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

df=pd.read_csv('/content/drive/MyDrive/Colab Notebooks/original_netflix.csv')
df.head(10)
 #For stacking Cast columns with title
constraint=df['cast'].apply(lambda x: str(x).split(', ')).tolist()
df_new=pd.DataFrame(constraint,index=df['title'])
df_new=df_new.stack().reset_index()
df_new=df_new[['title',0]]
df_new.columns=['title','cast']
df_new['cast'].replace(to_replace="nan",
           value='Unknown',inplace=True)



 #For stacking director columns with title

df.head()
c=df['director'].apply(lambda x: str(x).split(', ')).tolist()
df1=pd.DataFrame(c,index=df['title'])
df1=df1.stack().reset_index()
df1=df1[['title',0]]
df1.columns=['title','director']
df1['director'].replace(to_replace="nan",
           value='Unknown',inplace=True)

 #filling the nan values with the most occuring elements in the director column

  #For stacking Genre columns with title
df.head()
d=df['listed_in'].apply(lambda x: str(x).split(', ')).tolist()
df2=pd.DataFrame(d,index=df['title'])
df2=df2.stack().reset_index()
df2=df2[['title',0]]
df2.columns=['title','listed_in']
df2['listed_in'].isnull().sum()  #checking null values

#For stacking Country columns with title
e=df['country'].apply(lambda x: str(x).split(', ')).tolist()
df3=pd.DataFrame(e,index=df['title'])
df3=df3.stack().reset_index()
df3=df3[['title',0]]
df3.columns=['title','country']
df3['country'].replace(to_replace="nan",
           value=df3['country'].mode()[0],inplace=True)


 #Merging all the data frame into a single one

d=df_new.merge(df1,how='inner',on='title')
e=d.merge(df2,how='inner',on='title')
df_updated=e.merge(df3,how='inner',on='title')
df_updated.head(10)
#Now merging data frame with the original data frame
df_final=df_updated.merge(df,on='title')
df_final.drop(['director_y','cast_y','country_y','listed_in_y'],axis=1,inplace=True)
df_final.rename(columns={'director_x': 'director',
                        'cast_x': 'cast',
                        'country_x': 'country','listed_in_x':'listed_in'},inplace=True)
df_final.shape  #checking the new size after cleaning,transforming and merging
#removing null values from the updated data frame
df_final.loc[df_final['rating'].isnull()] #checking null values in rating column
df_final['rating'].fillna(df_final['rating'].mode()[0], inplace=True) #filling null values with mode
df_final.loc[df_final['date_added'].isnull()] #filling null values in date added column with mode
df_final['date_added'].fillna(df_final['date_added'].mode()[0], inplace=True) #filling null values with mode
df_final['duration'].fillna(df_final['duration'].mode()[0], inplace=True) #filling null values with mode


#Performing Exploratory Data Analysis on the final data set after cleaning

df_final.describe()
#From the above command we can conclude that maximum percentage of the movies released in the year 2019
df_pop=df_final.groupby(['cast'],as_index=False)['title'].nunique().sort_values(by='title',ascending=False)#from this command Anupam Kher is
```

```
df_pop
```

	cast	title
34214	Unknown	825
2833	Anupam Kher	43
30489	Shah Rukh Khan	35
16697	Julie Tejwani	33
24215	Naseeruddin Shah	32
...	...	...
14221	Jamie Lee	1
14219	Jamie Kenna	1
14218	Jamie Kaler	1
14217	Jamie Johnston	1
36439	Ṣọpẹ́ Dìrísù	1

36440 rows × 2 columns

```
#Which Type is more popular:Tv shows or Movies?
df_type=df_final.groupby(['type'],as_index=False)['type'].value_counts().sort_values(by='type',ascending=False)
df_type
```

	type	count
1	TV Show	56148
0	Movie	145843

```
ax = sns.countplot(x='rating', data=df_final,hue='type')
plt.figure(figsize=(20, 12))

ax.set_xticklabels(ax.get_xticklabels(), rotation=45)  # Adjust the rotation angle (45) as needed
plt.show()
#Movie rating based on type(tv show or movies): from this we can conclude that TV-MA rating for movies type is highest
```

