kaggle task

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:
```
ratings=pd.read_csv(r'E:\rating.csv')
```

In [3]:
```
ratings
```

Out[3]:

	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 [4]:
```
movies=pd.read_csv(r'E:\movie.csv')
```

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 [6]:
```
tags=pd.read_csv(r'E:\tag.csv')
```

In [7]:
```
tags
```

Out[7]:

	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 [8]:

```
print(tags.columns)
print(ratings.columns)
print(movies.columns)
```

```
Index(['userId', 'movieId', 'tag', 'timestamp'], dtype='object')
Index(['userId', 'movieId', 'rating', 'timestamp'], dtype='object')
Index(['movieId', 'title', 'genres'], dtype='object')
```

In [9]:

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

In [10]:

```
print(tags.columns)
print(ratings.columns)
print(movies.columns)
```

```
Index(['userId', 'movieId', 'tag'], dtype='object')
Index(['userId', 'movieId', 'rating'], dtype='object')
Index(['movieId', 'title', 'genres'], dtype='object')
```

In [11]:

```
tags.head(3)
```

Out[11]:

	userId	movieId	tag
0	18	4141	Mark Waters
1	65	208	dark hero
2	65	353	dark hero

In [12]:

```
row_0 = tags.iloc[0]
row_0
```

Out[12]:

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

In [13]:

```
row_0.index
```

Out[13]:

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

In [14]:

```
row_0['userId']
```

Out[14]:

```
np.int64(18)
```

In [15]:

```
'rating' in row_0
```

Out[15]:

```
False
```

In [16]:

```
row_0.name
```

Out[16]:

```
0
```

In [17]:
```
row_0=row_0.rename('firstRow')
row_0.name
```

Out[17]:
```
'firstRow'
```

In [18]:
```
tags.head()
```

Out[18]:

	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

In [19]:
```
tags.index
```

Out[19]:
```
RangeIndex(start=0, stop=465564, step=1)
```

In [20]:
```
tags.columns
```

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

In [21]:
```
tags.iloc[[0,11,500]]
```

Out[21]:

	userId	movieId	tag
0	18	4141	Mark Waters
11	65	1783	noir thriller
500	342	55908	entirely dialogue

In [22]:
```
#descriptive statistics
ratings ['rating'].describe()
```

Out[22]:
```
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 [23]:
```
ratings.describe()
```

Out[23]:

	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 [24]:

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

Out[24]:

```
0.5
```

In [25]:

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

Out[25]:

```
5.0
```

In [26]:

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

Out[26]:

```
1.051988919275684
```

In [27]:

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

Out[27]:

```
0    4.0
Name: rating, dtype: float64
```

In [28]:

```
ratings.corr()
```

Out[28]:

	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 [29]:

```
filter1=ratings['rating']>10
print(filter1)
filter1.any()
```

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

Out[29]:
```
np.False_
```

In [30]:
```
filter2=ratings['rating']>0
filter2.all()
```

Out[30]:
```
np.True_
```

In [31]:
```
#Data cleaning:handling missing data
movies.shape
```

Out[31]:
```
(27278, 3)
```

In [32]:
```
movies.isnull().any().any()
```

Out[32]:
```
np.False_
```

In [33]:
```
ratings.shape
```

Out[33]:
```
(20000263, 3)
```

In [34]:
```
ratings.isnull().any().any()
```

Out[34]:
```
np.False_
```

In [35]:
```
tags.shape
```

Out[35]:
```
(465564, 3)
```

In [36]:
```
tags.isnull().any().any()
```

Out[36]:
```
np.True_
```

In [37]:
```
tags=tags.dropna()
```

In [38]:
```
tags.isnull().any().any()
```

Out[38]:
```
np.False_
```

In [39]:
```
tags.shape
```

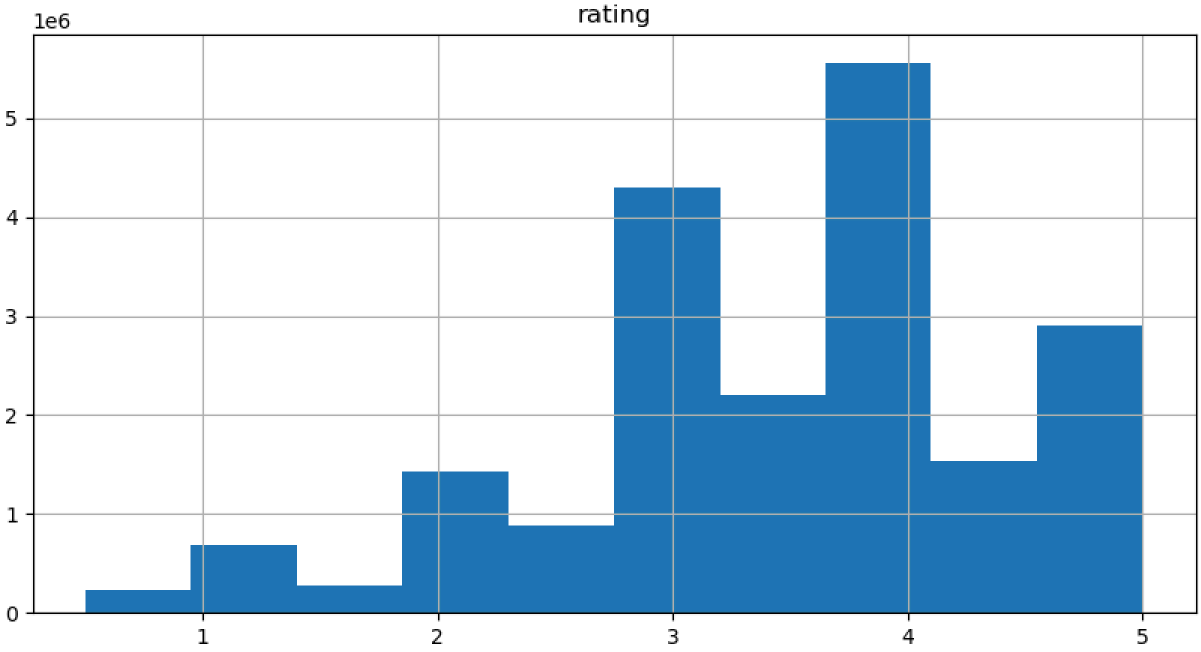
```
Out[39]:  (465548, 3)
```

Data Visualization

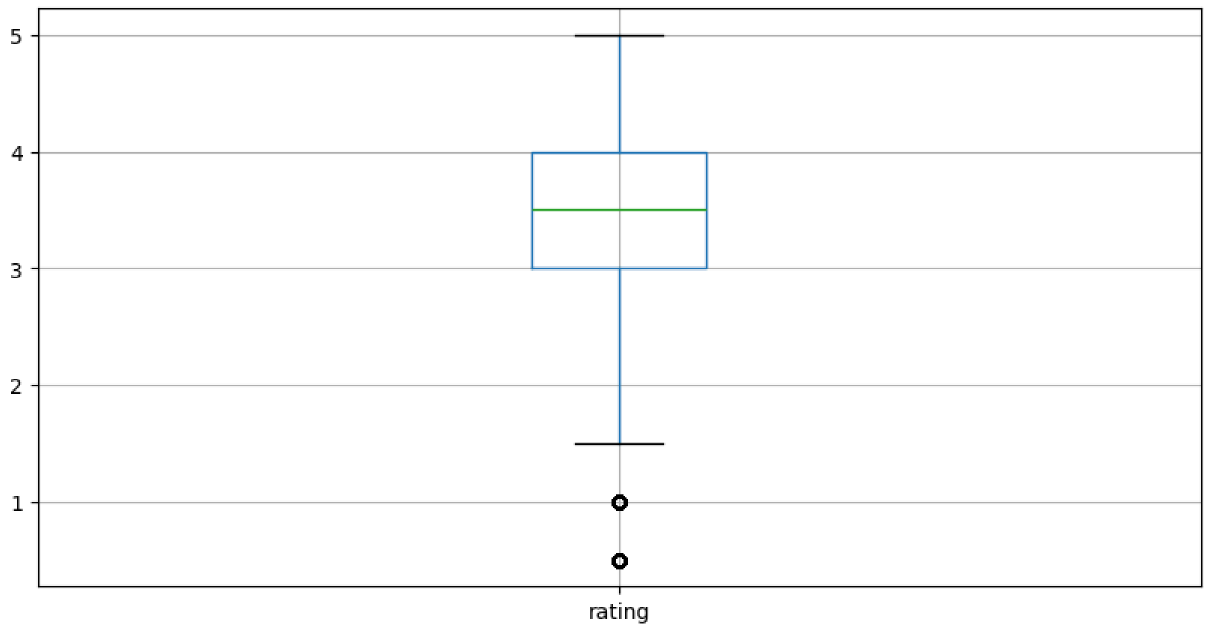
```
In [40]:  %matplotlib inline
          ratings.hist(column='rating',figsize=(10,5))
```

