```
import pandas as pd
import numpy as np
from sklearn.metrics.pairwise import cosine_similarity
import os
import matplotlib.pyplot as plt
import seaborn as sns


from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

Performing EDA and analysing data

```
# First Let's load the movie details into soe dataframe..
# movie details are in 'netflix/movie_titles.csv'

movie_titles = pd.read_csv("/content/drive/MyDrive/movie recommender system/movie.csv")

movie_titles.head()
```

	item_id	title
0	1	Toy Story (1995)
1	2	GoldenEye (1995)
2	3	Four Rooms (1995)
3	4	Get Shorty (1995)
4	5	Copycat (1995)

```
movie_titles.shape
```

```
(1682, 2)
```

```
#reading csv file
user=pd.read_csv("/content/drive/MyDrive/movie recommender system/user.csv")
print("No of data points and features is:",user.shape)
user.head()
```

```
No of data points and features is: (100004, 4)
```

	user_id	item_id	rating	timestamp
0	0.0	50.0	5.0	881250949.0
1	0.0	172.0	5.0	881250949.0
2	0.0	133.0	1.0	881250949.0
3	196.0	242.0	3.0	881250949.0
4	186.0	302.0	3.0	891717742.0

```
data=pd.merge(user,movie_titles,on='item_id')
data.head()
```

	user_id	item_id	rating	timestamp	title
0	0.0	50.0	5.0	881250949.0	Star Wars (1977)
1	290.0	50.0	5.0	880473582.0	Star Wars (1977)
2	79.0	50.0	4.0	891271545.0	Star Wars (1977)
3	2.0	50.0	5.0	888552084.0	Star Wars (1977)
4	8.0	50.0	5.0	879362124.0	Star Wars (1977)

```
#Checking avg rating given
data.describe()['rating']
```

```
count    100003.000000
mean          3.529864
std           1.125704
min           1.000000
25%           3.000000
50%           4.000000
75%           4.000000
max           5.000000
Name: rating, dtype: float64
```

```
#checking for NAN and duplicate values
print("No of null values is:",data.isnull().count())
print("No of duplicate values is:",sum(data.duplicated()))
```

```
No of null values is: user_id      100003
item_id      100003
rating       100003
timestamp    100003
title        100003
dtype: int64
No of duplicate values is: 0
```

```
#Basic stats
print("Total data size",data.shape)
print("No of users",data['user_id'].unique().shape[0])
print("No of movies",data['item_id'].unique().shape[0])
```

```
Total data size (100003, 5)
No of users 944
No of movies 1682
```

```
data.groupby('title')['rating'].mean().sort_values(ascending=False).head()
```

```
title
Entertaining Angels: The Dorothy Day Story (1996)    5.0
Someone Else's America (1995)                        5.0
Star Kid (1997)                                      5.0
Saint of Fort Washington, The (1993)                 5.0
Santa with Muscles (1996)                            5.0
Name: rating, dtype: float64
```

```
data.groupby('title')['rating'].count().sort_values(ascending=False).head()
```

```
title
Star Wars (1977)             584
Contact (1997)               509
Fargo (1996)                 508
Return of the Jedi (1983)    507
Liar Liar (1997)             485
Name: rating, dtype: int64
```

