IMPORT THE LIBRARY

In [1]:
```
import pandas as pd #import the libarry
```

READ THE DATASET

In [2]:
```
movie = pd.read_csv(r'C:\Users\91996\Desktop\python\M movie.csv.csv')
```

In [3]:
```
movie
```

Out[3]:

	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 [4]:
```
movie.head()
```

Out[4]:

	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 [5]:
```
tags = pd.read_csv(r'C:\Users\91996\Desktop\python\M tag.csv.csv')
```

In [6]:
```
tags.head(1)
```

Out[6]:

	userId	movieId	tag	timestamp
0	18	4141	Mark Waters	2009-04-24 18:19:40

In [7]:
```
tags.info
```

Out[7]:
```
<bound method DataFrame.info of         userId  movieId             tag             timest
amp
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 x 4 columns]>
```

In [8]:
```
rating=pd.read_csv(r'C:\Users\91996\Desktop\python\M rating.csv.csv')
```

In [9]:
```
rating.head(1)
```

Out[9]:

	userId	movieId	rating	timestamp
0	1	2	3.5	2005-04-02 23:53:47

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

Out[10]:
```
(20000263, 4)
```

In [11]:
```
rating.describe()
```

Out[11]:

	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 [12]:
```
movie.columns
```

Out[12]:
```
Index(['movieId', 'title', 'genres'], dtype='object')
```

In [13]:
```
rating.columns
```

Out[13]:
```
Index(['userId', 'movieId', 'rating', 'timestamp'], dtype='object')
```

In [14]:
```
tags.columns
```

Out[14]:
```
Index(['userId', 'movieId', 'tag', 'timestamp'], dtype='object')
```

In [15]:
```
del rating['timestamp']
```

In [16]:
```
rating.columns
```

Out[16]:
```
Index(['userId', 'movieId', 'rating'], dtype='object')
```

In [17]:
```
len(rating.columns)
```

Out[17]:
```
3
```

In [18]:
```
del tags['timestamp']
```

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

Out[19]:
```
Index(['userId', 'movieId', 'tag'], dtype='object')
```

DATA STRUCTURES

In [20]:
```
tags.head()
```

Out[20]:

	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 [21]:
```
row_0 = tags.iloc[0]
```

In [22]:
```
row_0
```

Out[22]:
```
userId              18
movieId           4141
tag        Mark Waters
Name: 0, dtype: object
```

In [23]:
```
type(row_0)
```

Out[23]:
```
pandas.core.series.Series
```

In [24]:
```
row_0.index
```

Out[24]:
```
Index(['userId', 'movieId', 'tag'], dtype='object')
```

In [25]:
```
row_0['userId']
```

Out[25]:
```
18
```

In [26]:
```
movie['movieId']
```

Out[26]:
```
0             1
1             2
2             3
3             4
4             5
```

```
          ...
27273    131254
27274    131256
27275    131258
27276    131260
27277    131262
Name: movieId, Length: 27278, dtype: int64
```

In [27]:
```
'tag' in row_0
```

Out[27]:
```
True
```

In [28]:
```
row_0.tag
```

Out[28]:
```
'Mark Waters'
```

In [29]:
```
tags.iloc[ [0,11,300] ]
```

Out[29]:

	userId	movieId	tag
0	18	4141	Mark Waters
11	65	1783	noir thriller
300	316	45186	Ethan Hunt Should Stop Hogging The Screen!

DESCRIPTIVE STATS

In [30]:
```
movie.describe
```

Out[30]:
```
<bound method NDFrame.describe of        movieId                               title  \
0            1                    Toy Story (1995)
1            2                      Jumanji (1995)
2            3             Grumpier Old Men (1995)
3            4            Waiting to Exhale (1995)
4            5  Father of the Bride Part II (1995)
...        ...                                 ...
27273   131254        Kein Bund für's Leben (2007)
27274   131256       Feuer, Eis & Dosenbier (2002)
27275   131258                  The Pirates (2014)
27276   131260                 Rentun Ruusu (2001)
27277   131262                     Innocence (2014)

                                       genres
0      Adventure|Animation|Children|Comedy|Fantasy
1                       Adventure|Children|Fantasy
2                                    Comedy|Romance
3                              Comedy|Drama|Romance
4                                           Comedy
...                                            ...
27273                                       Comedy
27274                                       Comedy
27275                                    Adventure
27276                          (no genres listed)
27277                       Adventure|Fantasy|Horror

[27278 rows x 3 columns]>
```

In [31]:
```
rating.describe
```

Out[31]:
```
<bound method NDFrame.describe of           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 x 3 columns]>
```

