EDA on Netflix dataset

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

```
df=pd.read_csv('mymoviedb.csv',lineterminator='\n')
df.head(2)
```

```
  Release_Date                    Title  \
0   2021-12-15  Spider-Man: No Way Home   
1   2022-03-01               The Batman   

                                            Overview  Popularity 
Vote_Count  \
0  Peter Parker is unmasked and no longer able to...    5083.954 
8940   
1  In his second year of fighting crime, Batman u...    3827.658 
1151   

   Vote_Average Original_Language                               Genre 
\
0           8.3                en  Action, Adventure, Science Fiction 
  
1           8.1                en            Crime, Mystery, Thriller 
  

                                          Poster_Url  
0  https://image.tmdb.org/t/p/original/1g0dhYtq4i...  
1  https://image.tmdb.org/t/p/original/74xTEgt7R3...  
```

```
df.isnull().sum()
```

```
Release_Date         0
Title                0
Overview             0
Popularity           0
Vote_Count           0
Vote_Average         0
Original_Language    0
Genre                0
Poster_Url           0
dtype: int64
```

```
df.info()
```

```
<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
```

```
df.duplicated().sum()
```

```
np.int64(0)
```

```
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
```

Exploration Summary

1. We have a dataframe consisting 9827 rows and 9 columns.
2. Our dataset looks a bit tidy with No NaN and duplicate Values.
3. Release Date Time Column need to be Casted into Date Time Format AND extract Year value from it.
4. Overview,Poster_Url and Original_Language columns would Not be useful for our analysis,So we will drop them.
5. Genre has a comma seperated values and white spaces that needs to be handled and casted into category.
6. Vote_Average better be categorized for proper analysis.

```
df['Release_Date'] = pd.to_datetime(df['Release_Date'])
```

```
df.dtypes
```

```
Release_Date         datetime64[ns]
Title                        object
```

```
Overview                     object
Popularity                  float64
Vote_Count                    int64
Vote_Average                float64
Original_Language            object
Genre                        object
Poster_Url                   object
dtype: object
```

```
df['Release_Year']=df['Release_Date'].dt.year
df.head(2)
```

```
  Release_Date                    Title  \
0   2021-12-15  Spider-Man: No Way Home
1   2022-03-01               The Batman

                                            Overview  Popularity
Vote_Count  \
0  Peter Parker is unmasked and no longer able to...    5083.954
8940
1  In his second year of fighting crime, Batman u...    3827.658
1151

   Vote_Average Original_Language                               Genre
\
0           8.3                en  Action, Adventure, Science Fiction

1           8.1                en            Crime, Mystery, Thriller


                                          Poster_Url  Release_Year
0  https://image.tmdb.org/t/p/original/1g0dhYtq4i...          2021
1  https://image.tmdb.org/t/p/original/74xTEgt7R3...          2022
```

Droping the columns

```
Cols=['Overview','Original_Language','Poster_Url','Release_Date']

df.drop(Cols,axis=1,inplace=True)
df.head(2)
```

```
                     Title  Popularity  Vote_Count  Vote_Average  \
0  Spider-Man: No Way Home    5083.954        8940           8.3
1               The Batman    3827.658        1151           8.1

                                Genre  Release_Year
0  Action, Adventure, Science Fiction          2021
1            Crime, Mystery, Thriller          2022
```