```
#Analysis of actors/directors of different types of shows/movies
df_final.head()
#Analysis of actors/directors of different types of shows/movies for eg: getting all the movies where cast is 'Shahrukh khan'
df_pop=df_final[['cast','title']]
df_pop[df_pop['cast']=='Shah Rukh Khan']
```

	cast	title
2491	Shah Rukh Khan	Anjaam
2492	Shah Rukh Khan	Anjaam
2493	Shah Rukh Khan	Anjaam
7327	Shah Rukh Khan	Chennai Express
7328	Shah Rukh Khan	Chennai Express
...	...	...
181970	Shah Rukh Khan	Shakti: The Power
181971	Shah Rukh Khan	Shakti: The Power
197883	Shah Rukh Khan	Trimurti
197884	Shah Rukh Khan	Trimurti
197885	Shah Rukh Khan	Trimurti

108 rows × 2 columns

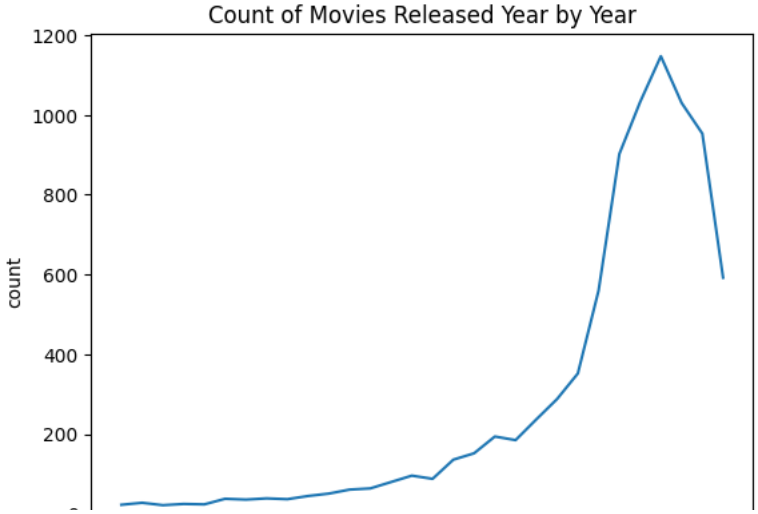
```
#who is the most popular actor-director pair?
df_acd=df_final.groupby(['cast','director']).size().reset_index(name='popularity').sort_values(by='popularity',ascending=False)
df_acd.head(10)
#Top 10 popular actor director pairs are given below:
```

	cast	director	popularity
59162	Unknown	Unknown	738
13508	David Attenborough	Unknown	72
55848	Takahiro Sakurai	Unknown	54
61867	Yuki Kaji	Unknown	43
28907	Jun Fukuyama	Unknown	38
61834	Yuichi Nakamura	Unknown	38
28993	Junichi Suwabe	Unknown	37
29930	Kate Harbour	Unknown	37
1186	Ai Kayano	Unknown	37
12529	Daisuke Ono	Unknown	36

```
#How has the number of movies released per year changed over the last 20-30 years?
df_yr=df_final.groupby('release_year')['title'].nunique().reset_index(name='count').sort_values(by='release_year')
df_yr
#Lets analyze the trend over the last 30 years?
dfyr1=df_yr.tail(30)
```

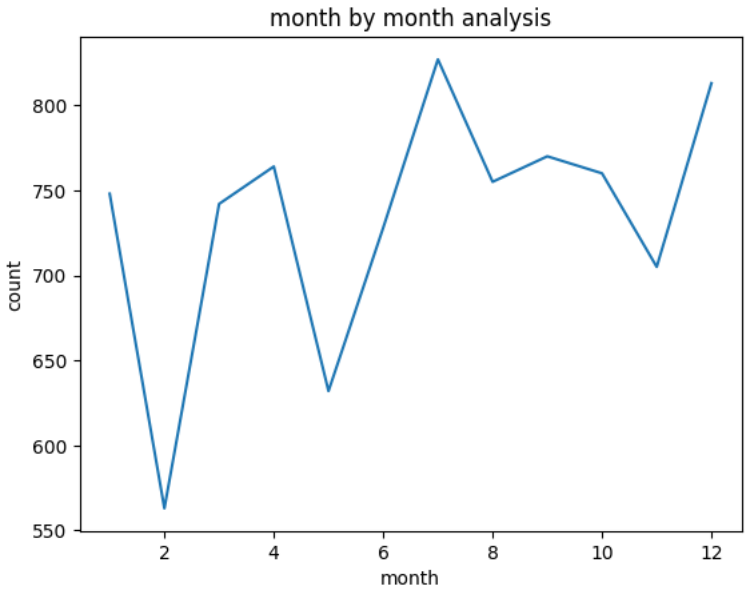
```
#Plotting the trend using line chart

