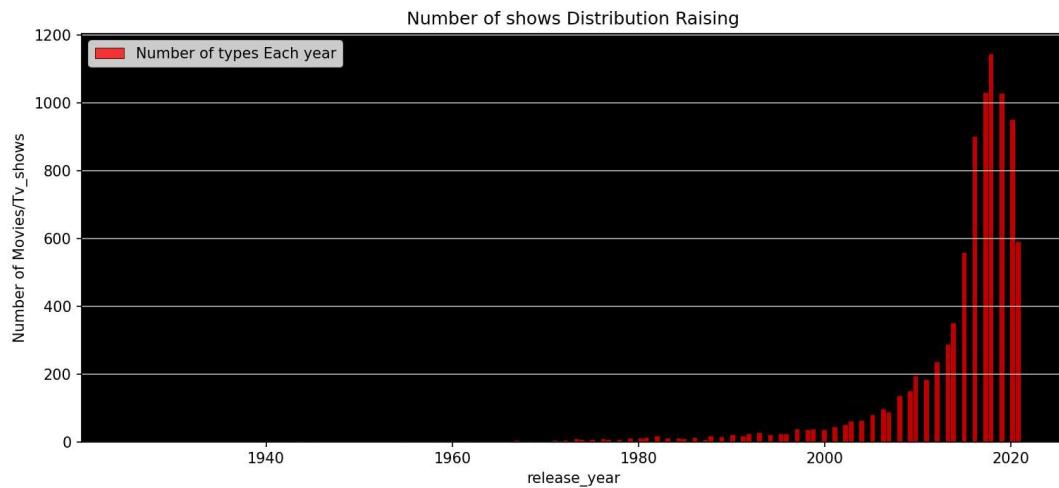
```
In [381]: df.head()
```

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

TRENDING OF SHOWS IN A YEAR

```
In [382]: plt.figure(figsize=(12,5))
sns.histplot(data=df, x='release_year', color="red", label = "Number of types Each year")
plt.grid(axis="y")
plt.ylabel("Number of Movies/Tv_shows")
plt.title("Number of shows Distribution Raising")
ax = plt.gca()
ax.set_facecolor("Black")
plt.legend()
plt.show()
```

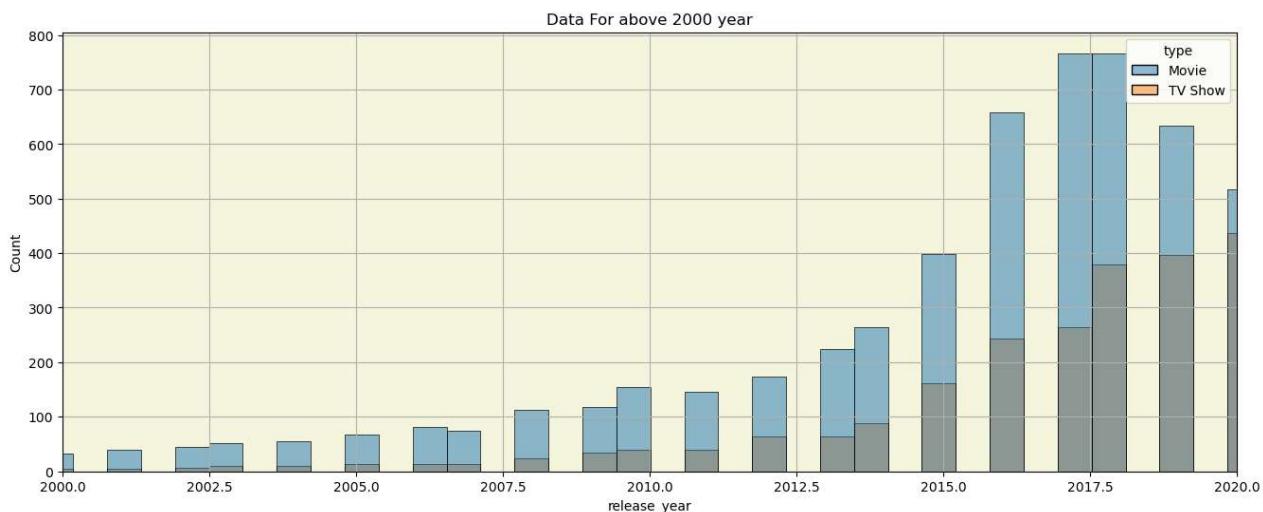
<IPython.core.display.Javascript object>



Graph retuns the Movies is been Released in a particluar year, As, the number of movies in year 1940 is ver less than movies released on year 2020 🎬. Business view: The action towards releasing year is more hyper than the previous year, so by the analysis we can perform the rights that the business are growing very high towards recent years.

TRENDING OF SHOWS ABOVE FROM 2000 YEAR

```
In [21]: plt.figure(figsize = (16,6))
sns.histplot(data =df, x= 'release_year', hue = 'type')
plt.xlim(2000,2020)
ax = plt.gca()
ax.set_facecolor("beige")
plt.grid()
plt.title("Data For above 2000 year")
plt.show()
```

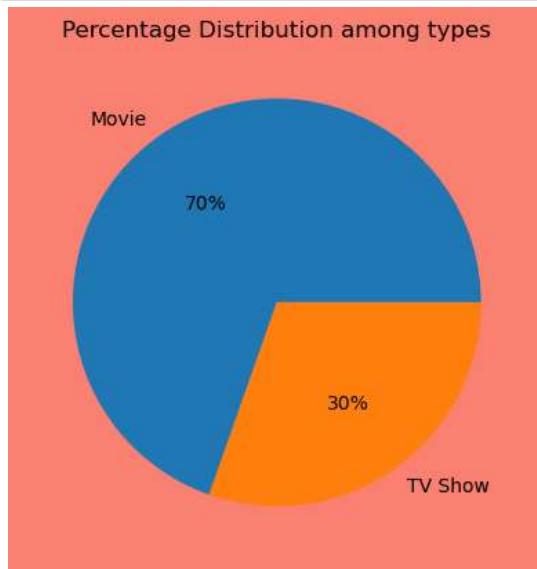


The above Dataset results, the releasing year above from the year 2000, where we can observe the majority of the people and director makes the movie very vast and till the year 2017 and some deflation we can see from the year 2017.

Business View: The Analysis have a relative with the movies and shows, we can surely see the depicts from the graph that the tendency of movies and tv shows, so upon that the directors must allow the level of movoies and shows making all over world.

PERCENTAGE CONTRIBUTION OF SHOWS || COMPARISON

```
In [22]: movie_type_counts = df['type'].value_counts()
movie_type_counts
plt.figure(facecolor="Salmon", frameon=True, label="Tv_Show")
plt.title("Percentage Distribution among types")
plt.pie(movie_type_counts.values, labels=movie_type_counts.index, autopct='%.0f%%')
plt.show()
```

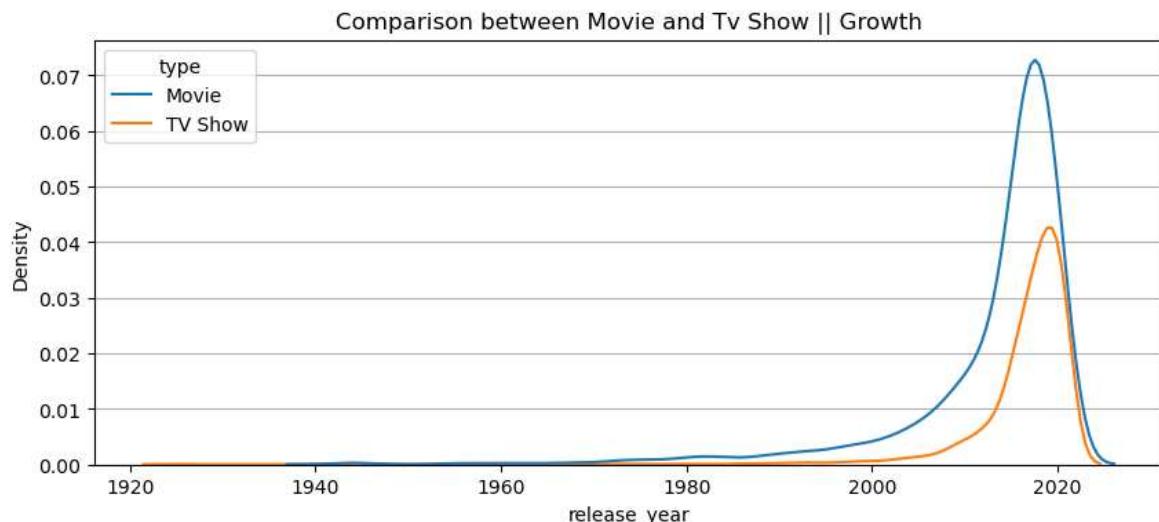


The pie chart return the 30% of the director has produced the Tv_shows as well as the 70% for the Movies. And also the majority of the audience also be in the touch for the Movies as related to Tv_Shows.

Business view: The major contribution in all over the world is movies, from the 70% of the population is shifted towards movies while the tv shows having minor acceptance. Here, definately shows the major popluation have interest in Movies.

