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

df=pd.read_csv('mymoviedb (1).csv',lineterminator='\n')
df.sample(3)
```

```
     Release_Date                             Title  \
104    2021-04-22                      Wrath of Man   
2099   2021-08-19                      Reminiscence   
1399   2014-02-13  Tinker Bell and the Pirate Fairy   

                                               Overview  Popularity  \
104   A cold and mysterious new security guard for a...     273.622   
2099  Nicolas Bannister, a rugged and solitary veter...      39.419   
1399  Zarina, a smart and ambitious dust-keeper fair...      52.890   

      Vote_Count  Vote_Average Original_Language
Genre  \
104         3376           7.7                en   Action, Crime,
Thriller   
2099        1090           6.9                en  Science Fiction,
Mystery   
1399         777           6.8                en         Animation,
Family   

                                             Poster_Url  
104   https://image.tmdb.org/t/p/original/M7SUK85sKj...  
2099  https://image.tmdb.org/t/p/original/17siH6wJRQ...  
1399  https://image.tmdb.org/t/p/original/qZLBe9Z8Y6...  
```

```python
#view the dataset info
print('information of datastet:\n',df.info())

#chackinjg any duplicated rows
df.duplicated().sum() # no duplicate rows is present
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9827 entries, 0 to 9826
Data columns (total 9 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Release_Date       9827 non-null   object 
 1   Title              9827 non-null   object 
 2   Overview           9827 non-null   object 
 3   Popularity         9827 non-null   float64
 4   Vote_Count         9827 non-null   int64  
 5   Vote_Average       9827 non-null   float64
 6   Original_Language  9827 non-null   object 
 7   Genre              9827 non-null   object
 8   Poster_Url         9827 non-null   object
dtypes: float64(2), int64(1), object(6)
memory usage: 691.1+ KB
information of datastet:
 None

0

# desription
df.describe()

        Popularity    Vote_Count  Vote_Average
count  9827.000000   9827.000000   9827.000000
mean     40.326088   1392.805536      6.439534
std     108.873998   2611.206907      1.129759
min      13.354000      0.000000      0.000000
25%      16.128500    146.000000      5.900000
50%      21.199000    444.000000      6.500000
75%      35.191500   1376.000000      7.100000
max    5083.954000  31077.000000     10.000000
```

Data cleaning

```
# extracting year,month and day from 'Releae_Date' column
df['Release_Date']=pd.to_datetime(df['Release_Date'])   #
'Release_date' is of string types , it is converted to datetime
datatype

df['Release_year']=df['Release_Date'].dt.year # Extracting the year
df['Release_month']=df['Release_Date'].dt.month # Extracting month
df['Release_day']=df['Release_Date'].dt.day # extracting the day of
the month
df['Release_dow_name']=df['Release_Date'].dt.day_name()   # extracting
the day name of the week
df['Release_is_weekend']=np.where(df['Release_dow_name'].isin(['Sunda
y','Saturday']),1,0)
df.drop(columns=['Release_Date'])
df.sample(5)

     Release_Date                            Title  \
1318   2014-03-07                             Noah
2799   1973-12-01                 Fantastic Planet
3458   2019-05-04                 47 Hours to Live
3844   2014-02-05  Jack and the Cuckoo-Clock Heart
1454   2013-08-07                          Elysium

                                               Overview  Popularity  \
1318  A man who suffers visions of an apocalyptic de...      55.213
```

```
2799  On the planet Ygam, the Draags, extremely tech...      31.790
3458  Two socially awkward teenage girls, are bored ...      27.174
3844  In Scotland 1874, Jack is born on the coldest ...      25.178
1454  In the year 2159, two classes of people exist:...      51.533

      Vote_Count  Vote_Average Original_Language  \
1318        5292           5.6                en
2799         643           7.7                fr
3458          43           6.2                en
3844         570           7.2                fr
1454        7365           6.5                en