plt.plot(dfyr1['release_year'], dfyr1['count'])
plt.xlabel('release_year')
plt.ylabel('count')
plt.title('Count of Movies Released Year by Year')
plt.show()
#From the below graph we can analyze that the movies released year by year is increasing.
```




```
#What is the best time to launch a TV show or a movie?
df_final['date_added']=pd.to_datetime(df_final['date_added'])
get_month = lambda x: x.month
df_final['Month'] = df_final['date_added'].apply(get_month)
df_t=df_final.groupby('Month')['title'].nunique().reset_index(name='count').sort_values(by='Month')

plt.plot(df_t['Month'], df_t['count'])
plt.xlabel('month')
plt.ylabel('count')
plt.title('month by month analysis')
plt.show()
#From this analysis we can conclude that the best release time to launch a movie or a tv show is between september to december(estimated)
```




```
#Understanding what content is available in different countries?
df_c=df_final.groupby(['country'])['listed_in'].unique().reset_index(name='Content Available')
df_c.head(10)
```

	country	Content Available
0		[International TV Shows, TV Dramas, Dramas, In...
1	Afghanistan	[Documentaries, International Movies]
2	Albania	[Dramas, International Movies]
3	Algeria	[Dramas, Independent Movies, International Mov...
4	Angola	[Action & Adventure, International Movies]

```
#let us find what type of content is available in India:
df_c[df_c['country']=='India'].values
```

```
array([['India',
        array(['International TV Shows', 'Romantic TV Shows', 'TV Comedies',
               'Comedies', 'International Movies', 'Romantic Movies', 'Thrillers',
               "Kids' TV", 'TV Dramas', 'TV Sci-Fi & Fantasy', 'Docuseries',
               'Action & Adventure', 'Dramas', 'Independent Movies',
               'Horror Movies', 'Music & Musicals', 'Sci-Fi & Fantasy',
               'TV Shows', 'Children & Family Movies', 'Reality TV',
               'Documentaries', 'Sports Movies', 'Stand-Up Comedy',
               'British TV Shows', 'Cult Movies', 'TV Action & Adventure',
               'TV Horror', 'Crime TV Shows', 'Stand-Up Comedy & Talk Shows',
               'Classic Movies', 'TV Mysteries', 'Teen TV Shows', 'TV Thrillers',
               'LGBTQ Movies', 'Faith & Spirituality'], dtype=object)             ]],
      dtype=object)
```

```
#Who is the most popular director in india?
df_d=df_final[df_final['country']=='India']
df_d.groupby(['director'],as_index=False)['title'].nunique().sort_values(by='title',ascending=False)
```

	director	title
684	Unknown	85
151	David Dhawan	9
80	Anurag Kashyap	9
682	Umesh Mehra	8
168	Dibakar Banerjee	7
...	...	...
341	Musthafa	1
95	Ashim Ahluwalia	1
343	N. Chandra	1
94	Arvind Swamy	1
368	Nikhil Nagesh Bhat	1

736 rows × 2 columns

```
#Identifying the Top 5 genres of content on Netflix
df_g=df_final.groupby(['listed_in']).size().reset_index(name='popularity').sort_values(by='popularity',ascending=False)
df_g.head(5)
#From the below table,Dramas,International Movies,comedies,Tv shows,Action and Adevnture are the most popular genre across the world
```

	listed_in	popularity
12	Dramas	29775
16	International Movies	28211