TRENDING COMPARISON OF MOVIES AND SHOWS

```
In [175]: plt.figure(figsize=(10,4))
sns.kdeplot(data=df, x ="release_year", hue = "type")
plt.grid(axis="y")
plt.title("Comparison between Movie and Tv Show || Growth")
plt.show()
```



The Above sceario shows the the trend between the Movies and the Tv show in the year wise area! 📈 Where the Tv shows also comes into the trend in the coming year, where we can clear se from the year 2000 it starts rising but the movies stills in the culture of trending high for the Movies.

TOP 10 COUNTRIES WITH MOVIES AND TV SHOWS

In [383]: ⏷ df[['country', 'type']].value_counts()[:10]

```
Out[383]: country      type
United States   Movie     2058
India           Movie     893
United States   TV Show   760
UNKWN          Movie     440
                  TV Show   391
United Kingdom TV Show   213
                  Movie     206
Japan           TV Show   169
South Korea    TV Show   158
Canada          Movie     122
dtype: int64
```

The above series tells about, the top 10 number of Countries with the highest number of movies and Tv Shows 📈 Business View: The major number of movies and shows are nbeing placed in into the big countries like United states and India with the highest number of movies and shows which reflects the potential of people in these two countries where the major people interact and interested in the watching movies and shows respect to all population.

NUMBER OF TV SHOW

In [384]: ⏷ tvshow = df[df['type'] == "TV Show"].nunique()

```
Out[384]: type      1
title     2676
director  223
cast      2284
country   197
date_added 1052
release_year 46
rating    10
duration   15
listed_in  236
description 2672
dtype: int64
```

NUMBER OF MOVIES

In [385]: ⏷ movies = df[df['type']=='Movie'].nunique()

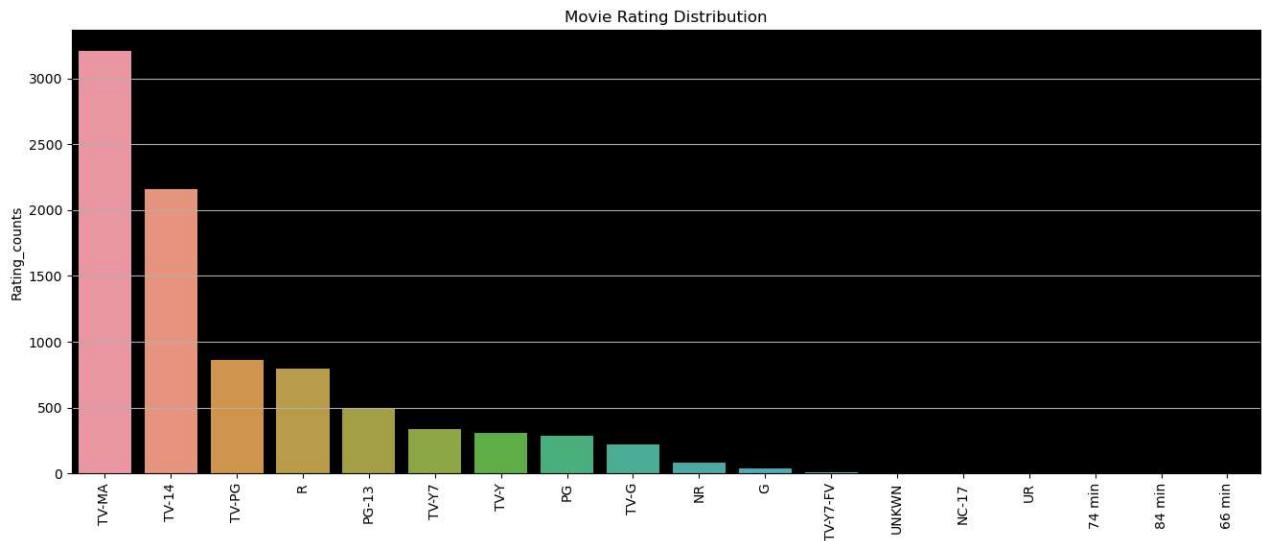
```
Out[385]: type      1
title     6131
director  4355
cast      5446
country   652
date_added 1533
release_year 73
rating    18
duration   206
listed_in  278
description 6105
dtype: int64
```

From the above observation we can see, the clear difference between the type of genre like movies and Tv shows, Where the TV shows having very less director as compare to Movies. 🎬 Also the less countries were having a follow the TV shows (197) and movies were(652) respectively. 🌎 Business view: Here we can observe that the tendency of the movies and shows has the very large impact on the making sceanrio for the director and cast, the number shows the dependencies and the number of movies all over around has enough potential to describe the scenario for the growth.

RATING DISTRIBUTION AMONG COUNT OF ALL

```
In [28]: rat_index = df['rating'].value_counts().index
rat_val = df['rating'].value_counts().values

plt.figure(figsize=(16,6))
sns.barplot(data=df,x=rat_index, y=rat_val)
plt.xticks(rotation = 90)
ax = plt.gca()
ax.set_facecolor("Black")
plt.grid(axis = "y")
plt.ylabel("Rating_counts")
plt.title("Movie Rating Distribution")
plt.show()
```

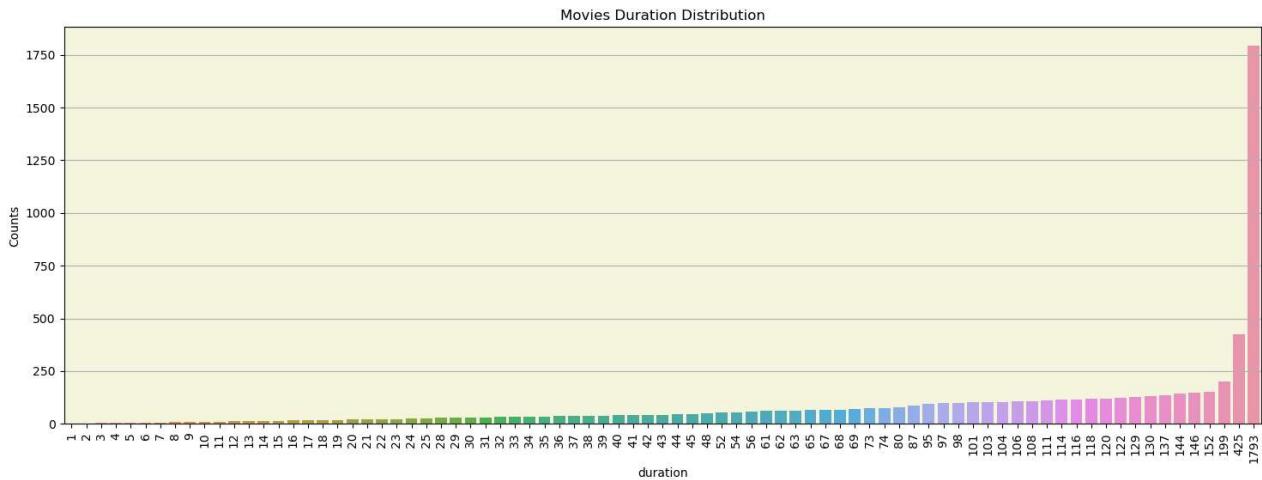


The Picture represents the main frame of the movies distribution with the help of the counting and rating of the particular Movies and TV shows.
Where, we can observe the major rating has been allotted to the "TV-MA" channel with the major contribution more than 3000 values.
Conclusion: people more often to do ratings on the "TV-MA" in case of the rest of the mediums.

DURATION OF MOVIES AND SHOWS

```
In [29]: dura_ind = df['duration'].value_counts()
dura_val = df['duration'].value_counts()

plt.figure(figsize=(18,6))
sns.barplot(data=df, x=dura_ind, y=dura_val)
ax = plt.gca()
ax.set_facecolor("beige")
plt.grid(axis="y")
plt.ylabel("Counts")
plt.title("Movies Duration Distribution")
plt.xticks(rotation = 90)
plt.show()
```



The above scene sets the distribution channel for the displays of the movies and TV_shows, likewise in the graph we can observe the duration for the movies will have the sequence more minutes with the movies and Tv_shows displays the highest with the 1793min, which clarifies the it has many sesons and more watch minutes in particular episode.