Categorizing the Vote_Average Column

We would be cut the Vote_Average column into three categories: popular,average,below average not_Popular. We will use the following thresholds by using the Categorize_col() function.

```
# Define bins and labels
bins = [0, 3, 5, 7, 10]  # Define category ranges
labels = ['Not Popular', 'Below Average', 'Average', 'Popular']  #
Define labels

# Apply categorization to your actual dataset column
df['Vote_Category'] = pd.cut(df['Vote_Average'], bins=bins,
labels=labels)

# Display the first few rows to verify the changes
print(df[['Vote_Average', 'Vote_Category']].head())
```

```
   Vote_Average Vote_Category
0           8.3       Popular
1           8.1       Popular
2           6.3       Average
3           7.7       Popular
4           7.0       Average
```

```
df['Vote_Category'].value_counts()
```

```
Vote_Category
Average          6295
Popular          2840
Below Average     561
Not Popular        31
Name: count, dtype: int64
```

```
df.dropna(inplace=True)
df.isna().sum()
```

```
Title            0
Popularity       0
Vote_Count       0
Vote_Average     0
Genre            0
Release_Year     0
Vote_Category    0
dtype: int64
```

```
df['Genre']=df['Genre'].str.split(', ')
```

```
df=df.explode('Genre').reset_index(drop=True)
df.head(100)
```

```
                      Title  Popularity  Vote_Count  Vote_Average  \
0   Spider-Man: No Way Home    5083.954        8940           8.3   
1   Spider-Man: No Way Home    5083.954        8940           8.3   
2   Spider-Man: No Way Home    5083.954        8940           8.3   
3                The Batman    3827.658        1151           8.1   
4                The Batman    3827.658        1151           8.1   
..                      ...         ...         ...           ...   
95          West Side Story     678.186         562           7.4   
96          West Side Story     678.186         562           7.4   
97          West Side Story     678.186         562           7.4   
98        Through My Window     659.105        1331           7.8   
99        Through My Window     659.105        1331           7.8   

              Genre  Release_Year Vote_Category  
0            Action          2021       Popular  
1         Adventure          2021       Popular  
2   Science Fiction          2021       Popular  
3             Crime          2022       Popular  
4           Mystery          2022       Popular  
..              ...           ...           ...  
95            Drama          2021       Popular  
96          Romance          2021       Popular  
97            Crime          2021       Popular  
98          Romance          2022       Popular  
99            Drama          2022       Popular  

[100 rows x 7 columns]
```

```
print(type('Genre'))
```

```
<class 'str'>
```

```
df['Genre']=df['Genre'].astype('category')
df['Genre'].dtypes
```

```
CategoricalDtype(categories=['Action', 'Adventure', 'Animation',
'Comedy', 'Crime',
                  'Documentary', 'Drama', 'Family', 'Fantasy',
'History',
                  'Horror', 'Music', 'Mystery', 'Romance', 'Science
Fiction',
                  'TV Movie', 'Thriller', 'War', 'Western'],
, ordered=False, categories_dtype=object)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25552 entries, 0 to 25551
Data columns (total 7 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
```

```
 0   Title           25552 non-null  object  
 1   Popularity      25552 non-null  float64 
 2   Vote_Count      25552 non-null  int64   
 3   Vote_Average    25552 non-null  float64 
 4   Genre           25552 non-null  category
 5   Release_Year    25552 non-null  int32   
 6   Vote_Category   25552 non-null  category
dtypes: category(2), float64(2), int32(1), int64(1), object(1)
memory usage: 949.2+ KB
```

```
df.nunique()
```

```
Title            9415
Popularity       8088
Vote_Count       3265
Vote_Average       73
Genre              19
Release_Year      100
Vote_Category       4
dtype: int64
```

```
df.drop(columns='Vote_Average',inplace=True)
```

```
df.head()
```

```
                    Title  Popularity  Vote_Count            Genre  \
0  Spider-Man: No Way Home    5083.954        8940           Action   
1  Spider-Man: No Way Home    5083.954        8940        Adventure   
2  Spider-Man: No Way Home    5083.954        8940  Science Fiction   
3               The Batman    3827.658        1151            Crime   
4               The Batman    3827.658        1151          Mystery   

   Release_Year Vote_Category  
0          2021       Popular  
1          2021       Popular  
2          2021       Popular  
3          2022       Popular  
4          2022       Popular  
```