```
df_final.describe()
```

	release_year	Month
count	201991.000000	201991.000000
mean	2013.452891	6.631909
std	9.003933	3.444674
min	1925.000000	1.000000
25%	2012.000000	4.000000
50%	2016.000000	7.000000
75%	2019.000000	10.000000
max	2021.000000	12.000000

```
#Outlier Check
y=df_final['date_added'].dt.year
sns.boxplot(y)
#In the below graph, can clearly see that values below 2015  are acting as outliers.
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-2-a7270dcc8954> in <cell line: 2>()
      1 #Outlier Check
----> 2 y=df_final['date_added'].dt.year
      3 sns.boxplot(y)
      4 #In the below graph, can clearly see that values below 2015  are acting as outliers.
      5

NameError: name 'df_final' is not defined
```

SEARCH STACK OVERFLOW

```
#Comments on the range of attributes:
df_final.shape #(the final data set after transforming and cleaning has 201991 rows and 13 rows)
```

```
(201991, 13)
```

```
df_final.info()
#Most of the attributes are of object type except date_added and release_year with no null entries.All the null values are detected and recti
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 201991 entries, 0 to 201990
Data columns (total 13 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   title         201991 non-null  object
 1   cast          201991 non-null  object
 2   director      201991 non-null  object
 3   listed_in     201991 non-null  object
 4   country       201991 non-null  object
 5   show_id       201991 non-null  object
 6   type          201991 non-null  object
 7   date_added    201991 non-null  datetime64[ns]
 8   release_year  201991 non-null  int64
 9   rating        201991 non-null  object
 10  duration      201991 non-null  object
```

```
 11  description   201991 non-null  object
 12  Month         201991 non-null  int64
dtypes: datetime64[ns](1), int64(2), object(10)
memory usage: 29.6+ MB
```

```
df_final.size
#total no of elements in the dataframe
```

```
2625883
```

```
df_final.ndim #the datset is of 2 dimensional(Dataframe)
```

```
2
```

```
#Business insights:
"""
By performing EDA of the final dataset we can analyze various trends in our data in every aspect and came out with the following:
1.From the statistical analysis we can analyze that 75%  of the movies released in netflix by the year 2019.
2.The first movie was released in the year 1925 and Anupam Kher is the most popular actor having 43 Titles.
3.Movies gor maximum rating of type 'TV-MA' and Tv shows of the same type
4.After analysing movies count year by year we can see that production of movies is increasing year by year in netflix.
5.Anakysing month by month the peak time for any movie release is between september to december.
6.The most popular actor in india is found out to be david Dhawan with movies count of 9.
7.The most popular genres in Netflix is found out to be Dramas,International Movies,comedies,Tv shows,Action and Adventure.
8.From the outlier Check using Box plot the movies released before 2015 are founded out to be outliers.
"""
```

```
'\nBy performing EDA of the final dataset we can analyze various trends in our data in every aspect an
d came out with the following:\n1.From the statistical analysis we can analyze that 75%  of the movies
released in netflix by the year 2019.\n2.The first movie was released in the year 1925 and Anupam Kher
is the most popular actor having 43 Titles.\n3.Movies gor maximum rating of type 'TV-MA' and Tv shows
of the same type\n4.After analysing movies count year by year we can see that production of movies is
increasing year by year in netflix.\n5.Anakysing month by month the peak time for any movie release is
between september to december.\n6.The most popular actor in india is found out to be david Dhawan with
```

```
#Recommendations:
1.From the above analysis the content of  movies or tv series with high rating should be focused more
2.The major issue of the dataset is max of Nan values in each column so if we use the mode operator our insights and data prediction may go w
3.The peak time for any series released in between September to December...so the movies releasing between these time interval should be focu
4.Some of the contents of netlix are not available in many contries...to make people aware of the present scenario of the world more no of mo
5.The movie  or TV show that got a rating of TV-MA should be broadcasted to a wider audience.
```

Colab paid products - Cancel contracts here
✓ 3s completed at 11:49 AM