FEATURE SELECTION

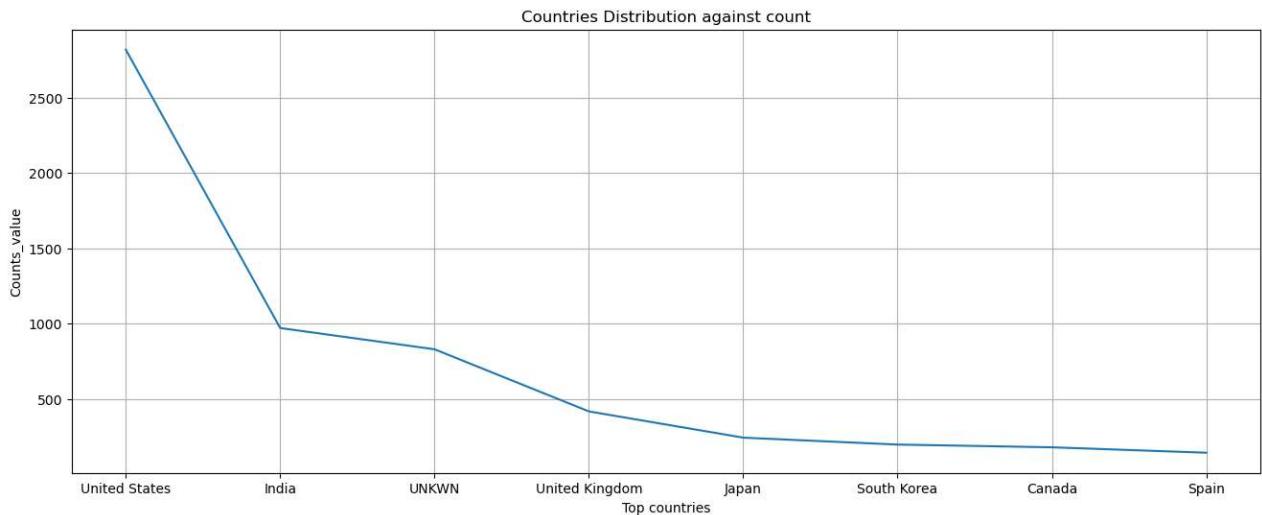
```
In [30]: new_data = df[['title', 'type', 'director', 'cast', 'country', 'release_year', 'rating', 'duration', 'description']]
new_data.set_index('title', inplace=True)
new_data.head()
```

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

The Above generated data frame is called the Feature selection Data where we can see the particular movie title have the specific elaboration among the categories with the helpful inflation of the particular movies or Tv_shows. With this way of analysing will give more certain assumption in respect to the title which is movie or show names and can also we can judge the scenario while every specific options.

TRENDING OF MOVIES AND SHOWS IN ALL OVER COUNTRY

```
In [45]: group_counts = df.groupby('country').size()
group_counts = pd.Series(group_counts)
group_counts = group_counts.sort_values(ascending = False)
plt.figure(figsize=(16,6))
sns.lineplot(group_counts[:8])
ax = plt.gca()
ax.set_facecolor("white")
plt.title("Countries Distribution against count")
plt.xlabel("Top countries")
plt.ylabel("Counts_value")
plt.grid()
plt.show()
```



The graph declares the top wise countries which has a maximum number of movies and Tv shows in the picture! The staright seen is the united states having the majority with the all of the countires in terms of everything like: director, type, cast, the majority is falling into the united states, then india and soo on.

Business View: The Major insights comes with the united states followed by the countries, where the major movies and shows are alinged with the united states, the production and the attribute towards the inclination of movies is the united states , india and so on, so the template for the business investment in the range for the movies is the united states and india and so on..

INDIA TOP 10 OLDEST MOVIES

let's see the scenario of the indian movies with the most oldest top 10 of movies/shows india has released. The below is the DataFrame where the oldest picture have the date as 1959 and 1973 respectively.

```
In [388]: india_movies= df[df['country'] == "India"]
india_movies[['release_year','title']].sort_values(by="release_year", ascending = True)[:10]
```

Out[388]:	release_year	title
542	1959	Ujala
5400	1962	Professor
7932	1964	Sangam
5391	1966	Amrapali
5461	1969	Prince
5396	1971	Lal Patthar
5545	1971	Elaan
7224	1972	Koshish
6254	1972	Bawarchi
5300	1973	Khoon Khoon

INDIA TOP 10 NEWEST MOVIE

let's see the scenario of the indian movies with the most newest top 10 of movies/shows india has released.
The below is the DataFrame where the newest picture have the date as 2021 as newest recent release respectively.

```
In [389]: ⚡ india_movies= df[df['country'] == "India"]
india_movies[['release_year','title']].sort_values(by="release_year", ascending = False)[:10]
```

Out[389]:

	release_year	title
4	2021	Kota Factory
1037	2021	Tuesdays & Fridays
735	2021	Sarbat
850	2021	99 Songs
871	2021	Sardar Ka Grandson
873	2021	Ahaan
876	2021	Alma Matters
903	2021	Nayattu
909	2021	Milestone
918	2021	Lava Ka Dhaava

TOP 10 DIRECTORS FOLLOWED BY THEIR MOVIES COUNTS

```
In [390]: ⚡ df[['director','type']].value_counts().sort_values(ascending = False).head(10)
```

Out[390]:

director	type	count
UNKWN	TV Show	2446
	Movie	188
Rajiv Chilaka	Movie	19
Raúl Campos, Jan Suter	Movie	18
Suhas Kadav	Movie	16
Marcus Raboy	Movie	15
Jay Karas	Movie	14
Cathy Garcia-Molina	Movie	13
Jay Chapman	Movie	12
Youssef Chahine	Movie	12

dtype: int64

Above scene is the way to touch the directors followed by the Tv shows and movies according to the number of count of values.
Here is the top 10 directors who made the Movies and Tv shows with the corresponding number of shows.

TOP DIRECTORS WITH UNIQUE NUMBERS OF SHOWS

```
In [391]: ⚡ df.groupby(by='director')['type'].nunique():10
```

Out[391]:

director	count
A. L. Vijay	1
A. Raajdheep	1
A. Salaam	1
A.R. Murugadoss	1
Aadish Keluskar	1
Aamir Bashir	1
Aamir Khan	1
Aanand Rai	1
Aaron Burns	1
Aaron Hancox, Michael McNamara	1

Name: type, dtype: int64

This Above feature exploration comes into the play with the director and which has made the movies and Tv shows with the top one's
Like this exploration comes up with the top 10 director had made the movies and Tv_shows

TOP DIRECTORS WITH THEIR MOVIES NAMES

```
In [392]: df.groupby(by='director')['title'].value_counts()[:10]
```

```
Out[392]: director      title
A. L. Vijay      Abhinetri      1
                  Watchman       1
A. Raajdheep     Asura Guru      1
A. Salaam        Salaakhen       1
A.R. Murugadoss Sarkar          1
                  Spyder          1
Aadish Keluskar Jaoon Kahan Bata Ae Dil 1
Aamir Bashir     Harud          1
Aamir Khan       Taare Zameen Par 1
Aanand Rai       Tanu Weds Manu 1
Name: title, dtype: int64
```

The above data exploration comes with the top 10 directors had made the movies and Tv shows followed by the title which is movie name!

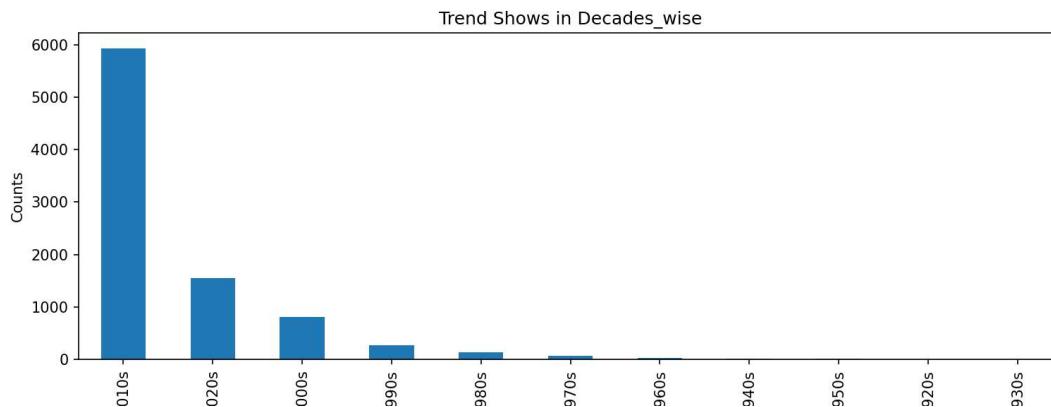
Like: The director name with the A.L. Vijay had made the best movie which is "Abhinetri"

```
In [393]: df['release_decade'] = pd.cut(df['release_year'], bins=[0, 1929, 1939, 1949, 1959, 1969, 1979, 1989, 1999, 2009, 2019, float('Inf')])
```

EVERY DECADES TRENDING SHOWS

```
In [397]: plt.figure(figsize=(12,4))
df['release_decade'].value_counts().plot(kind='bar')
plt.xlabel("Decades-Release")
plt.ylabel("Counts")
plt.title("Trend Shows in Decades_wise")
plt.show()
```

<IPython.core.display.Javascript object>



Analysing the movies into the decade wise, the growth and the trend of movies can be clearly shows the latest date year having the more movie in making a compare to the before as 1930's.

The trend and growth are likely more in the 2010's part.

