import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import warnings

warnings.filterwarnings("ignore")

df=pd.read\_csv('/kaggle/input/top-250-imdb-series-episode-ratings/imdb\_top\_250\_series\_episode\_ratings.csv')

df1=pd.read\_csv('/kaggle/input/top-250-imdb-series-episode-ratings/imdb\_top\_250\_series\_global\_ratings.csv')

df.isna().sum()

df1.isna().sum()

df.head()

df1.head()

df.drop('Unnamed: 0',axis=1,inplace=True)

**## Q1 : What are the top TV Series by their Mean Episode Rating?**

df\_Q1=df[['Title','Rating']].groupby('Title').mean().round(2).sort\_values(by='Rating',ascending= False).head(10).reset\_index()

df\_Q1

import matplotlib.pyplot as plt

plt.bar(df\_Q1.Title,df\_Q1.Rating , color = 'orange')

plt.title('Top TV Series by their Mean Episode Rating',fontsize = 15 , color='m')

plt.xlabel('Title', fontsize = 12 , color='k')

plt.ylabel('Rating', fontsize = 12 , color='k')

plt.xticks(rotation=45)

plt.show

**## Q2 : Also by their Median Episode Rating**

df\_Q2=df[['Title','Rating']].groupby('Title').median().round(2).sort\_values(by='Rating',ascending= False).nlargest(10,'Rating').reset\_index()

df\_Q2

import matplotlib.pyplot as plt

plt.bar(df\_Q2.Title,df\_Q2.Rating , color = 'skyblue')

plt.title('Top TV Series by their Median Episode Rating',fontsize = 15 , color='m')

plt.xlabel('Title', fontsize = 12 , color='k')

plt.ylabel('Rating', fontsize = 12 , color='k')

plt.xticks(rotation=45)

plt.show

**##Chernobyl is the top TV Series by their Mean Episode Rating**

**## Q3 What are the top TV Seasons by their Mean Episode Rating**

df1.head()

df\_Q3=df.groupby(['Title','Season']).mean().round(2).sort\_values(by='Rating',ascending= False).nlargest(10,'Rating').reset\_index().drop('Episode',axis = 1)

df\_Q3

import matplotlib.pyplot as plt

plt.bar(df\_Q3.Title,df\_Q3.Rating , color = 'purple')

plt.title('Top TV Seasons by their Mean Episode Rating',fontsize = 15 , color='r')

plt.xlabel('Title', fontsize = 12 , color='k')

plt.ylabel('Rating', fontsize = 12 , color='k')

plt.xticks(rotation=45)

plt.show

**##Also by their Median Episode Rating**

df\_Q4=df.groupby(['Title','Season']).median().round(2).sort\_values(by='Rating',ascending= False).nlargest(10,'Rating').reset\_index().drop('Episode',axis = 1)

df\_Q4

import matplotlib.pyplot as plt

plt.bar(df\_Q4.Title,df\_Q4.Rating , color = 'y')

plt.title('Top TV Seasons by their Median Episode Rating',fontsize = 15 , color='r')

plt.xlabel('Title', fontsize = 12 , color='k')

plt.ylabel('Rating', fontsize = 12 , color='k')

plt.xticks(rotation=45)

plt.show

**##Chernobyl is the top TV Seasons by their Mean Episode Rating**

**##Q5 - What are the most loved TV Series endings (with the highest rating in the last episode)**

df.head()

df\_Q5= df.groupby(['Season','Title']).max().sort\_values('Rating',ascending = False).drop('Code',axis=1).nlargest(10,'Rating').reset\_index()

df\_Q5

import matplotlib.pyplot as plt

plt.bar(df\_Q5.Title,df\_Q5.Rating , color = 'lightgreen')

plt.title('The most loved TV Series ending',fontsize = 15 , color='k')

plt.xlabel('Title', fontsize = 12 , color='k')

plt.ylabel('Rating', fontsize = 12 , color='k')

plt.xticks(rotation=45)

