In [2]:
```
import pandas as pd
```

In [4]:
```
movies = pd.read_csv(r'F:\Gen AI & Agentic AI by Praskash Senapati\Gen AI, Agent
movies.shape
```

Out[4]:
```
(27278, 3)
```

In [5]:
```
movies
```

Out[5]:

	movieId	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy
...	...	...	...
27273	131254	Kein Bund für's Leben (2007)	Comedy
27274	131256	Feuer, Eis & Dosenbier (2002)	Comedy
27275	131258	The Pirates (2014)	Adventure
27276	131260	Rentun Ruusu (2001)	(no genres listed)
27277	131262	Innocence (2014)	Adventure|Fantasy|Horror

27278 rows × 3 columns

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

Out[7]:
```
Index(['movieId', 'title', 'genres'], dtype='object')
```

In [6]:
```
print(type(movies))
```

```
<class 'pandas.core.frame.DataFrame'>
```

In [7]:
```
movies.head(10) #top 10 rows
```

Out[7]:

	movieId	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy
5	6	Heat (1995)	Action|Crime|Thriller
6	7	Sabrina (1995)	Comedy|Romance
7	8	Tom and Huck (1995)	Adventure|Children
8	9	Sudden Death (1995)	Action
9	10	GoldenEye (1995)	Action|Adventure|Thriller

In [8]:

```
movies.tail(10) #last 10 rows
```

Out[8]:

	movieId	title	genres
27268	131241	Ants in the Pants (2000)	Comedy|Romance
27269	131243	Werner - Gekotzt wird später (2003)	Animation|Comedy
27270	131248	Brother Bear 2 (2006)	Adventure|Animation|Children|Comedy|Fantasy
27271	131250	No More School (2000)	Comedy
27272	131252	Forklift Driver Klaus: The First Day on the Jo...	Comedy|Horror
27273	131254	Kein Bund für's Leben (2007)	Comedy
27274	131256	Feuer, Eis & Dosenbier (2002)	Comedy
27275	131258	The Pirates (2014)	Adventure
27276	131260	Rentun Ruusu (2001)	(no genres listed)
27277	131262	Innocence (2014)	Adventure|Fantasy|Horror

In [9]:

```
tags=pd.read_csv(r'F:\Gen AI & Agentic AI by Praskash Senapati\Gen AI, Agentic A
tags
```

Out[9]:

	userId	movieId	tag	timestamp
0	18	4141	Mark Waters	2009-04-24 18:19:40
1	65	208	dark hero	2013-05-10 01:41:18
2	65	353	dark hero	2013-05-10 01:41:19
3	65	521	noir thriller	2013-05-10 01:39:43
4	65	592	dark hero	2013-05-10 01:41:18
...	...	...	...	...
465559	138446	55999	dragged	2013-01-23 23:29:32
465560	138446	55999	Jason Bateman	2013-01-23 23:29:38
465561	138446	55999	quirky	2013-01-23 23:29:38
465562	138446	55999	sad	2013-01-23 23:29:32
465563	138472	923	rise to power	2007-11-02 21:12:47

465564 rows × 4 columns

```
In [10]: tags.shape
```

```
Out[10]: (465564, 4)
```

```
In [11]: tags.columns
```

```
Out[11]: Index(['userId', 'movieId', 'tag', 'timestamp'], dtype='object')
```

```
In [14]: del tags['timestamp']
```

```
In [15]: tags
```

Out[15]:

	userId	movieId	tag
0	18	4141	Mark Waters
1	65	208	dark hero
2	65	353	dark hero
3	65	521	noir thriller
4	65	592	dark hero
...	...	...	...
465559	138446	55999	dragged
465560	138446	55999	Jason Bateman
465561	138446	55999	quirky
465562	138446	55999	sad
465563	138472	923	rise to power

465564 rows × 3 columns

In [19]:
```
row_0=tags.iloc[0]
print(row_0)
```

```
userId                18
movieId             4141
tag          Mark Waters
Name: 0, dtype: object
```

In [20]:
```
row_0.index
```

Out[20]:
```
Index(['userId', 'movieId', 'tag'], dtype='object')
```

In [22]:
```
print(row_0['userId'])
```

```
18
```

In [24]:
```
tags.iloc[[0,10,100]]
```

Out[24]:

	userId	movieId	tag
0	18	4141	Mark Waters
10	65	1694	jesus
100	121	52973	drugs

In [12]:
```
ratings = pd.read_csv(r'F:\Gen AI & Agentic AI by Praskash Senapati\Gen AI, Agen
ratings.shape
```

Out[12]:
```
(20000263, 4)
```

In [13]:
```
ratings
```

Out[13]:

	userId	movieId	rating	timestamp
0	1	2	3.5	2005-04-02 23:53:47
1	1	29	3.5	2005-04-02 23:31:16
2	1	32	3.5	2005-04-02 23:33:39
3	1	47	3.5	2005-04-02 23:32:07
4	1	50	3.5	2005-04-02 23:29:40
...	...	...	...	...
20000258	138493	68954	4.5	2009-11-13 15:42:00
20000259	138493	69526	4.5	2009-12-03 18:31:48
20000260	138493	69644	3.0	2009-12-07 18:10:57
20000261	138493	70286	5.0	2009-11-13 15:42:24
20000262	138493	71619	2.5	2009-10-17 20:25:36