Business view: The trending with the centuries followed from the earliest and newest, decade wise shows the major popularity in the movies and shows has the highest rankin in the newest decades. The coming early shows in the major comprehensive is the coming centuries from the data.

```
In [398]: df.head(2)
```

	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description	release_decade
0	Movie	Dick Johnson Is Dead	Kirsten Johnson	UNKWN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, film...	2020s
1	TV Show	Blood & Water	UNKWN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...	2020s

```
In [404]: df = pd.read_csv('Netflix_case_studies.csv')
df.head()
```

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

TO CALCULATE THE MOVIES TIME

```
In [405]: df['movie_mins'] = np.where(df['type']=='Movie', df['duration'].str[:-4], None)
df['movie_mins'] = df['movie_mins'].astype('float')
df.head(2)
```

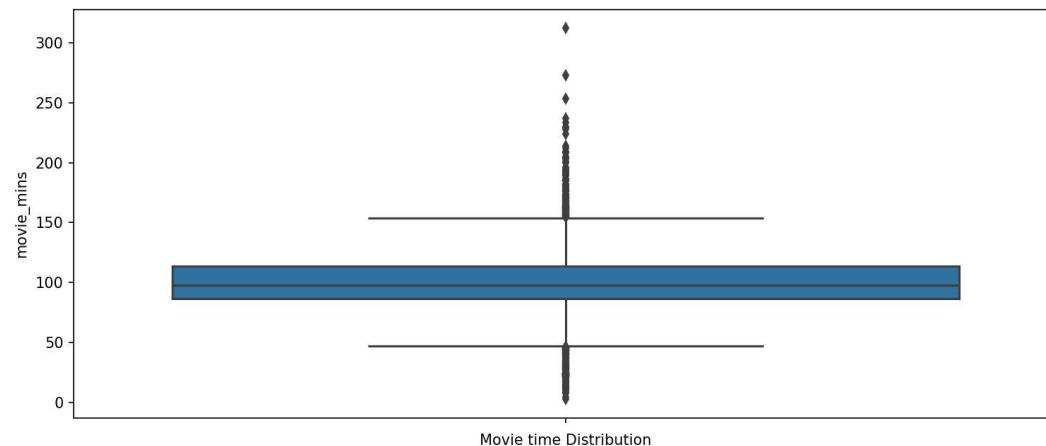
	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description	movie_mins
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Nan	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...	90.0
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...	NaN

Here, we have made an another column for the Movie Duration where we can find the duration of movies with the help of the box plot.

BOX PLOT DISTRIBUTION AGAINST MOVIES TIME DURATION

```
In [407]: plt.figure(figsize=(12,5))
sns.boxplot(data=df, y='movie_mins')
plt.xlabel("Movie time Distribution")
plt.show()
```

<IPython.core.display.Javascript object>

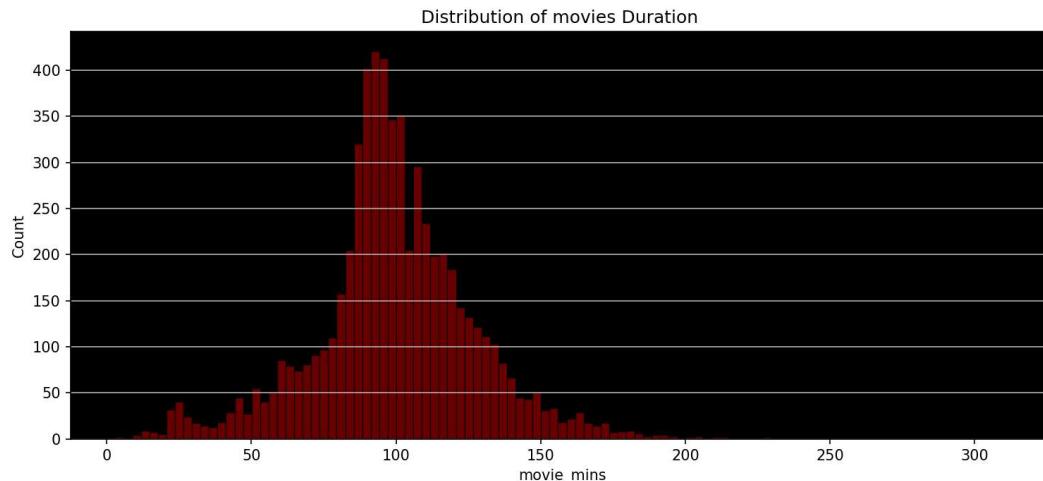


The above time duration for movie signifies the minimum duration time for movie is lagging upto the 50mins, and the upper bound for the movie time duration is approx 160 minutes

Business view: The least and the maximum duration gives insights for the movies show, here from the avg time as 50% the duration is its peak with the almost the 90min, which shows the max of the movie time incorporates the legend of maximum watching and trending big hits time.

```
In [410]: plt.figure(figsize=(12,5))
sns.histplot(data=df, x='movie_mins', color="darkred")
plt.grid(axis="y")
plt.title("Distribution of movies Duration ")
ax = plt.gca()
ax.set_facecolor("Black")
plt.show()
```

<IPython.core.display.Javascript object>



The same observation has been came up with the movies distribution duration channel with the hist plot comprises the maximum peak value as duration for the movies is approx 80-90 minutes.

Business view: The least and the maximum duration gives insights for the movies show, here from the avg time as 50% the duration is its peak with the almost the 90min, which shows the max of the movie time incorporates the legend of maximum watching and trending big hits time. Targeting the the notch values from the graph will give desire result for the directors and good responsive notch while coming to the market.

EXTRACTING SEASONS FROM TV SHOWS

```
In [412]: df['tvshow_mins'] = np.where(df['type']=='TV Show', df['duration'].str[:-7], None)
df['tvshow_mins'] = df['tvshow_mins'].astype("float")
df.head(1)
```

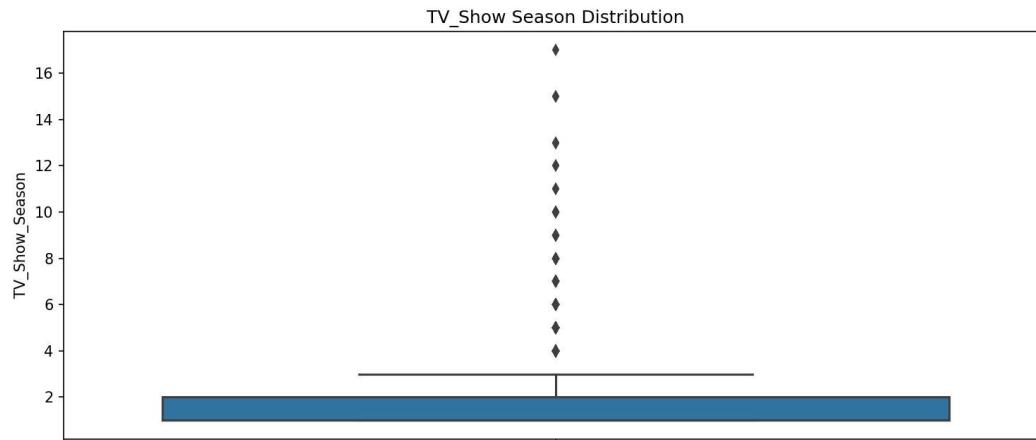
	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description	movie_mins	tvshow_mins
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, film...	90.0	NaN

Here, we have made an another column for the Tv_show Duration as a season where we can find the duration of Tv_shows as in season with the help of the box plot.

BOX PLOT AGAINST NUMBER OF SEASONS

```
In [418]: plt.figure(figsize=(12,5))
sns.boxplot(data=df, y="tvshow_mins")
plt.ylabel("TV_Show_Season")
plt.title("TV_Show Season Distribution")
plt.show()
```

<IPython.core.display.Javascript object>

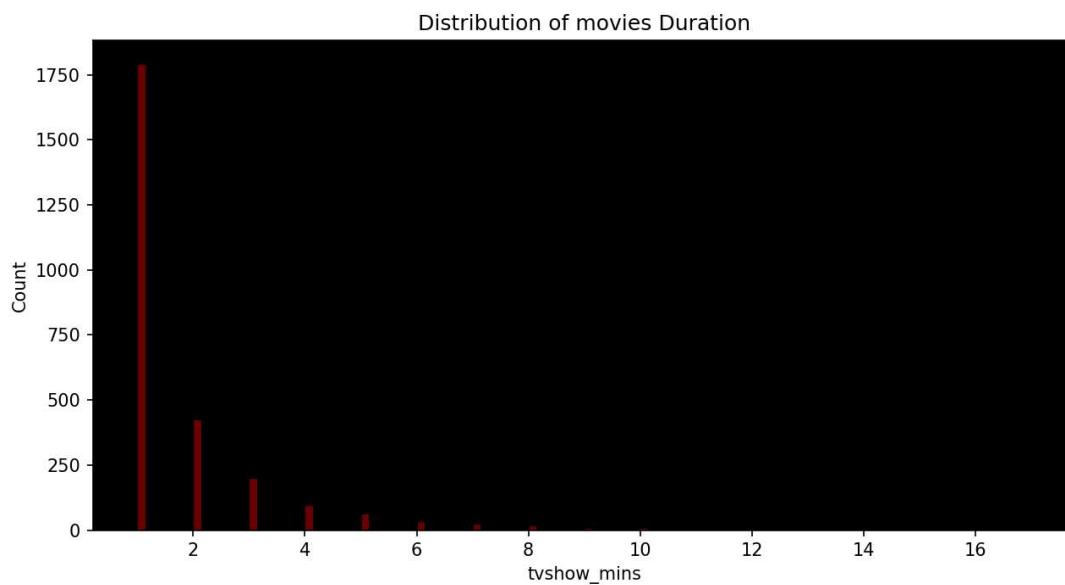


Here, we can observe the tendency of the Tv show which comes into the seasons which have a minimum of approx 1 season for the tv show and maximum is for approx 3 seasons.
Where some of the outliers are distributed which can come up with the 16 seasons for the particular show.

