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

In [2]:
```
ratings=pd.read_csv(r'C:\Users\nlnar\OneDrive\Desktop\rating.csv')
```

In [3]:
```
movies=pd.read_csv(r'C:\Users\nlnar\OneDrive\Desktop\movie.csv')
```

In [4]:
```
tags=pd.read_csv(r'C:\Users\nlnar\OneDrive\Desktop\tag.csv')
```

In [5]:
```
print(tags.columns)
```

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

In [6]:
```
print(movies.columns)
```

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

In [7]:
```
print(ratings.columns)
```

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

In [8]:
```
del tags['timestamp']
```

In [9]:
```
del ratings['timestamp']
```

In [10]:
```
tags.head()
```

Out[10]:

	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 [11]:
```
ratings.head()
```

Out[11]:

	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

In [12]:
```
rows_0=tags.iloc[0]
```

In [13]:
```
print(rows_0)
```

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

In [14]:
```
type(rows_0)
```

Out[14]: pandas.core.series.Series

In [15]:
```
rows_0.index
```

Out[15]: Index(['userId', 'movieId', 'tag'], dtype='object')

In [16]:
```
rows_0['userId']
```

Out[16]: np.int64(18)

In [17]:
```
'ratings' in rows_0
```

Out[17]: False

In [18]:
```
'movieId' in rows_0
```

Out[18]: True

In [19]:
```
rows_0.name
```

Out[19]: 0

In [20]:
```
rows_0=tags.iloc[1]
```

In [21]:
```
rows_0.name
```

Out[21]: 1

In [26]:
```
rows_0=rows_0.rename('first row')
```

In [25]:
```
rows_0.name
```

Out[25]: 'first row'

In [24]:
```
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')
```

data frames

In [27]:
```
tags.head()
```

Out[27]:

	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 [28]:
```
movies.head()
```

Out[28]:

	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 [29]:
```
ratings.head()
```

Out[29]:

	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

In [30]:
```
tags.index
```

Out[30]:
```
RangeIndex(start=0, stop=465564, step=1)
```

In [32]:
```
tags.columns
```

Out[32]:
```
Index(['userId', 'movieId', 'tag'], dtype='object')
```

In [33]:
```
tags.iloc[[0,11,500]]
```

Out[33]:

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

In [34]:
```
tags.iloc[[1,12,499]]
```

Out[34]:

	userId	movieId	tag
1	65	208	dark hero
12	65	2022	jesus
499	342	55442	social commentary

DESCRIPTIVE STATISTICS

In [35]:
```
ratings['rating'].describe()
```

Out[35]:
```
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 [36]:
```
ratings.describe()
```

Out[36]:

	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 [37]:
```
ratings['rating'].mean()
```

Out[37]:
```
np.float64(3.5255285642993797)
```

In [39]:
```
ratings.mean()
```

Out[39]:
```
userId     69045.872583
movieId     9041.567330
rating         3.525529
dtype: float64
```

In [40]:
```
ratings['rating'].min()
```

Out[40]:
```
0.5
```

In [41]:
```
ratings['rating'].max()
```

Out[41]:
```
5.0
```

In [42]:
```
ratings['rating'].std()
```

Out[42]:
```
1.051988919275684
```

In [44]:
```
ratings['rating'].mode()
```

Out[44]:
```
0    4.0
Name: rating, dtype: float64
```

In [45]:
```
ratings.corr()
```

Out[45]:

	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 [46]:
```
filter1=ratings['rating']>10
print(filter1)
```

```
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
```

In [48]:
```
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[48]: np.False_

In [49]:
```
filter2=ratings['rating']<10
print(filter2)
```

```
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 [50]:
```
filter2=ratings['rating']<10
print(filter2)
filter2.any()
```

```
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
```

Out[50]: np.True_

In [51]:
```
filter2=ratings['rating']<10
print(filter2)
filter2.all()
```