                                                Genre  \
1318                                Drama, Adventure
2799                      Animation, Science Fiction
3458                                Thriller, Horror
3844  Animation, Romance, Adventure, Drama, Fantasy
1454       Science Fiction, Action, Drama, Thriller

                                             Poster_Url  Release_year
\
1318  https://image.tmdb.org/t/p/original/trtD17IqSW...          2014

2799  https://image.tmdb.org/t/p/original/prq0j1S0K0...          1973

3458  https://image.tmdb.org/t/p/original/x2iAQLgwvd...          2019

3844  https://image.tmdb.org/t/p/original/ZSrU2mvlzM...          2014

1454  https://image.tmdb.org/t/p/original/aRjuJuPXHt...          2013


      Release_month  Release_day Release_dow_name  Release_is_weekened

1318              3            7           Friday                    0

2799             12            1         Saturday                    1

3458              5            4         Saturday                    1

3844              2            5        Wednesday                    0

1454              8            7        Wednesday                    0
```

Dropping the unwanted column

```
# dropping 'overview','Poster_Url' column
df=df.drop(columns=['Overview','Poster_Url'])
df.sample(3)
```

```
     Release_Date                            Title  Popularity  
Vote_Count  \
6267   1985-07-03                  Day of the Dead      18.003   
900   
2149   2003-06-27  Charlie's Angels: Full Throttle      38.715   
2508   
371    2011-04-21                             Thor     129.237   
17866  

      Vote_Average Original_Language                        Genre  \
6267           7.0                en            Horror, Thriller   
2149           5.4                en   Action, Adventure, Comedy   
371            6.8                en  Adventure, Fantasy, Action   

      Release_year  Release_month  Release_day Release_dow_name  \
6267          1985              7            3        Wednesday   
2149          2003              6           27           Friday   
371           2011              4           21         Thursday   

      Release_is_weekened  
6267                    0  
2149                    0  
371                     0  
```

- categorizing Vote_Average column

```
def categorize_col (df,col,labels):
    edges=[df[col].describe()['min'],
          df[col].describe()['25%'],
          df[col].describe()['50%'],
          df[col].describe()['75%'],
          df[col].describe()['max']]
    df[col]=pd.cut(df[col],edges,labels=labels,duplicates='drop')
    return df

labels=['not_popular','below_avg','average','popular']
categorize_col(df,'Vote_Average',labels)
df['Vote_Average'].unique()
```

```
['popular', 'below_avg', 'average', 'not_popular', NaN]
Categories (4, object): ['not_popular' < 'below_avg' < 'average' < 
'popular']
```

```
# exploring the column
print(df['Vote_Average'].value_counts())
df.head(3)
df.dropna(inplace=True)  # removing the missing values
df.isnull().sum()
# Now no missing values are present

Vote_Average
not_popular    2467
popular        2450
average        2412
below_avg      2398
Name: count, dtype: int64

Release_Date          0
Title                 0
Popularity            0
Vote_Count            0
Vote_Average          0
Original_Language     0
Genre                 0
Release_year          0
Release_month         0
Release_day           0
Release_dow_name      0
Release_is_weekened   0
dtype: int64
```

- working on 'Genre' column

```
# extract the part of the string before the comma
df['Genre']=df['Genre'].str.split(', ')

df=df.explode('Genre').reset_index(drop=True)
df.sample(2)

#casting column into category
df['Genre']=df['Genre'].astype('category')
df['Genre'].dtypes

df.nunique()

Release_Date          5846
Title                 9415
Popularity            8088
Vote_Count            3265
Vote_Average             4
Original_Language       42
Genre                   19
Release_year           100
```

```
Release_month             12
Release_day               31
Release_dow_name           7
Release_is_weekened        2
dtype: int64
```