HISTPLOT AGAINST NUMBER OF SEASONS

```
In [419]: plt.figure(figsize=(10,5))
sns.histplot(data=df, x='tvshow_mins', color="darkred")
plt.title("Distribution of movies Duration")
ax = plt.gca()
ax.set_facecolor("Black")
plt.show()
```

<IPython.core.display.Javascript object>



The same observation has been came up with the Tv shows distribution duration channel with the hist plot comprises the maximum peak value as duration for the tv shows season is 1 and 2 seasons
 Business view: Although the least measure for the shows is the season 1, will get the desired output for the people watches more with the less season, inspite if the tv show would have been considered more likely when it comes to less season with the less episode which can get more positive result for the market rate of desired values for directors and casts

GROUPING DIRECTORS

```
In [420]: ⚡ distinct_director = df.groupby(by='director')['show_id'].count().sort_values(ascending = False)
distinct_director
```

```
Out[420]: director
Rajiv Chilaka                19
Raúl Campos, Jan Suter        18
Suhas Kadav                  16
Marcus Raboy                 16
Jay Karas                     14
..
Jos Humphrey                  1
Jose Gomez                     1
Jose Javier Reyes             1
Joseduardo Giordano, Sergio Goyri Jr. 1
Khaled Youssef               1
Name: show_id, Length: 4528, dtype: int64
```

TOP DIRECTORS WITH THEIR COUNT OF MOVIES

```
In [103]: ⚡ director_df = pd.DataFrame(df.groupby(by='director')['show_id'].count().sort_values(ascending = False)[1:])
director_df
dirct = director_df.reset_index()
director_top10 = dirct[:10]
```

```
In [118]: ⚡ director_top10.rename(columns={'show_id':'count'}, inplace=True)
director_top10
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_18708\287084270.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing.html#returning-a-view-versus-a-copy)
director_top10.rename(columns={'show_id':'count'}, inplace=True)
```

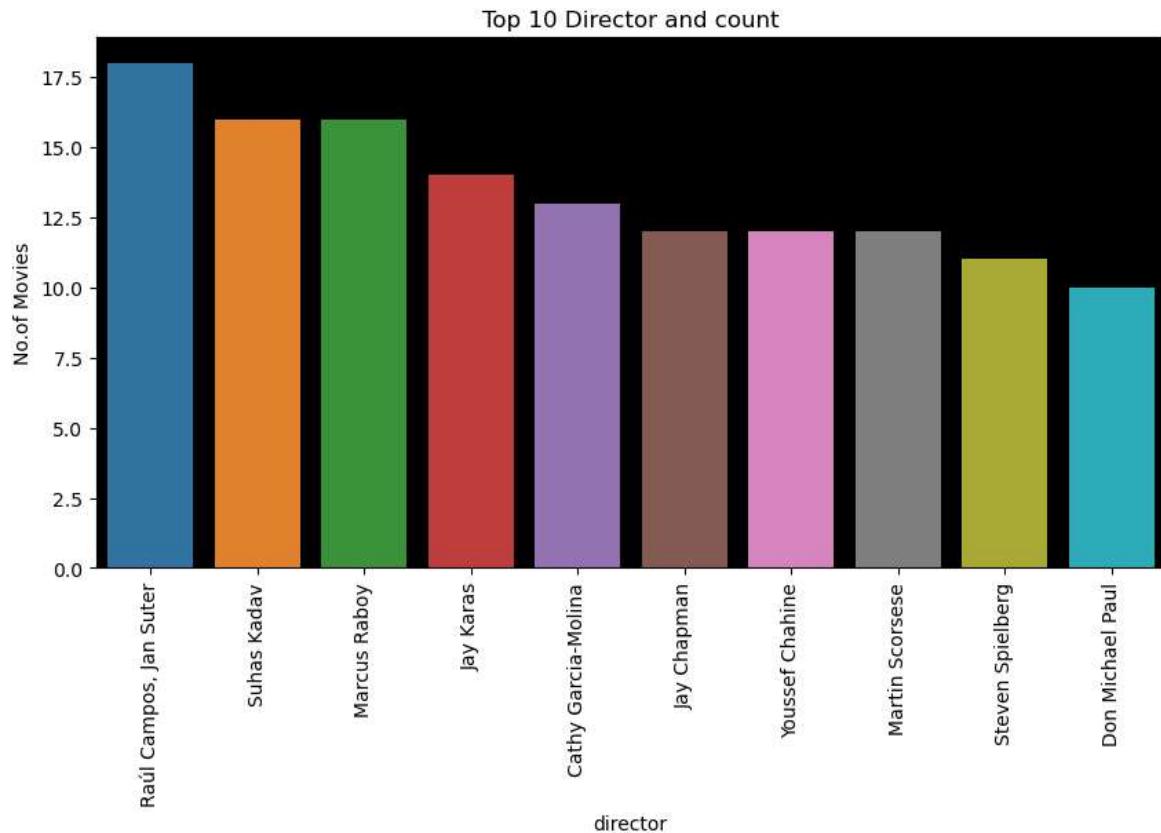
```
Out[118]:
```

	director	count
0	Raúl Campos, Jan Suter	18
1	Suhas Kadav	16
2	Marcus Raboy	16
3	Jay Karas	14
4	Cathy Garcia-Molina	13
5	Jay Chapman	12
6	Youssef Chahine	12
7	Martin Scorsese	12
8	Steven Spielberg	11
9	Don Michael Paul	10

This case scenario in non graphical system shows the top 10 rated directors from the data set having the number of movies and Tv shows where the top director "Raúl Campos, Jan Suter" who has created 18 movies as well as the lowest from the top 10 have the "Don Michael Paul" who made 10 Movies.
 Lets have the graphical representation for the above scene.

GRAPHICAL REPRESENTATION OF DIRECTOR MOVIES

```
In [114]: plt.figure(figsize=(10,5))
sns.barplot(data=director_top10, x='director', y='show_id')
plt.xticks(rotation = 90)
ax = plt.gca()
ax.set_facecolor("Black")
plt.ylabel("No.of Movies")
plt.title("Top 10 Director and count")
plt.show()
```



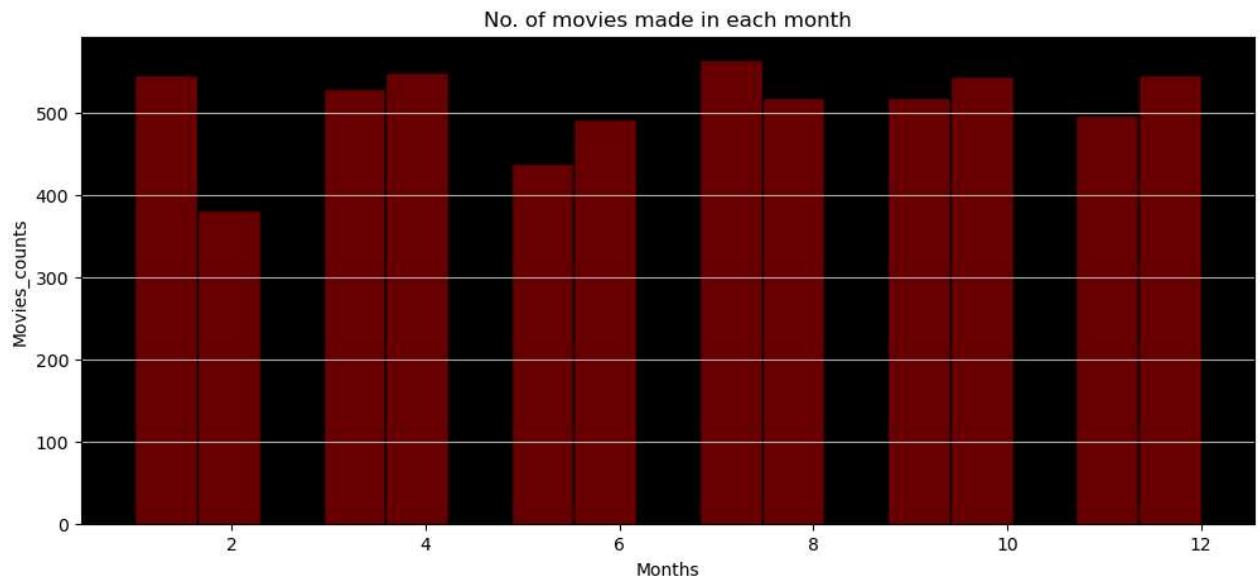
The Above case scenario shows the top rated directors from the Data set have given the top movies as the number of more movies which are the "Raúl Campos, Jan Suter" has given the top 18 movies to the industries.

```
In [24]: df = pd.read_csv('Netflix_case_studies.csv', parse_dates=['date_added'])
df.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	2021-09-25	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	2021-09-24	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	2021-09-24	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	2021-09-24	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to train l...

COUNT OF MOVIES DISTRIBUTION FOR EACH MONTH OR LAUNCHING DESIRE TIME

