In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
data=pd.read_csv('movie.csv')
data.head()
```

Out[2]:

	userId	movie	rating
0	3	Toy Story (1995)	4.0
1	6	Toy Story (1995)	5.0
2	8	Toy Story (1995)	4.0
3	10	Toy Story (1995)	4.0
4	11	Toy Story (1995)	4.5

In [3]:
```
data.userId.unique()
```

Out[3]:
```
array([   3,    6,    8, ..., 7080, 7087, 7105], dtype=int64)
```

In [4]:
```
data.movie.nunique()
```

Out[4]:
```
10
```

In [5]:
```
data['rating'].value_counts().sort_index()
```

Out[5]:
```
0.5      57
1.0     212
1.5      61
2.0     542
2.5     277
3.0    2736
3.5     679
4.0    2660
4.5     374
5.0    1394
Name: rating, dtype: int64
```

In [6]:
```
data.movie.value_counts()
```

Out[6]:
```
Toy Story (1995)                    2569
GoldenEye (1995)                    1548
Heat (1995)                         1260
Jumanji (1995)                      1155
Sabrina (1995)                       700
Grumpier Old Men (1995)              685
Father of the Bride Part II (1995)   657
Sudden Death (1995)                  202
Waiting to Exhale (1995)             138
Tom and Huck (1995)                   78
Name: movie, dtype: int64
```

In [7]:
```
data.columns
```

Out[7]:
```
Index(['userId', 'movie', 'rating'], dtype='object')
```

In [8]:
```
user_movie_df=data.pivot(index="userId", columns='movie', values='rating')
user_movie_df
```

Out[8]:

movie userId	Father of the Bride Part II (1995)	GoldenEye (1995)	Grumpier Old Men (1995)	Heat (1995)	Jumanji (1995)	Sabrina (1995)	Sudden Death (1995)	Tom and Huck (1995)	Toy Story (1995)	Waiting to Exhale (1995)
1	NaN	NaN	NaN	NaN	3.5	NaN	NaN	NaN	NaN	NaN
2	NaN	NaN	4.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	4.0	NaN
4	NaN	4.0	NaN	3.0	NaN	NaN	NaN	NaN	NaN	NaN
5	NaN	NaN	NaN	NaN	3.0	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...
7115	4.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
7116	3.5	NaN	NaN	NaN	NaN	NaN	NaN	NaN	4.0	NaN
7117	NaN	3.0	4.0	5.0	NaN	3.0	1.0	NaN	4.0	NaN
7119	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	5.0	NaN
7120	NaN	NaN	NaN	NaN	4.0	4.0	NaN	NaN	4.5	NaN

4081 rows × 10 columns

In [9]:
```
user_movie_df.fillna(0,inplace=True)
user_movie_df
```

Out[9]:

movie userId	Father of the Bride Part II (1995)	GoldenEye (1995)	Grumpier Old Men (1995)	Heat (1995)	Jumanji (1995)	Sabrina (1995)	Sudden Death (1995)	Tom and Huck (1995)	Toy Story (1995)	Waiting to Exhale (1995)
1	0.0	0.0	0.0	0.0	3.5	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.0	0.0
4	0.0	4.0	0.0	3.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	3.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...
7115	4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7116	3.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.0	0.0
7117	0.0	3.0	4.0	5.0	0.0	3.0	1.0	0.0	4.0	0.0
7119	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.0	0.0
7120	0.0	0.0	0.0	0.0	4.0	4.0	0.0	0.0	4.5	0.0

4081 rows × 10 columns

In [10]:
```
#Pair wise disance
from scipy.spatial.distance import cosine,correlation
from sklearn.metrics import pairwise_distances
```

In [11]:
```
user_sim=1-pairwise_distances(user_movie_df,metric="cosine")
```

In [12]:
```
user_sim_df=pd.DataFrame(user_sim)
```

In [13]:
```
data.userId.values
```

Out[13]:
```
array([   3,    6,    8, ..., 7105, 7113, 7117], dtype=int64)
```

In [14]:
```
user_sim_df.index=data.userId.unique()
user_sim_df.columns=data.userId.unique()
```

In [15]:
```
np.fill_diagonal(user_sim,0)
```

In [16]:
```
user_sim_df.iloc[:5,:5]
```

Out[16]:

	3	6	8	10	11
3	0.0	0.0	0.0	0.0	1.0
6	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0
11	1.0	0.0	0.0	0.0	0.0

In [17]:
```
user_sim_df.idxmax(axis=1).head()
```

Out[17]:
```
3       11
6      168
8       16
10    4047
11       3
dtype: int64
```

In [18]:
```
data[(data['userId']==3)|(data['userId']==11)]
```

Out[18]:

	userId	movie	rating
0	3	Toy Story (1995)	4.0
4	11	Toy Story (1995)	4.5
7446	11	GoldenEye (1995)	2.5

In [19]:
```
user_1=data[(data['userId']==6)]
user_2=data[data['userId']==168 ]
user_2
```

Out[19]:

	userId	movie	rating
60	168	Toy Story (1995)	4.5

In [20]:
```
pd.merge(user_1,user_2,how="outer",on="movie")
```

Out[20]:

	userId_x	movie	rating_x	userId_y	rating_y
0	6	Toy Story (1995)	5.0	168.0	4.5
1	6	Grumpier Old Men (1995)	3.0	NaN	NaN
2	6	Sabrina (1995)	5.0	NaN	NaN