```
Out[40]:  array([[<Axes: title={'center': 'rating'}>]], dtype=object)
```

```
In [41]:  plt.show()
```



```
In [42]:  ratings.boxplot(column='rating',figsize=(10,5))
          plt.show()
```



Slicing out Columns

In [43]:

```
tags['tag'].head()
```

Out[43]:

```
0      Mark Waters
1        dark hero
2        dark hero
3    noir thriller
4        dark hero
Name: tag, dtype: object
```

In [44]:

```
movies[['title','genres']].head()
```

Out[44]:

	title	genres
0	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	Jumanji (1995)	Adventure|Children|Fantasy
2	Grumpier Old Men (1995)	Comedy|Romance
3	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	Father of the Bride Part II (1995)	Comedy

In [45]:

```
ratings[-10:]
```

Out[45]:

	userId	movieId	rating
20000253	138493	60816	4.5
20000254	138493	61160	4.0
20000255	138493	65682	4.5
20000256	138493	66762	4.5
20000257	138493	68319	4.5
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

In [46]:

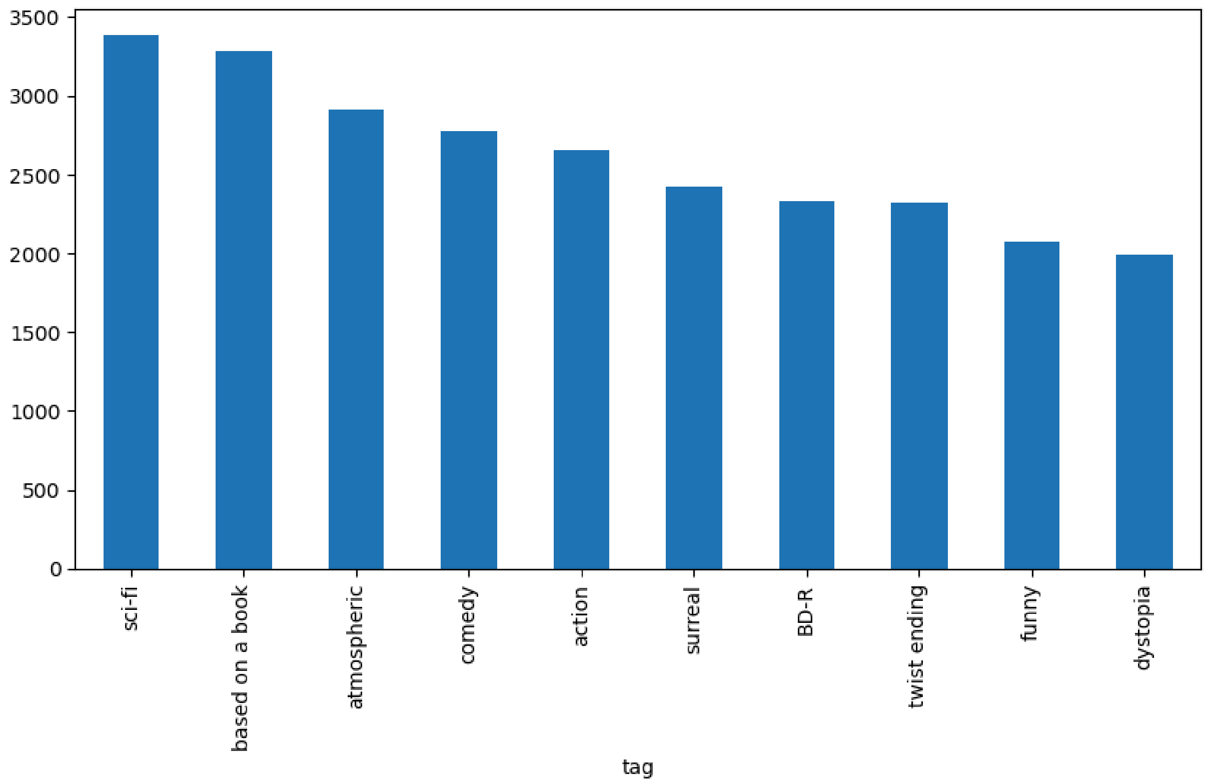
```
tags_counts=tags['tag'].value_counts()
tags_counts[-10:]
```

Out[46]:

```
tag
Hell naw                      1
This is my happy face         1
I heel toe on Uday's house    1
Why?                          1
Bobo                          1
Diamond Dallas Page           1
I'm Devon Butler!             1
No arguement                  1
Really Bad                    1
Botox                         1
Name: count, dtype: int64
```

In [47]:

```
tags_counts[:10].plot(kind='bar',figsize=(10,5))
plt.show()
```



filters for selecting rows

In [52]:
```
is_highly_rated=ratings['rating']>=5.0
ratings[is_highly_rated][30:50]
```

Out[52]:

	userId	movieId	rating
239	3	50	5.0
242	3	175	5.0
244	3	223	5.0
245	3	260	5.0
246	3	316	5.0
247	3	318	5.0
248	3	329	5.0
252	3	457	5.0
253	3	480	5.0
254	3	490	5.0
256	3	541	5.0
258	3	593	5.0
263	3	858	5.0
264	3	904	5.0
267	3	924	5.0
268	3	953	5.0
271	3	1060	5.0
272	3	1073	5.0
275	3	1084	5.0
276	3	1089	5.0

In [53]:

```
is_action=movies['genres'].str.contains('Action')
movies[is_action][5:15]
```

Out[53]:

	movieId	title	genres
22	23	Assassins (1995)	Action|Crime|Thriller
41	42	Dead Presidents (1995)	Action|Crime|Drama
43	44	Mortal Kombat (1995)	Action|Adventure|Fantasy
50	51	Guardian Angel (1994)	Action|Drama|Thriller
65	66	Lawnmower Man 2: Beyond Cyberspace (1996)	Action|Sci-Fi|Thriller
69	70	From Dusk Till Dawn (1996)	Action|Comedy|Horror|Thriller
70	71	Fair Game (1995)	Action
75	76	Screamers (1995)	Action|Sci-Fi|Thriller
77	78	Crossing Guard, The (1995)	Action|Crime|Drama|Thriller
85	86	White Squall (1996)	Action|Adventure|Drama

In [54]:

```
movies[is_action].head(15)
```

Out[54]:

	movieId	title	genres
5	6	Heat (1995)	Action|Crime|Thriller
8	9	Sudden Death (1995)	Action
9	10	GoldenEye (1995)	Action|Adventure|Thriller
14	15	Cutthroat Island (1995)	Action|Adventure|Romance
19	20	Money Train (1995)	Action|Comedy|Crime|Drama|Thriller
22	23	Assassins (1995)	Action|Crime|Thriller
41	42	Dead Presidents (1995)	Action|Crime|Drama
43	44	Mortal Kombat (1995)	Action|Adventure|Fantasy
50	51	Guardian Angel (1994)	Action|Drama|Thriller
65	66	Lawnmower Man 2: Beyond Cyberspace (1996)	Action|Sci-Fi|Thriller
69	70	From Dusk Till Dawn (1996)	Action|Comedy|Horror|Thriller
70	71	Fair Game (1995)	Action
75	76	Screamers (1995)	Action|Sci-Fi|Thriller
77	78	Crossing Guard, The (1995)	Action|Crime|Drama|Thriller
85	86	White Squall (1996)	Action|Adventure|Drama

Group by and aggregate

In [56]:

```
ratings_count=ratings[['movieId','rating']].groupby('rating').count()
ratings_count
```

Out[56]:

	movieId
rating	
0.5	239125
1.0	680732
1.5	279252
2.0	1430997
2.5	883398
3.0	4291193
3.5	2200156
4.0	5561926
4.5	1534824
5.0	2898660

In [58]:

```
average_rating=ratings[['movieId','rating']].groupby('movieId').mean()
average_rating.head()
```

Out[58]:

	rating
movieId	
1	3.921240
2	3.211977
3	3.151040
4	2.861393
5	3.064592

In [59]:

```
movie_count=ratings[['movieId','rating']].groupby('movieId').count()
movie_count.head()
```

Out[59]:

movieId	rating
1	49695
2	22243
3	12735
4	2756
5	12161

In [60]:
```
movie_count=ratings[['movieId','rating']].groupby('movieId').count()
movie_count.tail()
```

Out[60]:

movieId	rating
131254	1
131256	1
131258	1
131260	1
131262	1

Marge Dataframes

In [61]:
```
tags.head()
```

Out[61]:

	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

In [62]:
```
movies.head()
```

Out[62]:

	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

In [68]:

```
t=movies.merge(tags,on='movieId',how='inner')
t.head()
```

Out[68]:

	movieId	title	genres	userId	tag
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	1644	Watched
1	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	1741	computer animation
2	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	1741	Disney animated feature
3	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	1741	Pixar animation
4	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	1741	TÃ©a Leoni does not star in this movie

Combine aggreation,merging and filters to get useful analytics

In [71]:

```
avg_ratings=ratings.groupby('movieId',as_index=False).mean()
del avg_ratings['userId']
avg_ratings.head()
```

Out[71]:

	movieId	rating
0	1	3.921240
1	2	3.211977
2	3	3.151040
3	4	2.861393
4	5	3.064592

In [74]:

```
box_office=movies.merge(avg_ratings,on='movieId',how='inner')
box_office.tail()
```

Out[74]:

	movieId	title	genres	rating
26739	131254	Kein Bund für's Leben (2007)	Comedy	4.0
26740	131256	Feuer, Eis & Dosenbier (2002)	Comedy	4.0
26741	131258	The Pirates (2014)	Adventure	2.5
26742	131260	Rentun Ruusu (2001)	(no genres listed)	3.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	4.0

In [75]:

```
is_highly_rated = box_office['rating'] >= 4.0
box_office[is_highly_rated][-5:]
```

Out[75]:

	movieId	title	genres	rating
26737	131250	No More School (2000)	Comedy	4.0
26738	131252	Forklift Driver Klaus: The First Day on the Jo...	Comedy|Horror	4.0
26739	131254	Kein Bund für's Leben (2007)	Comedy	4.0
26740	131256	Feuer, Eis & Dosenbier (2002)	Comedy	4.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	4.0

In [76]:

```
is_Adventure = box_office['genres'].str.contains('Adventure')
box_office[is_Adventure][:5]
```

Out[76]:

	movieId	title	genres	rating
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	3.921240
1	2	Jumanji (1995)	Adventure|Children|Fantasy	3.211977
7	8	Tom and Huck (1995)	Adventure|Children	3.142049
9	10	GoldenEye (1995)	Action|Adventure|Thriller	3.430029
12	13	Balto (1995)	Adventure|Animation|Children	3.272416

In [78]:

```
box_office[is_Adventure & is_highly_rated][-5:]
```

Out[78]:

	movieId	title	genres	rating
26611	130586	Itinerary of a Spoiled Child (1988)	Adventure|Drama	4.5
26655	130996	The Beautiful Story (1992)	Adventure|Drama|Fantasy	5.0
26667	131050	Stargate SG-1 Children of the Gods - Final Cut...	Adventure|Sci-Fi|Thriller	5.0
26736	131248	Brother Bear 2 (2006)	Adventure|Animation|Children|Comedy|Fantasy	4.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	4.0

Vectorized string Operation

In [79]:

```
movies.head()
```

Out[79]:

	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

In [84]:

```
#split genrs into multiple columns
movies_genres=movies['genres'].str.split('|',expand=True)
movies_genres[:10]
```

Out[84]:

	0	1	2	3	4	5	6	7	8	9
0	Adventure	Animation	Children	Comedy	Fantasy	None	None	None	None	None
1	Adventure	Children	Fantasy	None	None	None	None	None	None	None
2	Comedy	Romance	None	None	None	None	None	None	None	None
3	Comedy	Drama	Romance	None	None	None	None	None	None	None
4	Comedy	None	None	None	None	None	None	None	None	None
5	Action	Crime	Thriller	None	None	None	None	None	None	None
6	Comedy	Romance	None	None	None	None	None	None	None	None
7	Adventure	Children	None	None	None	None	None	None	None	None
8	Action	None	None	None	None	None	None	None	None	None
9	Action	Adventure	Thriller	None	None	None	None	None	None	None

In [86]:

```
#add a new column for comedy genre flag
movies_genres['isComedy']=movies['genres'].str.contains('Comedy')
movies_genres[:10]
```

Out[86]:

	0	1	2	3	4	5	6	7	8	9	isC
0	Adventure	Animation	Children	Comedy	Fantasy	None	None	None	None	None	
1	Adventure	Children	Fantasy	None	None	None	None	None	None	None	
2	Comedy	Romance	None	None	None	None	None	None	None	None	
3	Comedy	Drama	Romance	None	None	None	None	None	None	None	
4	Comedy	None	None	None	None	None	None	None	None	None	
5	Action	Crime	Thriller	None	None	None	None	None	None	None	
6	Comedy	Romance	None	None	None	None	None	None	None	None	
7	Adventure	Children	None	None	None	None	None	None	None	None	
8	Action	None	None	None	None	None	None	None	None	None	
9	Action	Adventure	Thriller	None	None	None	None	None	None	None	

In [89]:

```
#extract year from title
movies['year']=movies['title'].str.extract('.*\((.*)\).*',expand=True)
movies.tail()
```

Out[89]:

	movieId	title	genres	year
27273	131254	Kein Bund für's Leben (2007)	Comedy	2007
27274	131256	Feuer, Eis & Dosenbier (2002)	Comedy	2002
27275	131258	The Pirates (2014)	Adventure	2014
27276	131260	Rentun Ruusu (2001)	(no genres listed)	2001
27277	131262	Innocence (2014)	Adventure|Fantasy|Horror	2014

In [90]:

```
tags.dtypes
```

Out[90]:

```
userId       int64
movieId      int64
tag         object
dtype: object
```

In [92]:

```
tags.head(10)
```

Out[92]:

	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
5	65	668	bollywood
6	65	898	screwball comedy
7	65	1248	noir thriller
8	65	1391	mars
9	65	1617	neo-noir

In [95]:

```
average_rating=ratings[['movieId','rating']].groupby('movieId',as_index=False).mean
average_rating.tail()
```

Out[95]:

	movieId	rating
26739	131254	4.0
26740	131256	4.0
26741	131258	2.5
26742	131260	3.0
26743	131262	4.0

In [103…

```
joined=movies.merge(average_rating,on='movieId',how='inner')
joined.head()
```

Out[103…

	movieId	title	genres	year	rating
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	1995	3.921240
1	2	Jumanji (1995)	Adventure|Children|Fantasy	1995	3.211977
2	3	Grumpier Old Men (1995)	Comedy|Romance	1995	3.151040
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance	1995	2.861393
4	5	Father of the Bride Part II (1995)	Comedy	1995	3.064592

In [101…

```
average_rating.corr()
```

Out[101…

	movieId	rating
movieId	1.000000	-0.090369
rating	-0.090369	1.000000

In [ ]:

```
```