```
In [57]: movies = df[df['type']=='Movie']
movies_month = movies['date_added'].dt.month
plt.figure(figsize=(12,5))
sns.histplot(data=movies, x=movies_month, color="darkred")
ax=plt.gca()
ax.set_facecolor("black")
plt.grid(axis="y")
plt.xticks()
plt.xlabel("Months")
plt.ylabel("Movies_counts")
plt.title("No. of movies made in each month")
plt.show()
```



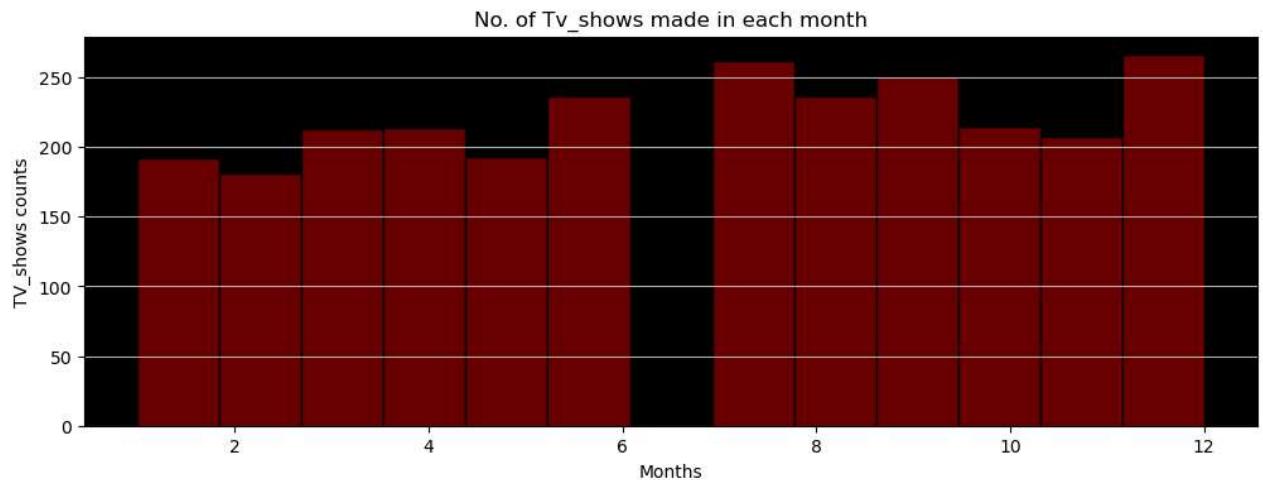
Picture, depicts the movie presents in each months! well here we can say the number of movie has been made in to the month of July as the maximum graph can be seen here, as well as the movies 🎬 has been made on feburay is too less from all others.

The graph depictst the most average date added movies in the months and vice-versa.

Business insights:Business works when the distribution and releasing falls in the right place, from the graph we have the month says aprila and july which have the major contribution as its peak for the good resposnse from the character of the movies, well the distribution signifies with the largest content behaviour for the movies.

COUNT OF TV SHOWS DISTRIBUTION FOR EACH MONTH LAUNCHING TIME

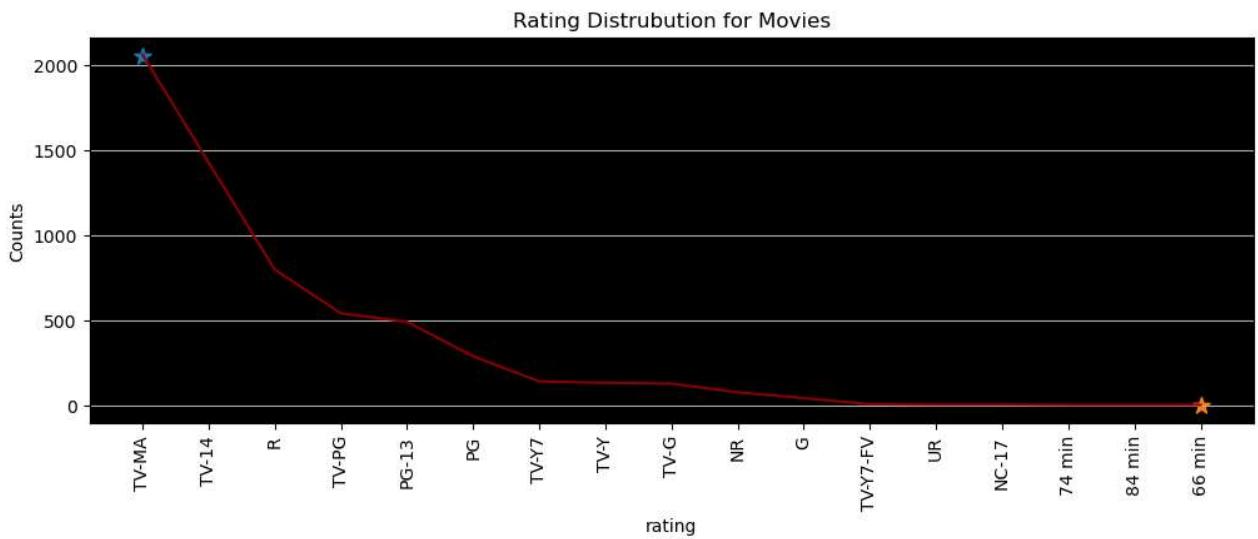
```
In [207]: tvshow = df[df['type']=='TV Show']
tvshow_month = tvshow['date_added'].dt.month
plt.figure(figsize=(12,4))
sns.histplot(data=tvshow, x=tvshow_month, color="darkred")
ax=plt.gca()
ax.set_facecolor("black")
plt.grid(axis="y")
plt.xticks()
plt.xlabel("Months")
plt.ylabel("TV_shows counts")
plt.title("No. of Tv_shows made in each month")
plt.show()
```



The hisplot graph shows the tv shows made or added in the dataset for all the months, where the most of the tv shows has been made from the month of the december, where the least tv shows 📺 made the in the month of the february, where the analysis says the most of the tv show has more than 250 shows added on the December.

RATING DISTRIBUTION TOWARDS MOVIES WITH THE COUNT

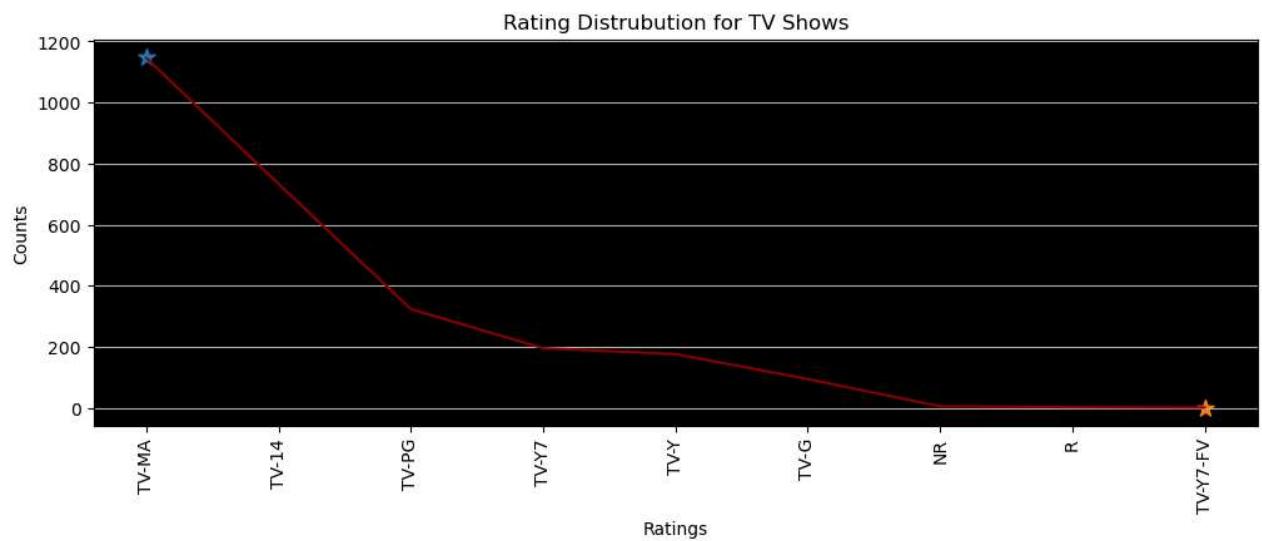
```
In [220]: movies_rating = df[df['type']=="Movie"]
final_mov_rat = movies_rating.groupby(by='rating')['show_id'].nunique().sort_values(ascending=False)
plt.figure(figsize=(12,4))
final_mov_rat = pd.DataFrame(final_mov_rat)
final_mov_rat.rename(columns={'show_id':'count'})
final_mov_rat.reset_index()
sns.lineplot(data=final_mov_rat, x='rating', y='show_id', color="darkred")
plt.xticks(rotation =90)
ax=plt.gca()
ax.set_facecolor("black")
plt.grid(axis="y")
plt.ylabel("Counts")
plt.title('Rating Distribution for Movies')
plt.scatter('TV-MA',2050,s=100, marker = "*")
plt.scatter('66 min',0,s=100, marker = "*")
plt.show()
```



The line graph image illustrates the distribution of movie ratings, with TV-MA having the highest ⚡ average score for films ratings overall and 66 Min having the lowest 🎬 rating platform where the ratings have been charted.

RATING DISTRIBUTION TOWARDS TV SHOWS WITH THE COUNT