```
df.shape
```

```
(25552, 12)
```

data visualization

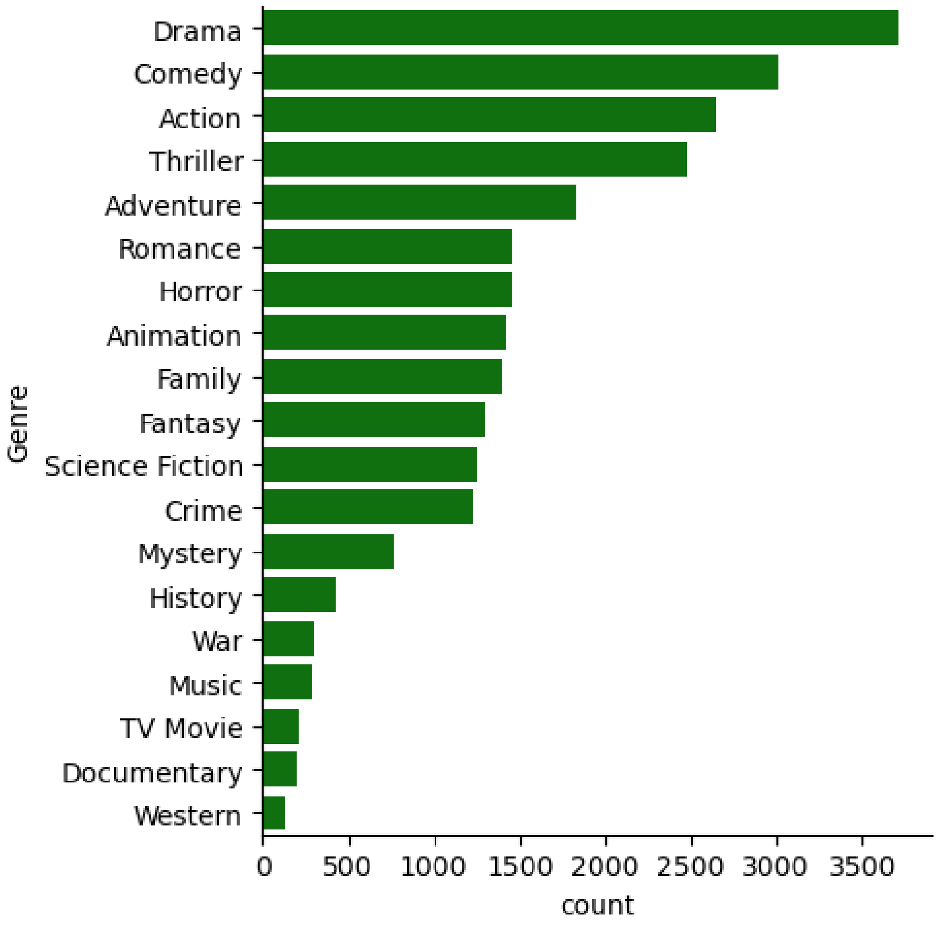
- most frequent genre