In [48]:
```
rating.mean()
```

Out[48]:
```
userId     69045.872583
movieId     9041.567330
rating         3.525529
dtype: float64
```

In [33]:
```
rating['rating'].mean()
```

Out[33]:
```
3.5255285642993797
```

In [34]:
```
rating['rating'].min()
```

Out[34]:
```
0.5
```

In [36]:
```
rating['rating'].max()
```

Out[36]:
```
5.0
```

In [37]:
```
rating['rating'].std()
```

Out[37]:
```
1.051988919275684
```

In [38]:
```
rating['rating'].mode()
```

Out[38]:
```
0    4.0
Name: rating, dtype: float64
```

In [39]:
```
rating.corr()
```

Out[39]:

	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 [40]:
```
filter1 = rating['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[40]:
```
False
```

In [41]:
```
filter2 = rating['rating'] > 0
filter2.all()
```

Out[41]:
```
True
```

In [43]:
```
movie.shape
```

Out[43]:
```
(27278, 3)
```

Data Cleaning: Handling Missing Data

In [44]:
```
movie.isnull().any().any()
```

Out[44]:
```
False
```

In [45]:
```
rating.shape
```

Out[45]:
```
(20000263, 3)
```

In [46]:
```
rating.isnull().any().any()
```

Out[46]:
```
False
```

In [48]:
```
tags.shape
```

Out[48]:
```
(465564, 3)
```

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

Out[49]:
```
True
```

In [50]:
```
tags=tags.dropna()
```

In [51]:
```
tags.isnull().any().any()
```

Out[51]:
```
False
```

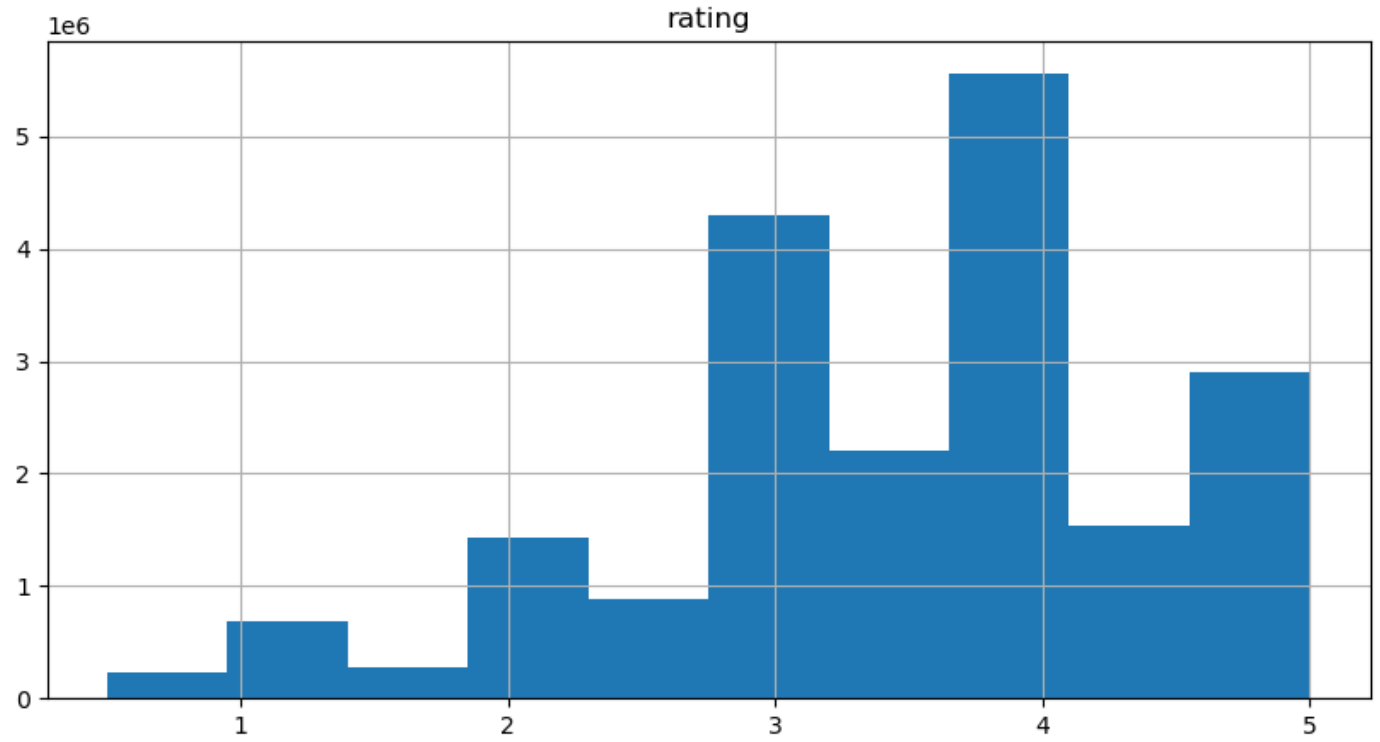
In [52]:
```
tags.shape
```

Out[52]:
```
(465548, 3)
```

Data Visualization

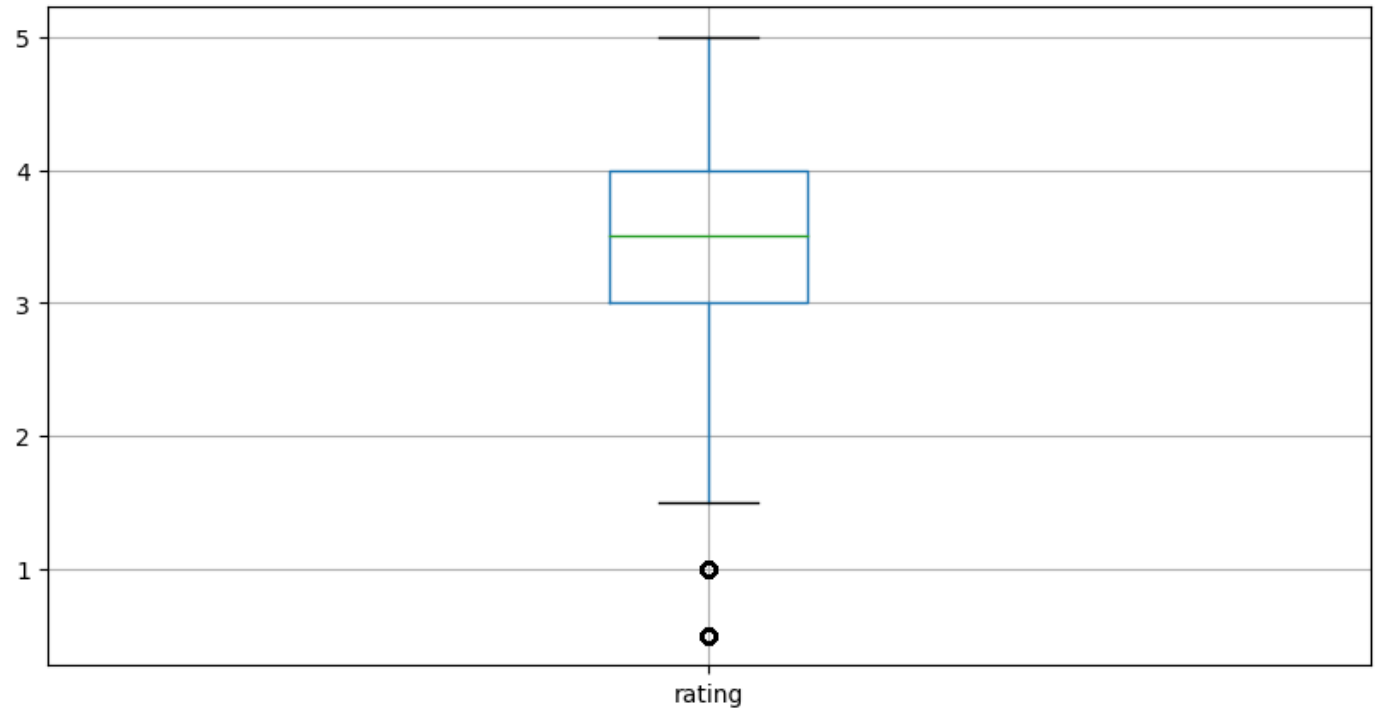
In [53]:
```
%matplotlib inline

