SOURCE CODE:

```
import pandas as pd

import numpy as np

from scipy import sparse

from sklearn.metrics.pairwise import cosine_similarity

from sklearn.model_selection import train_test_split

data = pd.read_csv('C:/Users/psriv/OneDrive/Documents/Simplilearn/Amazon/Amazon - Movies and TV Ratings.csv')

print("DATA")

print(data)

sums=data.sum(axis=None, skipna=None, level=None, numeric_only=None, min_count=0)

print("sums")

print(sums)

new_sum=sums.to_frame()

print(new_sum)

new_sum['count']=data.count()

print(new_sum)

new_sum.drop(['user_id'],inplace=True)

print(new_sum)

new_sum.rename(columns = {0:'Sum'}, inplace = True)

print(new_sum)

for ind, row in new_sum.iterrows():

    new_sum.loc[ind,'total review'] = row['Sum'] / row['count']

Highest_views= new_sum.sort_values(by=['count'], ascending=False)

print(Highest_views)

print("MOVIE HAVING MAXIMUM VIEWS")

print(Highest_views.head(1))

print("MAXIMUM AVERAGE OF EACH MOVIE")

print(new_sum)
```

```
Highest_rating= new_sum.sort_values(by=['total review'], ascending=False)
print(Highest_rating)
print("THE TOP 5 MOVIE WITH THE MAXIMUM RATING")
print(Highest_rating.head(5))
least_rating= new_sum.sort_values(by=['total review'], ascending=True)
print(least_rating)
print("THE TOP 5 MOVIE WITH THE LEAST RATING")
print(least_rating.head(5))
data1 = pd.read_csv('C:/Users/psriv/OneDrive/Documents/Simplilearn/Amazon/Amazon - Movies and TV Ratings.csv',index_col=0)
print("DATA")
print(data1)
new_data=data1.fillna(0)
print(new_data)
def standardize(row):
   n_row = (row -row.mean())/(row.max()-row.min())
   return n_row
rating = new_data.apply(standardize)
item=cosine_similarity(rating.T)
print(item)
item_data=pd.DataFrame(item,index=rating.columns,columns=new_data.columns)
print(item_data)
#Recommendation System with Collaborative Filtering
def get_similar_movie(movie_name,user_rating):
   similar_score=item_data[movie_name]*(user_rating-2.5)
   similar_score=similar_score.sort_values(ascending=False)
   print(similar_score)
print(" provides the ratings for each of the users based on there rating on single movie")
print(get_similar_movie("Movie205",5))
```

```
print(" provides the ratings for each of the users based on there rating on serval movie")

movie1_lover=[("Movie25",1),("Movie56",2),("Movie66",3),("Movie78",4),("Movie124",5),("Movie200",
2)]

simi_movies=pd.DataFrame()


for movie,rating in movie1_lover:

    simi_movies=simi_movies.append(get_similar_movie(movie,rating),ignore_index=True)


simi_movies.head()

simi_movies.sum().sort_values(ascending=False)
```