```
In [247]: tvshow_rating = df[df['type']=="TV Show"]
final_tvshow_rat = tvshow_rating.groupby(by='rating')['show_id'].nunique().sort_values(ascending=False)
final_tvshow_rat = pd.DataFrame(final_tvshow_rat)
plt.figure(figsize=(12,4))
final_tvshow_rat.rename(columns={'show_id':'count'})
final_tvshow_rat.reset_index()
sns.lineplot(data=final_tvshow_rat, x='rating', y='show_id', color="darkred")
plt.xticks(rotation =90)
ax=plt.gca()
ax.set_facecolor("black")
plt.grid(axis="y")
plt.ylabel("Counts")
plt.xlabel("Ratings")
plt.title('Rating Distribution for TV Shows')
plt.scatter(x="TV-MA",y=1150,s=100,marker="*")
plt.scatter(x="TV-Y7-FV",y=0,s=100,marker="*")
plt.show()
```



Here are the top 10 movie genres that have been produced and are most likely to be watched by viewers.
The most popular inspiration for creating such a scene is drama, a genre in which the majority of popular films have been produced.

```
In [2]: df = pd.read_csv('Netflix_case_studies.csv')
df.head(2)
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	Nan	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	Nan	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...

TOP 10 CAST MAJOR WORKED IN MOVIES

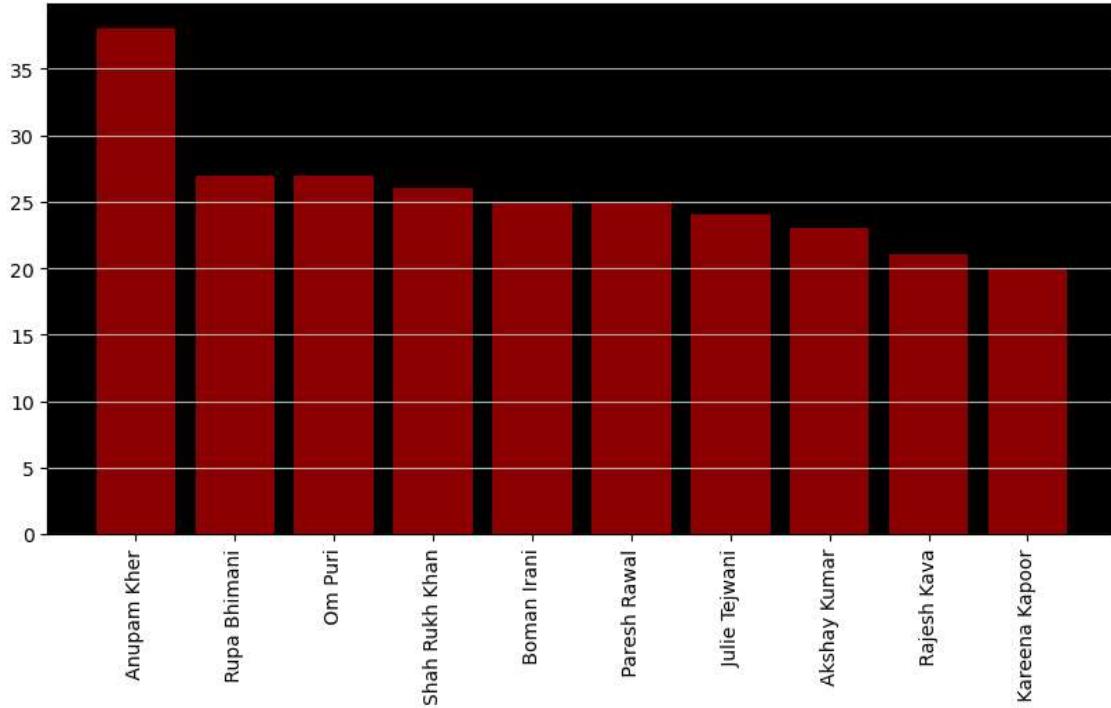
```
In [20]: ⚡ movies = df[df['type'] == "Movie"]
movies_cast = movies['cast'].str.split(", ", expand=True).stack().value_counts()[:10]
movies_cast
```

```
Out[20]:   Anupam Kher      38
             Rupa Bhimani    27
              Om Puri        27
            Shah Rukh Khan    26
            Boman Irani       25
           Paresh Rawal       25
            Julie Tejwani      24
          Akshay Kumar        23
            Rajesh Kava        21
          Kareena Kapoor       20
dtype: int64
```

The Above scenario shows the major concern with the Movies in respect to the cast with the number of the movies count the cast has been made. Furthur the nature of the Analysis shows the from the category movies, the top 10 cast are being worked with the major number of time in whole duration.

Business Driven: There are the top cast actor and actress who have played a major of the roles in the movies which tends to be more likely and positive driven, the people are liking more these cast which shows their movies tends be hit more positive

```
In [37]: ⚡ plt.figure(figsize=(10,5))
plt.bar(movies_cast.index, movies_cast.values, color="darkred")
ax=plt.gca()
ax.set_facecolor("black")
plt.xticks(rotation = 90)
plt.grid(axis="y")
plt.show()
```



The graph is for Analysing the data for the top of the casting character with their names and number of movies they hold for the same. Pictorial representation for the data analysis on the most side of the character of the data set.

TOP 10 CAST MAJOR WORKED IN TV SHOWS

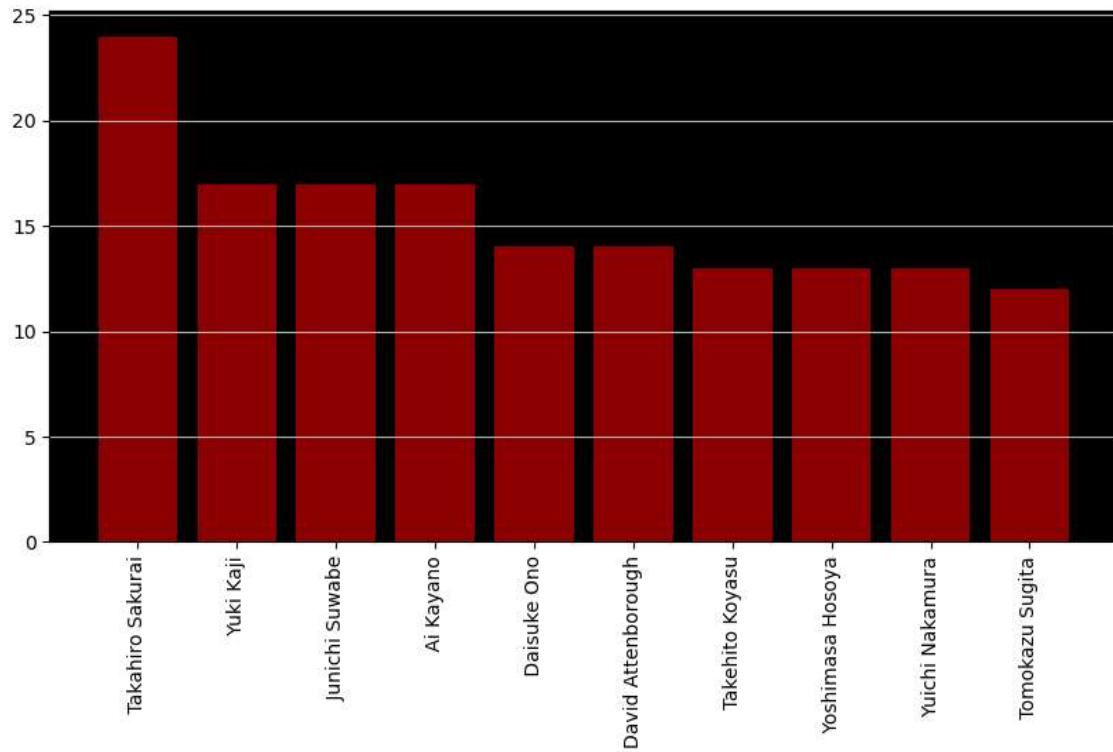
```
In [40]: tv_shows = df[df['type'] == "TV Show"]
tv_shows = tv_shows['cast'].str.split(", ", expand=True).stack().value_counts()[:10]
tv_shows
```

```
Out[40]:   Takahiro Sakurai      24
             Yuki Kaji          17
             Junichi Suwabe       17
             Ai Kayano           17
             Daisuke Ono          14
             David Attenborough    14
             Takehito Koyasu        13
             Yoshimasa Hosoya       13
             Yuichi Nakamura        13
             Tomokazu Sugita         12
dtype: int64
```

Here are the top 10 cast members who excelled in TV shows, based on their major works. This list features their outstanding performances and contributions to the industry.