```
ratings=pd.DataFrame(data.groupby('title')['rating'].mean())
ratings.head()
```

title	rating
1-900 (1994)	2.600000
101 Dalmatians (1996)	2.908257
12 Angry Men (1957)	4.344000
187 (1997)	3.024390
2 Days in the Valley (1996)	3.225806

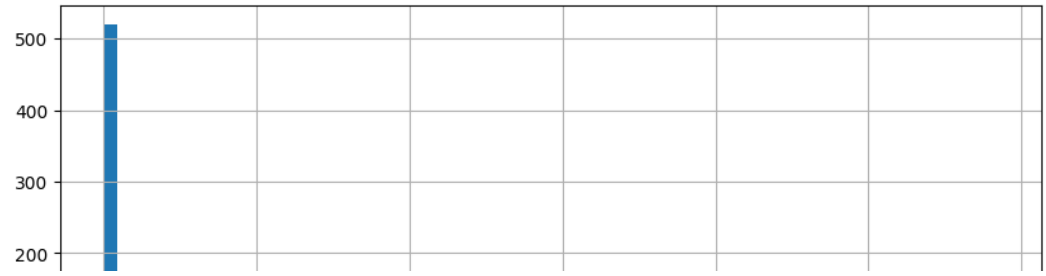
```
ratings['no of rating']=pd.DataFrame(data.groupby('title')['rating'].count())
ratings.reset_index(inplace=True)
ratings.head()
```

	title	rating	no of rating
0	1-900 (1994)	2.600000	5
1	101 Dalmatians (1996)	2.908257	109
2	12 Angry Men (1957)	4.344000	125
3	187 (1997)	3.024390	41
4	2 Days in the Valley (1996)	3.225806	93

Plotting few histograms for ratings dataframe

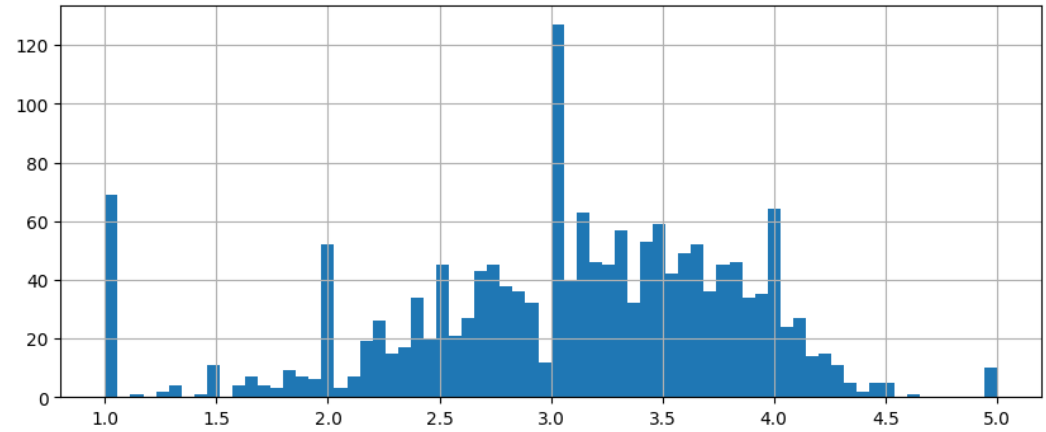
```
plt.figure(figsize=(10,4))
ratings['no of rating'].hist(bins=70)
```

<Axes: >




```
plt.figure(figsize=(10,4))
ratings['rating'].hist(bins=70)
```

<Axes: >




```
sim_mat=data.pivot_table(index='user_id',columns='title',values='rating')
sim_mat.head()
```

title	1-900 (1994)	101 Dalmatians (1996)	12 Angry Men (1957)	187 (1997)	2 Days in the Valley (1996)	20,000 Leagues Under the Sea (1954)	2001: A Space Odyssey (1968)	3 Ninjas: High Noon At Mega Mountain (1998)	39 Steps, The (1935)	8 1/2 (1963)	...	Ya (1
user_id												
0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	
1.0	NaN	2.0	5.0	NaN	NaN	3.0	4.0	NaN	NaN	NaN	...	
2.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN	...	
3.0	NaN	NaN	NaN	2.0	NaN	NaN	NaN	NaN	NaN	NaN	...	
4.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	

5 rows × 1664 columns

```
sim_mat.shape
```

(944, 1664)

```
ratings.sort_values('no of rating',ascending=False).head(10)
```

Now we will choose 2 diff category movies and see their similarities ex: Star Wars(Sci-fi)

```
starwar_rating=sim_mat['Star Wars (1977)']
starwar_rating.head()
```

```
user_id
0.0    5.0
1.0    5.0
2.0    5.0
3.0    NaN
4.0    5.0
Name: Star Wars (1977), dtype: float64
```

```
#movies similar to starwars
similar_to_starwars=sim_mat.corrwith(starwar_rating)
```

```
/usr/local/lib/python3.10/dist-packages/numpy/lib/function_base.py:2821: RuntimeWarning: Degrees of freedom <= 0 for slice
  c = cov(x, y, rowvar, dtype=dtype)
/usr/local/lib/python3.10/dist-packages/numpy/lib/function_base.py:2680: RuntimeWarning: divide by zero encountered in true_divide
  c *= np.true_divide(1, fact)
```