20000263 rows × 4 columns

In [16]:

```
del ratings['timestamp']
```

In [17]:

```
ratings
```

Out[17]:

	userId	movieId	rating
0	1	2	3.5
1	1	29	3.5
2	1	32	3.5
3	1	47	3.5
4	1	50	3.5
...	...	...	...
20000258	138493	68954	4.5
20000259	138493	69526	4.5
20000260	138493	69644	3.0
20000261	138493	70286	5.0
20000262	138493	71619	2.5

20000263 rows × 3 columns

Descriptive Statistics

In [28]:

```
ratings.describe()
```

Out[28]:

	userId	movieId	rating
count	2.000026e+07	2.000026e+07	2.000026e+07
mean	6.904587e+04	9.041567e+03	3.525529e+00
std	4.003863e+04	1.978948e+04	1.051989e+00
min	1.000000e+00	1.000000e+00	5.000000e-01
25%	3.439500e+04	9.020000e+02	3.000000e+00
50%	6.914100e+04	2.167000e+03	3.500000e+00
75%	1.036370e+05	4.770000e+03	4.000000e+00
max	1.384930e+05	1.312620e+05	5.000000e+00

In [26]:

```
ratings['rating'].describe()
```

Out[26]:

```
count    2.000026e+07
mean     3.525529e+00
std      1.051989e+00
min      5.000000e-01
25%      3.000000e+00
50%      3.500000e+00
75%      4.000000e+00
max      5.000000e+00
Name: rating, dtype: float64
```

In [27]:

```
ratings['movieId'].describe()
```

Out[27]:

```
count    2.000026e+07
mean     9.041567e+03
std      1.978948e+04
min      1.000000e+00
25%      9.020000e+02
50%      2.167000e+03
75%      4.770000e+03
max      1.312620e+05
Name: movieId, dtype: float64
```

In [29]:

```
ratings.mean()
```

Out[29]:

```
userId     69045.872583
movieId     9041.567330
rating         3.525529
dtype: float64
```

In [30]:

```
ratings.median()
```

Out[30]:

```
userId     69141.0
movieId     2167.0
rating         3.5
dtype: float64
```

In [32]:

```
ratings['rating'].min()
```

Out[32]:

```
0.5
```

In [33]:
```
ratings['rating'].max()
```

Out[33]:
```
5.0
```

In [34]:
```
ratings.corr()
```

Out[34]:

	userId	movieId	rating
userId	1.000000	-0.000850	0.001175
movieId	-0.000850	1.000000	0.002606
rating	0.001175	0.002606	1.000000

In [35]:
```
rating_5=ratings['rating']>=5
rating_5
```

Out[35]:
```
0           False
1           False
2           False
3           False
4           False
            ...
20000258    False
20000259    False
20000260    False
20000261     True
20000262    False
Name: rating, Length: 20000263, dtype: bool
```

In [39]:
```
rating_5.any()
```

Out[39]:
```
np.True_
```

In [40]:
```
print(rating_5.any())
```

```
True
```

In [41]:
```
rating_0=ratings['rating']>=0
rating_0
```

Out[41]:
```
0           True
1           True
2           True
3           True
4           True
            ...
20000258    True
20000259    True
20000260    True
20000261    True
20000262    True
Name: rating, Length: 20000263, dtype: bool
```

In [42]:
```
rating_0.any()
```

Out[42]:
```
np.True_
```

Data cleaning: Handling missing data

In [43]:
```
movies
```

Out[43]:

	movieId	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy
...	...	...	...
27273	131254	Kein Bund für's Leben (2007)	Comedy
27274	131256	Feuer, Eis & Dosenbier (2002)	Comedy
27275	131258	The Pirates (2014)	Adventure
27276	131260	Rentun Ruusu (2001)	(no genres listed)
27277	131262	Innocence (2014)	Adventure|Fantasy|Horror

27278 rows × 3 columns

In [44]:
```
ratings
```

Out[44]:

	userId	movieId	rating
0	1	2	5
1	1	29	5
2	1	32	5
3	1	47	5
4	1	50	5
...	...	...	...
20000258	138493	68954	5
20000259	138493	69526	5
20000260	138493	69644	5
20000261	138493	70286	5
20000262	138493	71619	5

20000263 rows × 3 columns

In [47]:
```
movies.isnull().any()
```

Out[47]:
```
movieId    False
title      False
genres     False
dtype: bool
```

In [48]:
```
ratings.isnull().any()
```

Out[48]:
```
userId     False
movieId    False
rating     False
dtype: bool
```

In [49]:
```
tags.isnull().any()
```

Out[49]:
```
userId     False
movieId    False
tag         True
dtype: bool
```

In [50]:
```
tags.duplicated()
```

Out[50]:
```
0         False
1         False
2         False
3         False
4         False
          ...  
465559    False
465560    False
465561    False
465562    False
465563    False
Length: 465564, dtype: bool
```

```
In [54]: tags=tags.dropna() #dropping null values
```

```
In [53]: print(tags.isnull().any())
```

```
userId     False
movieId    False
tag        False
dtype: bool
```

```
In [56]: tags.shape
```

```
Out[56]: (465548, 3)
```

```
In [ ]:
```