The Above outcome is non graphical analysis with major concern with the popularity of the cast

```
In [48]: plt.figure(figsize=(10,5))
plt.bar(tv_shows.index, tv_shows.values, color= "darkred")
ax= plt.gca()
ax.set_facecolor("black")
plt.grid(axis="y")
plt.xticks(rotation = 90)
plt.show()
```



Here are the top 10 cast members who excelled in TV shows, based on their major works. This list features their outstanding performances and contributions to the industry. With the pictorial graph representation it allows to distinguish the matter into the considerable outcome. The Above outcome is non analysis with major concern with the popularity of the cast

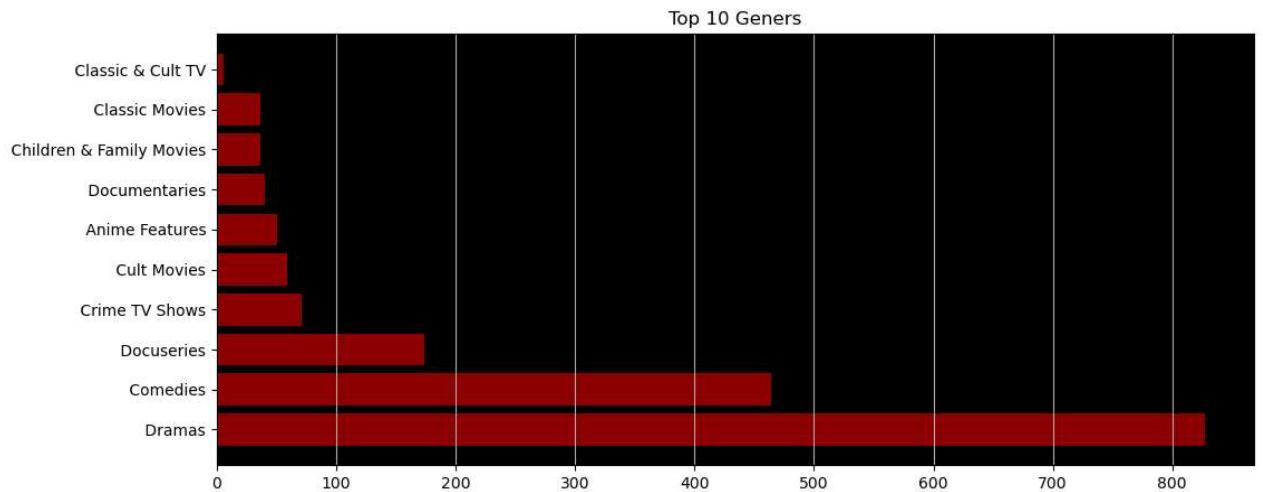
Business view: Tv shows tends to be active while getting into the deep insights with the cast and the character, the people who wants to look for the cast who will says as best for the people liking is the top 10 under the TV show.

TOP 10 GENRES MOST PRODUCED

```
In [62]: genre = df['listed_in'].str.split(", ",expand=True).stack().to_frame().groupby(by=0).value_counts()[:10].sort_values(ascending=False)
```

```
Out[62]: 0
          Dramas      827
          Comedies    464
          Docuseries   174
          Crime TV Shows 71
          Cult Movies   59
          Anime Features 50
          Documentaries 40
          Children & Family Movies 36
          Classic Movies 36
          Classic & Cult TV     6
          dtype: int64
```

```
In [85]: plt.figure(figsize=(12,5))
plt.barh(genre.index, genre.values, color="darkred")
ax=plt.gca()
ax.set_facecolor("black")
plt.title("Top 10 Generes")
plt.grid(axis="x")
plt.show()
```



The graph represents the highest liked and most created genre type from the graph!

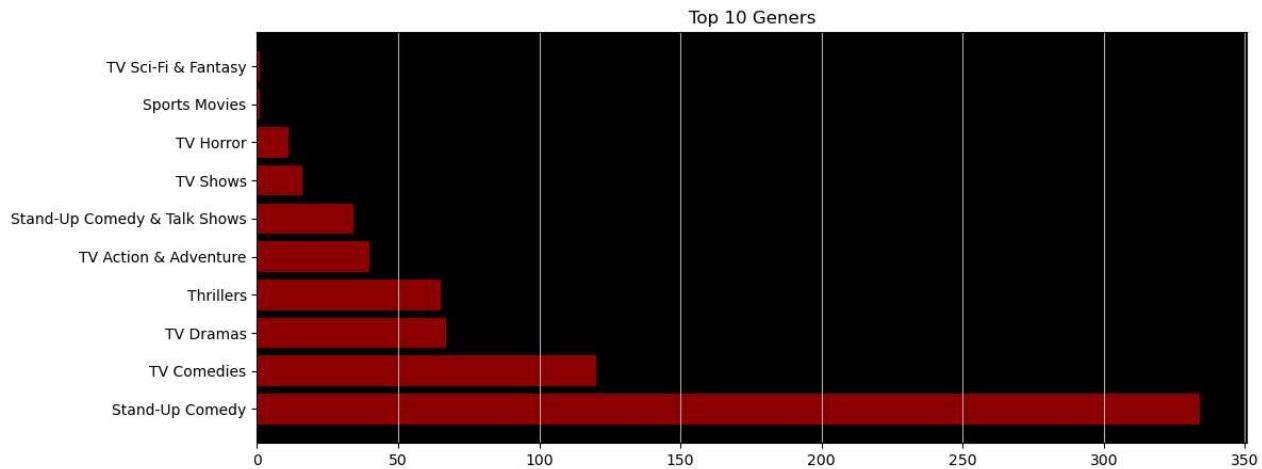
Business insight: The most likes and expanded during the world behaviour is the Drama, where the people and director have purely linked seen which it comes into the genre type, so the attracting and benefits comes form the genre is the top 10 genre where the business hits at highest rate of interest.

BOTTOM 10 GENRES WITH MOST AND LEAST PRODUCED

```
In [80]: genreb = df['listed_in'].str.split(", ",expand=True).stack().to_frame().groupby(by=0).value_counts()[-10:].sort_values(ascending=False)
```

```
Out[80]: 0
          Stand-Up Comedy    334
          TV Comedies        120
          TV Dramas          67
          Thrillers           65
          TV Action & Adventure 40
          Stand-Up Comedy & Talk Shows 34
          TV Shows            16
          TV Horror            11
          Sports Movies         1
          TV Sci-Fi & Fantasy     1
          dtype: int64
```

```
In [83]: plt.figure(figsize=(12,5))
plt.barh(genreb.index, genreb.values, color="darkred")
ax=plt.gca()
ax.set_facecolor("black")
plt.title("Top 10 Generes")
plt.grid(axis="x")
plt.show()
```



According to our analysis, stand-up comedy and TV sci-fi are the genres of films that are most unlikely to appeal to viewers and have the least amount of content. These types of films are also those that are most frequently in the queue for viewing.

Business insight: The most least likes and expanded during the world behaviour is the stand up comedy, where the people and director have purely not makes link which has been seen which it comes into the genere type, so the attracting and benefits comes form the genere is the top 10 genere which is least where the business hits at highest rate of interest at fall back.

WORD CLOUD FOR MOST PROMINENT AND FREQUENT WORDS

```
In [422]: ┆ from wordcloud import WordCloud, STOPWORDS
#for plotting what we have to not give in the context

comment_words = ''
stopwords = set(STOPWORDS)
stopwords = ['nan', 'NaN', 'Nan', 'NAN'] + list(STOPWORDS)

#titles from the dataset

values = df['title'].values

#iterating through all the dataset titles
for val in values:
    val = str(val)
    tokens = val.split()
    for i in range(len(tokens)):
        tokens[i] = tokens[i].lower()
    comment_words += ' '.join(tokens) + ' '

#design part
facecolor = 'black'

wordcloud = WordCloud(width=2000, height=600,
                      background_color=facecolor,
                      stopwords=stopwords,
                      min_font_size=5).generate(comment_words)

#libraries part

plt.figure(figsize=(10,4), facecolor=facecolor)
plt.imshow(wordcloud)
plt.axis('off')
plt.tight_layout(pad=1)
plt.show()
```

<IPython.core.display.Javascript object>



Word clouds provide a new perspective on the material being read and aid visual learners in understanding reading activities in a way they may not be used to.

This graph is known as WordCloud where we can show the list of the items in terms of names of the movies or shows or any kind og mix words where we can find attractiue way to see and guess the words for the movioes to remember or to see the related words to see.