rating.hist(column='rating', figsize=(10,5))
```

Out[53]:
```
array([[<AxesSubplot:title={'center':'rating'}>]], dtype=object)
```



In [54]:
```
rating.boxplot(column='rating', figsize=(10,5))
```

Out[54]:
```
<AxesSubplot:>
```



Slicing Out Columns

In [55]:
```
tags['tag'].head()
```

Out[55]:
```
0       Mark Waters
1         dark hero
2         dark hero
3    noir thriller
4         dark hero
Name: tag, dtype: object
```

In [56]:
```
movie[['title','genres']].head()
```

Out[56]:

	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 [57]:
```
rating[-10:]
```

Out[57]:

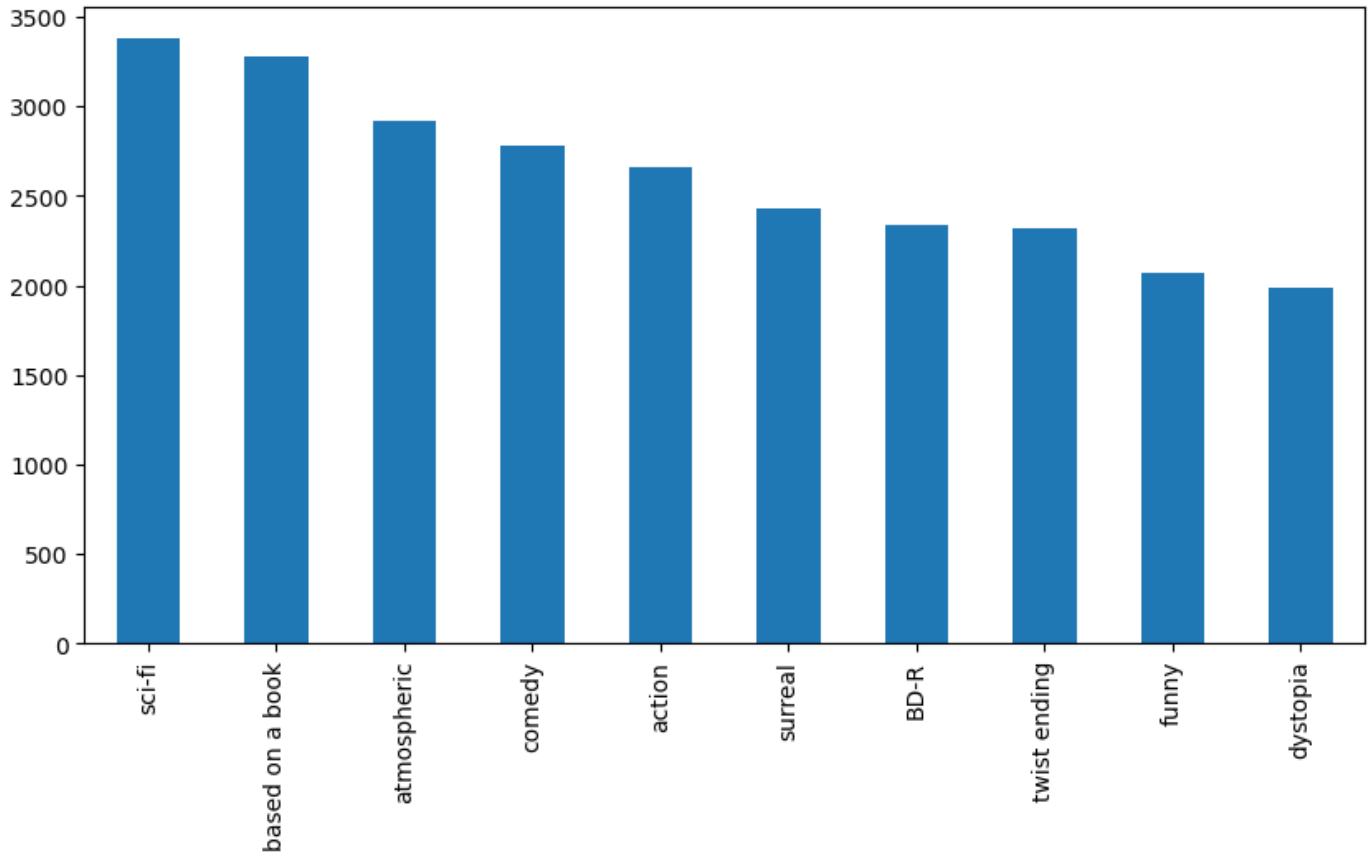
	userId	movieId	rating
20000253	138493	60816	4.5
20000254	138493	61160	4.0
20000255	138493	65682	4.5
20000256	138493	66762	4.5
20000257	138493	68319	4.5
20000258	138493	68954	4.5
20000259	138493	69526	4.5
20000260	138493	69644	3.0
20000261	138493	70286	5.0
20000262	138493	71619	2.5

In [58]:
```
tag_counts = tags['tag'].value_counts()
tag_counts[-10:]
```

Out[58]:
```
missing child                    1
Ron Moore                        1
Citizen Kane                     1
mullet                           1
biker gang                       1
Paul Adelstein                   1
the wig                          1
killer fish                      1
genetically modified monsters    1
topless scene                    1
Name: tag, dtype: int64
```

In [59]:
```
tag_counts[:10].plot(kind='bar', figsize=(10,5))
```

Out[59]:
```
<AxesSubplot:>
```



In [60]:
```
is_highly_rated = rating['rating'] >= 5.0
rating[is_highly_rated][30:50]
```

Out[60]:

	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 [61]:
```
is_action= movie['genres'].str.contains('Action')
movie[is_action][5:15]
```

Out[61]:

	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 [62]:
```
movie[is_action].head(15)
```

Out[62]:

	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 [63]:
```
ratings_count = rating[['movieId','rating']].groupby('rating').count()
ratings_count
```

Out[63]:

rating	movieId
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 [64]:
```
average_rating = rating[['movieId','rating']].groupby('movieId').mean()
average_rating.head()
```

Out[64]:

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

In [65]:
```
movie_count = rating[['movieId','rating']].groupby('movieId').count()
movie_count.head()
rating
```

Out[65]:

	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

In [67]:
```
 movie_count = rating[['movieId','rating']].groupby('movieId').count()
movie_count.tail()
```

Out[67]:

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

Merge Dataframes

In [69]:
```
tags.head()
```

Out[69]:

	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 [70]:
```
movie.head()
```

Out[70]:

	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 [71]:
```
t = movie.merge(tags, on='movieId', how='inner')
t.head()
```

Out[71]:

	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 aggreagation, merging, and filters to get useful analytics