```
df['Genre'].describe()
```

```
count     25552
unique       19
top       Drama
freq       3715
Name: Genre, dtype: object
```

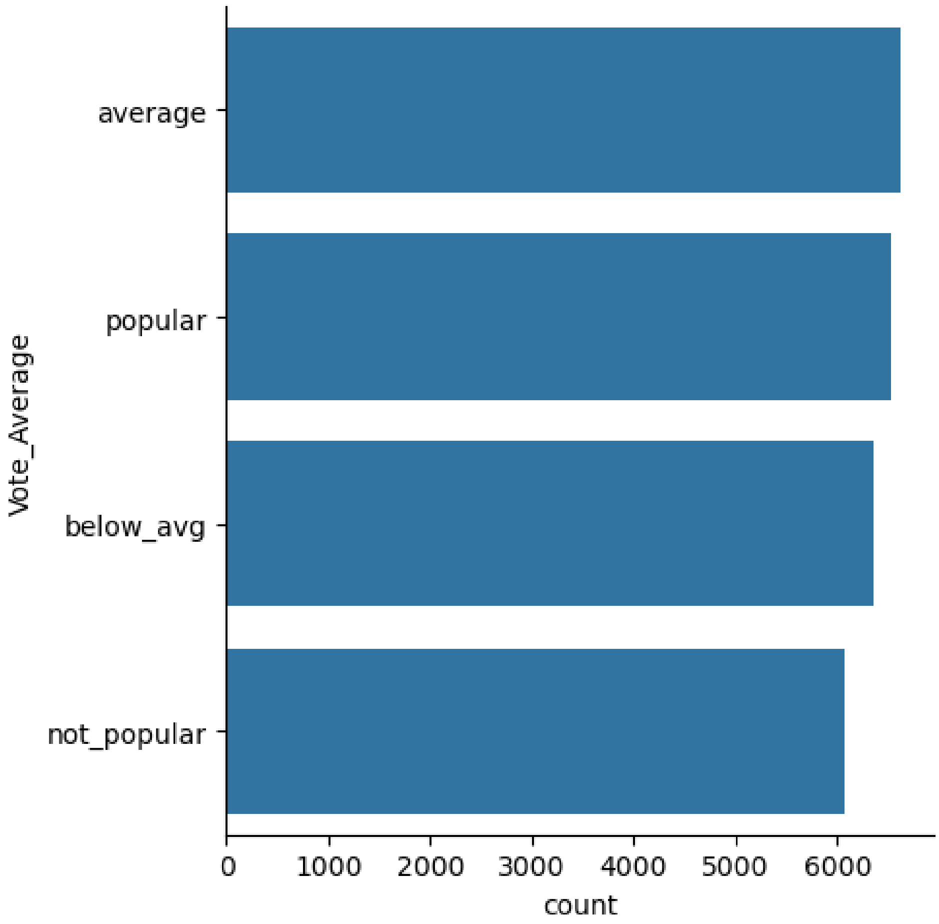
```
sns.catplot(data=df,y='Genre',kind='count',order=df['Genre'].value_cou
nts().index,color='green')
```

```
<seaborn.axisgrid.FacetGrid at 0x290727018b0>
```



- genre which has highest votes

```
# visualizing vote_Average column
sns.catplot(data=df,y='Vote_Average',kind='count',order=df['Vote_Avera
ge'].value_counts().index)

<seaborn.axisgrid.FacetGrid at 0x2907233f860>
```



- movie got highest popularity // what is its genre

```
# checking max popularity in dataset
df[df['Popularity']==df['Popularity'].max()]

  Release_Date                    Title  Popularity  Vote_Count
Vote_Average  \
0   2021-12-15  Spider-Man: No Way Home    5083.954        8940
popular
1   2021-12-15  Spider-Man: No Way Home    5083.954        8940
popular
2   2021-12-15  Spider-Man: No Way Home    5083.954        8940
popular

  Original_Language            Genre  Release_year  Release_month  \
0                en           Action          2021             12
1                en        Adventure          2021             12
2                en  Science Fiction          2021             12