Data Visualization

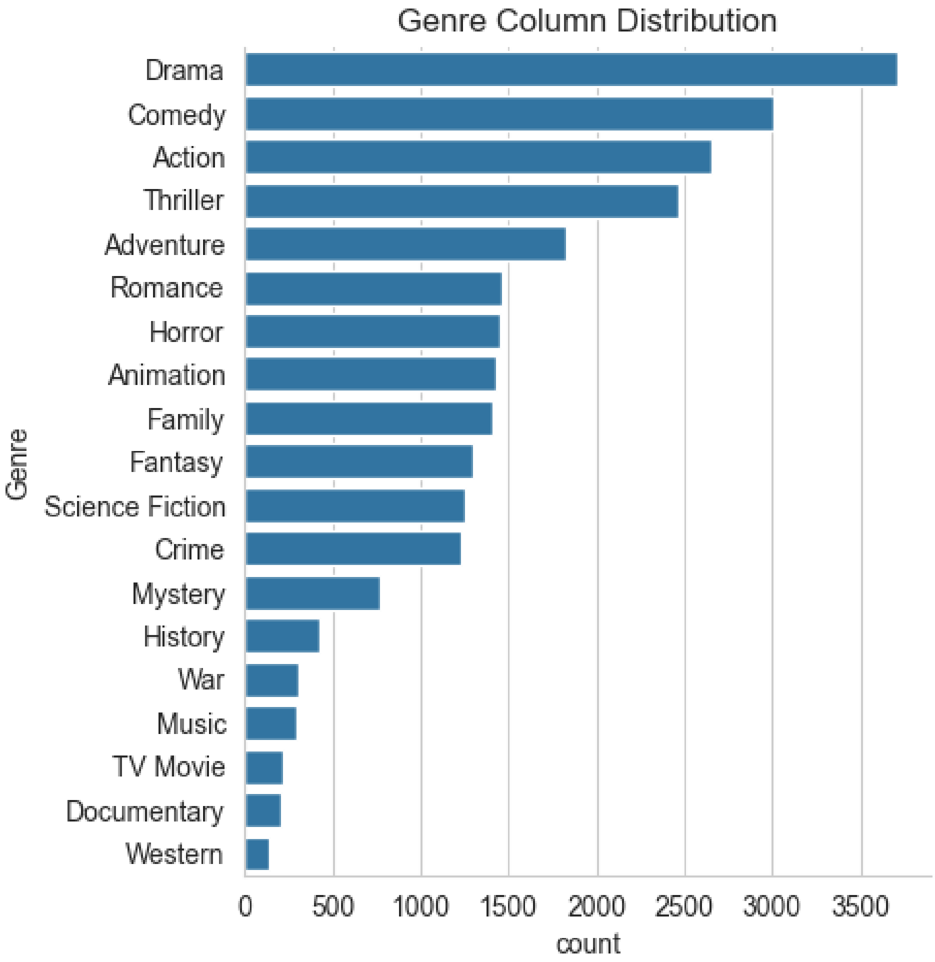
```
sns.set_style('whitegrid')
```

What is most frequent Genre of movies released on Netflix

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

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

```
sns.catplot(y='Genre',data=df,kind='count',
            order=df['Genre'].value_counts().index)
plt.title('Genre Column Distribution')
plt.show()
```