```
corr_to_starwars=pd.DataFrame(similar_to_starwars,columns=['Correlation']).reset_index()
corr_to_starwars.dropna(inplace=True)
corr_to_starwars.sort_values(by='Correlation',ascending=False).head(10)
```

	title	Correlation
934	Man of the Year (1995)	1.0
687	Hollow Reed (1996)	1.0
1417	Stripes (1981)	1.0
1397	Star Wars (1977)	1.0
342	Cosi (1996)	1.0
325	Commandments (1997)	1.0
1071	No Escape (1994)	1.0
1090	Old Lady Who Walked in the Sea, The (Vieille q...	1.0
1113	Outlaw, The (1943)	1.0
865	Line King: Al Hirschfeld, The (1996)	1.0

```
#in this we will filter movies with less than 100 reviews
corr_starwars=corr_to_starwars.join(ratings['no of rating'])
corr_starwars.head()
```

	title	Correlation	no of rating
0	1-900 (1994)	-0.645497	5
1	101 Dalmatians (1996)	0.211132	109
2	12 Angry Men (1957)	0.184289	125
3	187 (1997)	0.027398	41
4	2 Days in the Valley (1996)	0.066654	93

```
corr_starwars[corr_starwars['no of rating']>100].sort_values('Correlation',ascending=False).head()
```

	title	Correlation	no of rating
1397	Star Wars (1977)	1.000000	584
455	Empire Strikes Back, The (1980)	0.748353	368
1233	Return of the Jedi (1983)	0.672556	507
1204	Raiders of the Lost Ark (1981)	0.536117	420
103	Austin Powers: International Man of Mystery (1...	0.377433	130

```
#Creating function for recommending movies based on similarity given movie name
def recommend_movies(name):
  import warnings
  warnings.filterwarnings("ignore")

  #getting all ratings forgive movie using similarity matrix
  movie_rating=sim_mat[name]

  #finding similar movies using correlation
  similar_to_movie=sim_mat.corrwith(movie_rating)
  corr_to_movie=pd.DataFrame(similar_to_movie,columns=['Correlation']).reset_index()
```

```
  corr_to_movie.dropna(inplace=True)
  corr_to_movie.sort_values(by='Correlation',ascending=False)

  #Merging 'No of rating' to filter movies based on no of ratings given
  corr_movie=corr_to_movie.join(ratings['no of rating'])

  print('\033[1m'+"TOP 25 MOVIES THAT ARE SIMILAR TO",name,"ARE:")
  #considering movies with no of rating >100
  top_movies=corr_movie[corr_movie['no of rating']>100].sort_values('Correlation',ascending=False).head(26)
  #gave index as [1:] because first value is that movie itself
  return top_movies[1:]['title']


movie= input("Enter the movie you want similarities of:")
recommend_movies(movie)
```

```
Enter the movie you want similarities of:Star Wars (1977)
TOP 25 MOVIES THAT ARE SIMILAR TO Star Wars (1977) ARE:
455                       Empire Strikes Back, The (1980)
1233                            Return of the Jedi (1983)
1204                         Raiders of the Lost Ark (1981)
103     Austin Powers: International Man of Mystery (1...
1406                                     Sting, The (1973)
746              Indiana Jones and the Last Crusade (1989)
1155                                     Pinocchio (1940)
566                                Frighteners, The (1996)
828                              L.A. Confidential (1997)
1589                                   Wag the Dog (1997)
442                                          Dumbo (1941)
231                   Bridge on the River Kwai, The (1957)
1147                        Philadelphia Story, The (1940)
983                          Miracle on 34th Street (1994)
445                       E.T. the Extra-Terrestrial (1982)
1036       Mystery Science Theater 3000: The Movie (1996)
299                                      Cinderella (1950)
131                                          Batman (1989)
1440                                      Swingers (1996)
516                                 Field of Dreams (1989)
584                                          Gattaca (1997)
1396                  Star Trek: The Wrath of Khan (1982)
112                             Back to the Future (1985)
1349               Snow White and the Seven Dwarfs (1937)
1644                           Wizard of Oz, The (1939)
Name: title, dtype: object
```