   Release_day Release_dow_name  Release_is_weekened
0           15        Wednesday                    0
```

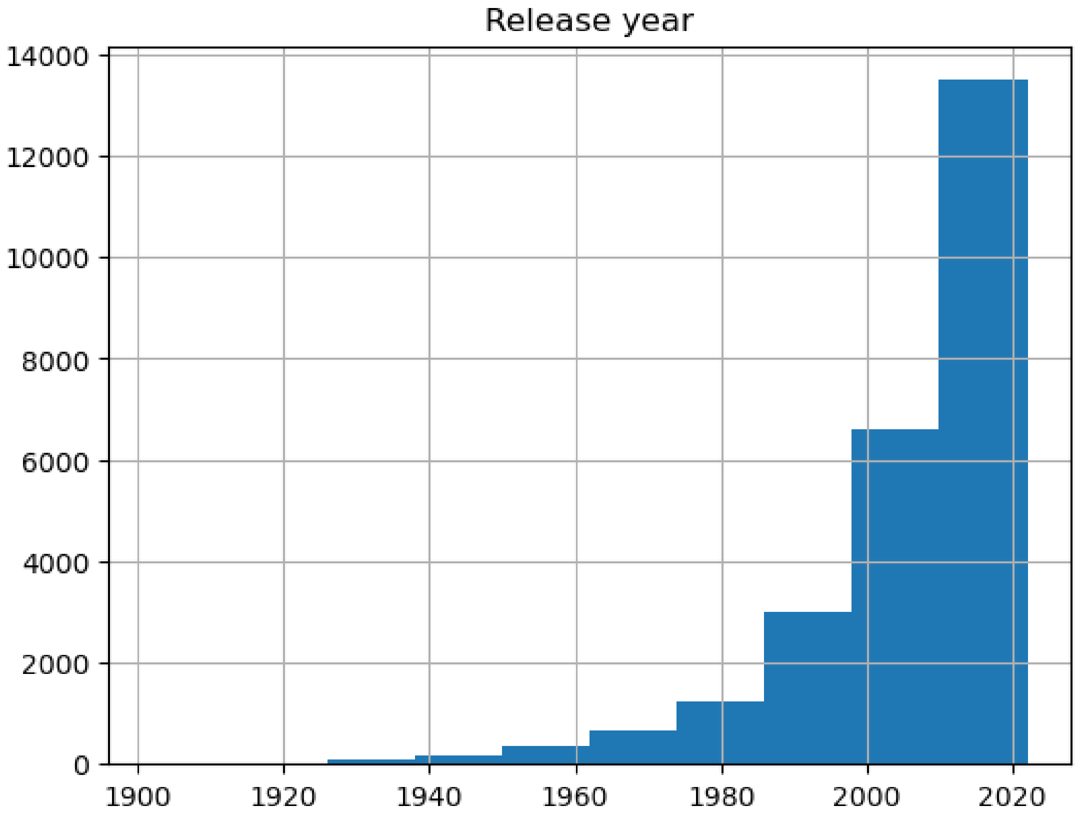
```
1           15        Wednesday                    0
2           15        Wednesday                    0
```

```
df[df['Popularity']==df['Popularity'].min()]
```

```
      Release_Date                                 Title  
Popularity  \
25546   2021-03-31  The United States vs. Billie Holiday      13.354

25547   2021-03-31  The United States vs. Billie Holiday      13.354

25548   2021-03-31  The United States vs. Billie Holiday      13.354

25549   1984-09-23                               Threads      13.354

25550   1984-09-23                               Threads      13.354

25551   1984-09-23                               Threads      13.354


       Vote_Count Vote_Average Original_Language            Genre  \
25546         152      average                en            Music
25547         152      average                en            Drama
25548         152      average                en          History
25549         186      popular                en              War
25550         186      popular                en            Drama
25551         186      popular                en  Science Fiction

       Release_year  Release_month  Release_day Release_dow_name  \
25546          2021              3           31        Wednesday
25547          2021              3           31        Wednesday
25548          2021              3           31        Wednesday
25549          1984              9           23           Sunday
25550          1984              9           23           Sunday
25551          1984              9           23           Sunday

       Release_is_weekened
25546                    0
25547                    0
25548                    0
25549                    1
25550                    1
25551                    1
```

- most filmmed movies

```
df['Release_year'].hist()
plt.title('Release year')
plt.show()
```



```
df.sample()

      Release_Date        Title  Popularity  Vote_Count
Vote_Average  \
22120   1994-04-29  With Honors      14.745         147       average


      Original_Language   Genre  Release_year  Release_month
Release_day  \
22120                en  Comedy          1994              4
29

      Release_dow_name  Release_is_weekened
22120           Friday                    0
```