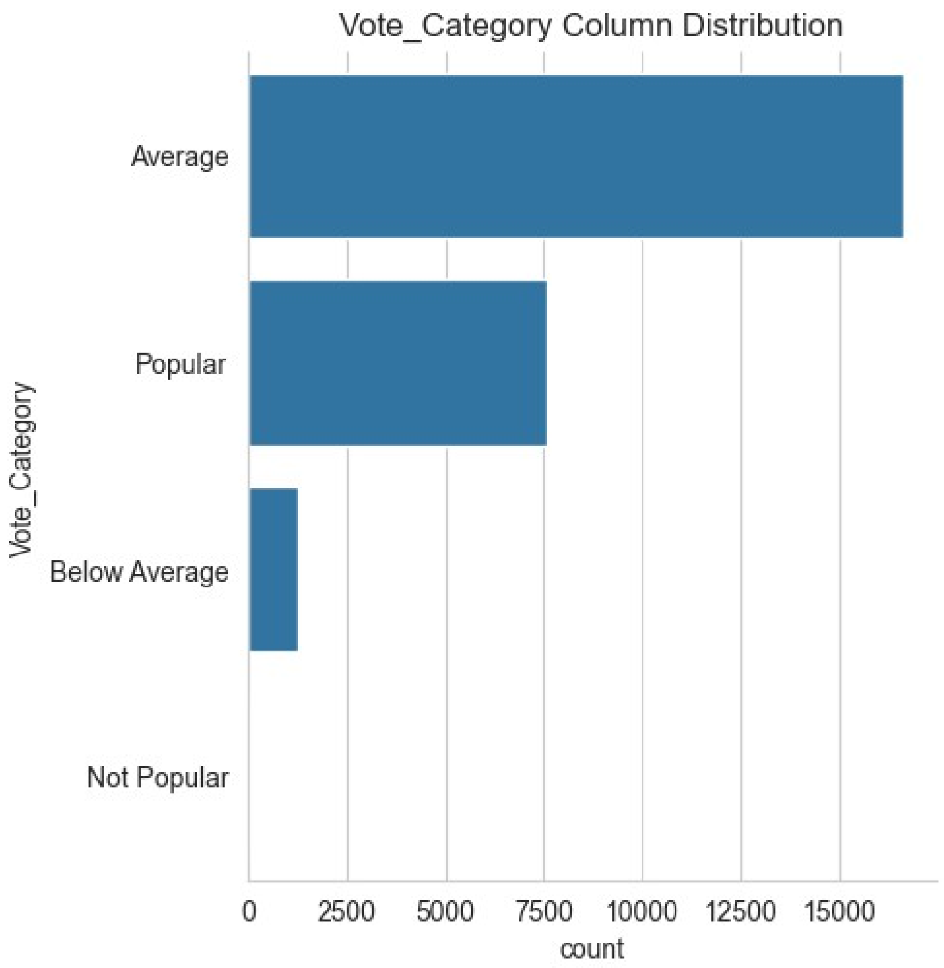
What is the highest vote in vote Category Column

```
df_Group=df.groupby('Vote_Category').agg({'Vote_Category':'count'})
df_Group
```

```
C:\Users\Abdul-Samad\AppData\Local\Temp\
ipykernel_11680\3853725479.py:1: FutureWarning: The default of
observed=False is deprecated and will be changed to True in a future
version of pandas. Pass observed=False to retain current behavior or
observed=True to adopt the future default and silence this warning.
  df_Group=df.groupby('Vote_Category').agg({'Vote_Category':'count'})
```

```
               Vote_Category
Vote_Category
Not Popular               64
Below Average           1297
Average                16631
Popular                 7560
```

```
sns.catplot(y='Vote_Category',data=df,kind='count',
            order=df['Vote_Category'].value_counts().index)
plt.title('Vote_Category Column Distribution')
plt.show()
```



What Movie got the hiest Popularity score

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

                    Title  Popularity  Vote_Count            Genre  \
0  Spider-Man: No Way Home    5083.954        8940           Action
1  Spider-Man: No Way Home    5083.954        8940        Adventure
2  Spider-Man: No Way Home    5083.954        8940  Science Fiction