plt.show

**##Breaking Bad is the most loved TV Series endings.**

**## Q6-What are the most loved TV Series starting (with the highest rating in the first episode)?**

df\_Q6= df.query('Episode==1').sort\_values('Rating',ascending=False).nlargest(10,'Rating').reset\_index().drop(['Code','index'], axis=1)

df\_Q6

import matplotlib.pyplot as plt

plt.bar(df\_Q6.Title,df\_Q6.Rating , color = 'c')

plt.title('The most loved TV Series ending',fontsize = 15 , color='k')

plt.xlabel('Title', fontsize = 12 , color='k')

plt.ylabel('Rating', fontsize = 12 , color='k')

plt.xticks(rotation=45)

plt.show

**##House M.D is the most loved TV Series starting**

**## Q6- What are the TV Series that continuously keep improving (with the highest slope in their rating trend)?**

df.head()

mylist=[]

for i in df['Title'].unique():

mylist.append(df[df['Title']==i].groupby('Title')['Rating'].pct\_change().round(2).max())

cf= df['Title'].unique()

NEwdflist= pd.DataFrame({'Title':cf,'Slope' : mylist})

NEwdflist.sort\_values('Slope',ascending= False).nlargest(20,'Slope')

NEwdflist\_1 = pd.merge(NEwdflist,df,how='inner',on='Title')

NEwdflist\_1.groupby('Title').max().sort\_values('Slope',ascending= False).nlargest(20,'Slope').drop(['Code','Season'],axis=1)

**##Table stating the Top 20 Percentage Changes with the highest positive slope change**

import matplotlib.pyplot as plt

plt.plot(NEwdflist\_1.Episode,NEwdflist\_1.Rating , color = 'r')

plt.title('Top 20 Percentage Changes with the highest positive slope change',fontsize = 15 , color='k')

plt.xlabel('Episode', fontsize = 12 , color='k')

plt.ylabel('Rating', fontsize = 12 , color='k')

plt.show

**#What's the global distribution of episode ratings?**

df.head()

df['Rating'].plot(kind='hist',title='The Global distribution of episode ratings' )

#Also global average and median?

df\_mean = df.groupby('Title').mean().round(2).reset\_index()

df\_median = df.groupby('Title').median().round(2).reset\_index()

df\_mean['Rating'].plot(kind='hist',title='Global average')

df\_median['Rating'].plot(kind='hist',title='Global median')

**## Which rage is above such average making it a "good episode"?**

df['Rating'].median()

df['Rating'].mean()

**#As per the Global Average and Global Mean the values more 8.07 is considered as a good range**

df.query('Rating > 8.065246725757374')

**## What's the series with the highest number of "good episodes"?**

df\_q= df[['Title','Rating']].query('Rating > 9.0').groupby('Title').count().sort\_values('Rating',ascending= False).nlargest(10,'Rating').reset\_index()

df\_q

plt.bar(df\_q.Title,df\_q.Rating,color = 'm')

plt.title('The top 10 series with the highest number of good episodes')

plt.xticks(rotation=45)

plt.show()

**##Gintama is the series with the most 72 episode with an IMDB rating more than 9**

**##Which series have a higher gap between their Mean Episode Rating and their TV Series Rating?**

df.head()

dfe=df[['Title','Rating']].groupby('Title').mean()

dfe.rename(columns={'Rating':'E\_Rating'}, inplace = True)

dfe=dfe.reset\_index()

dfe

df1

dfs=df1[['Title','Rating']].groupby('Title').mean()

dfs.rename(columns={'Rating':'S\_Rating'}, inplace = True)

dfs=dfs.reset\_index()

newdf=pd.merge(dfs,dfe,on='Title',how='inner')

newdf['GAP'] = newdf['E\_Rating']-newdf['S\_Rating']

newdf['GAP']=newdf['GAP'].abs()

newdf= newdf.sort\_values('GAP',ascending=False).nlargest(10,'GAP')

newdf

plt.bar(newdf.Title,newdf.GAP)

plt.xticks(rotation=45)

plt.title('Top 10 gap between their Mean Episode Rating and their TV Series Rating')

plt.show()

**## Mahabharat has the highest gap Mean Episode Rating and TV Series Rating.**

**Thank You**