In [74]:
```
avg_rating= rating.groupby('movieId', as_index=False).mean()
del avg_rating['userId']
avg_rating.head()
```

Out[74]:

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

In [75]:
```
box_office = movie.merge(avg_ratings, on='movieId', how='inner')
box_office.tail()
```

Out[75]:

	movieId	title	genres	userId	rating
26739	131254	Kein Bund für's Leben (2007)	Comedy	79570.0	4.0
26740	131256	Feuer, Eis & Dosenbier (2002)	Comedy	79570.0	4.0
26741	131258	The Pirates (2014)	Adventure	28906.0	2.5
26742	131260	Rentun Ruusu (2001)	(no genres listed)	65409.0	3.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	133047.0	4.0

In [76]:
```
is_highly_rated = box_office['rating'] >= 4.0
box_office[is_highly_rated][-5:]
```

Out[76]:

	movieId	title	genres	userId	rating
26737	131250	No More School (2000)	Comedy	79570.0	4.0
26738	131252	Forklift Driver Klaus: The First Day on the Jo...	Comedy|Horror	79570.0	4.0
26739	131254	Kein Bund für's Leben (2007)	Comedy	79570.0	4.0
26740	131256	Feuer, Eis & Dosenbier (2002)	Comedy	79570.0	4.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	133047.0	4.0

In [77]:
```
is_Adventure = box_office['genres'].str.contains('Adventure')
box_office[is_Adventure][:5]
```

Out[77]:

	movieId	title	genres	userId	rating
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	69282.396821	3.921240

1	2	Jumanji (1995)	Adventure|Children|Fantasy	69169.928202	3.211977
7	8	Tom and Huck (1995)	Adventure|Children	68677.092580	3.142049
9	10	GoldenEye (1995)	Action|Adventure|Thriller	69161.741045	3.430029
12	13	Balto (1995)	Adventure|Animation|Children	70136.308693	3.272416

In [78]:
```
box_office[is_Adventure & is_highly_rated][-5:]
```

Out[78]:

	movieId	title	genres	userId	rating
26611	130586	Itinerary of a Spoiled Child (1988)	Adventure|Drama	130784.0	4.5
26655	130996	The Beautiful Story (1992)	Adventure|Drama|Fantasy	31411.0	5.0
26667	131050	Stargate SG-1 Children of the Gods - Final Cut...	Adventure|Sci-Fi|Thriller	79570.0	5.0
26736	131248	Brother Bear 2 (2006)	Adventure|Animation|Children|Comedy|Fantasy	79570.0	4.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	133047.0	4.0

Vectorized String Operations

In [80]:
```
movie.head()
```

Out[80]:

	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

Split 'genres' into multiple columns

In [81]:
```
movie_genres = movie['genres'].str.split('|', expand=True)
```

In [82]:
```
movie_genres[:10]
```

Out[82]:

	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

Add a new column for comedy genre flag

In [83]:
```
movie_genres['isComedy'] = movie['genres'].str.contains('Comedy')
```

In [84]:
```
movie_genres[:10]
```

Out[84]:

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

Extract year from title e.g. (2007)

In [86]:
```
movie['year'] = movie['title'].str.extract('.*\((.*)\).*', expand=True)
```

In [88]:
```
movie.tail()
```

Out[88]:

	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 [ ]:
```

```