   Release_Year Vote_Category
0          2021       Popular
1          2021       Popular
2          2021       Popular
```

What Movie got the lowest Popularity and its genre

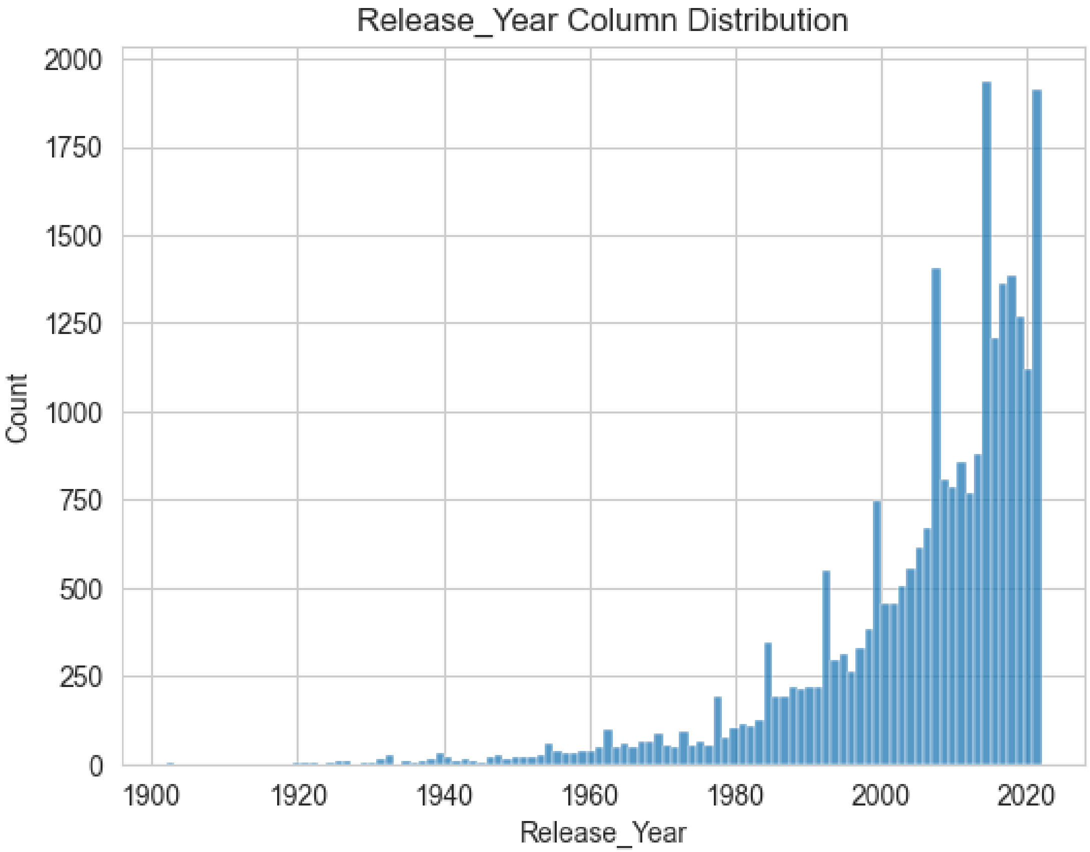
```
df[df['Popularity']==df['Popularity'].min()]
```

```
                                    Title  Popularity  Vote_Count  \
25546  The United States vs. Billie Holiday      13.354         152
25547  The United States vs. Billie Holiday      13.354         152
25548  The United States vs. Billie Holiday      13.354         152
25549                                 Threads      13.354         186
25550                                 Threads      13.354         186
25551                                 Threads      13.354         186

                Genre  Release_Year Vote_Category
25546            Music          2021       Average
25547            Drama          2021       Average
25548          History          2021       Average
25549              War          1984       Popular
25550            Drama          1984       Popular
25551  Science Fiction          1984       Popular
```

What year has the most filmmed Movies

```
sns.histplot(df['Release_Year'])
plt.title('Release_Year Column Distribution')
plt.show()
```



Conclusion:

Q1: # What is most frequent Genre of movies released on Netflix?

A1: The most frequent genre of movies released on Netflix is Drama.

Q2:# What is the highest vote in vote Category Column?

A2: The highest vote in the vote Category Column is Average vote = 8.5.

Q3:# What Movie got the lowest Popularity and its genre?

A3: The movie with the lowest popularity is "The United States vs. Billie Holiday AND Threads" with a popularity score of 13.354.

Q4: # What Movie got the hiest Popularity score ?

A4: The movie with the highest popularity is "Spider-Man: No Way Home " with a popularity score of 5083.954

Q5: # What year has the most filmmed Movies?

A5: The year with the most movies is 2020 with 43 movies.