```
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
```

Out[51]: np.True_

DATA HANDLING

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

Out[52]: (27278, 3)

In [53]:
```
movies.isnull()
```

Out[53]:

	movieId	title	genres
0	False	False	False
1	False	False	False
2	False	False	False
3	False	False	False
4	False	False	False
...	...	...	...
27273	False	False	False
27274	False	False	False
27275	False	False	False
27276	False	False	False
27277	False	False	False

27278 rows × 3 columns

In [54]:
```
movies.isna()
```

Out[54]:

	movieId	title	genres
0	False	False	False
1	False	False	False
2	False	False	False
3	False	False	False
4	False	False	False
...	...	...	...
27273	False	False	False
27274	False	False	False
27275	False	False	False
27276	False	False	False
27277	False	False	False

27278 rows × 3 columns

In [55]:
```
movies.isnull().any()
```

Out[55]:
```
movieId    False
title      False
genres     False
dtype: bool
```

In [56]:
```
movies.isnull().any().any()
```

Out[56]:
```
np.False_
```

In [57]:
```
ratings.shape
```

Out[57]:
```
(20000263, 3)
```

In [58]:
```
ratings.isnull().any().any()
```

Out[58]:
```
np.False_
```

In [59]:
```
tags.shape
```

Out[59]:
```
(465564, 3)
```

In [60]:
```
tags.isnull().any().any()
```

Out[60]:
```
np.True_
```

In [62]:
```
tags=tags.dropna()
```

In [63]:
```
tags
```

Out[63]:

	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

465548 rows × 3 columns

In [64]:
```
tags.isnull().any().any()
```

Out[64]:
```
np.False_
```

In [65]:
```
tags.shape
```

Out[65]:
```
(465548, 3)
```

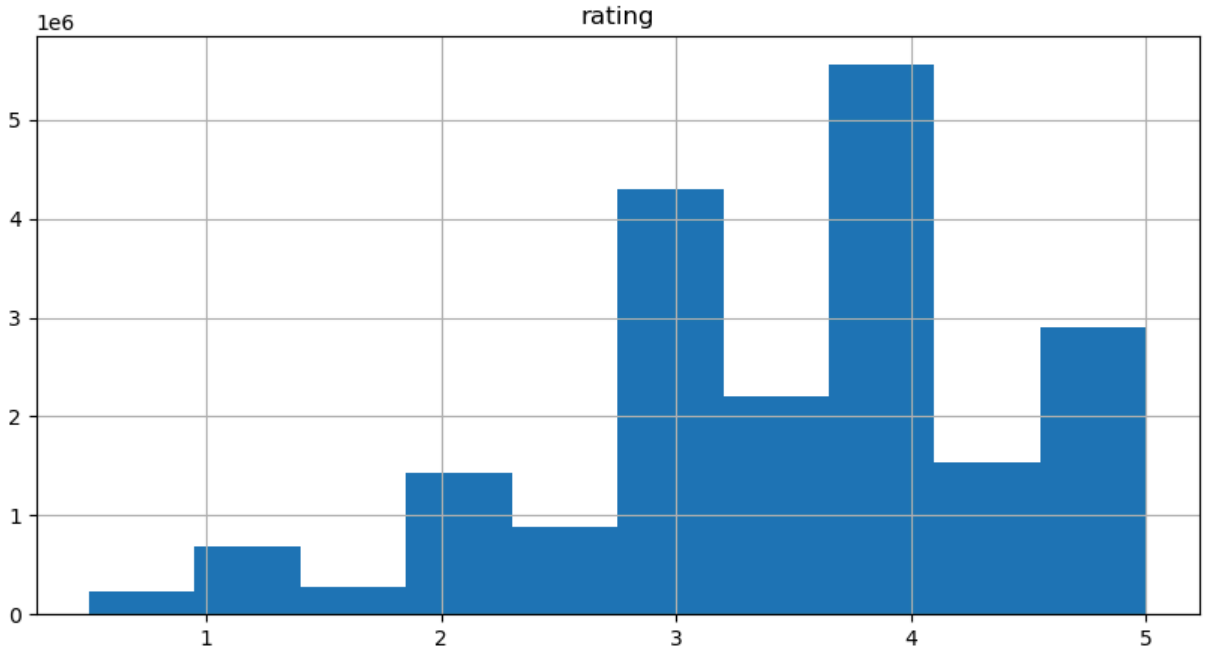
DATA VISUALISATION

In [66]:
```
import matplotlib.pyplot as plt
%matplotlib inline
```

In [68]:
```
ratings.hist(column='rating',figsize=(10,5))
```

Out[68]:
```
array([[<Axes: title={'center': 'rating'}>]], dtype=object)
```

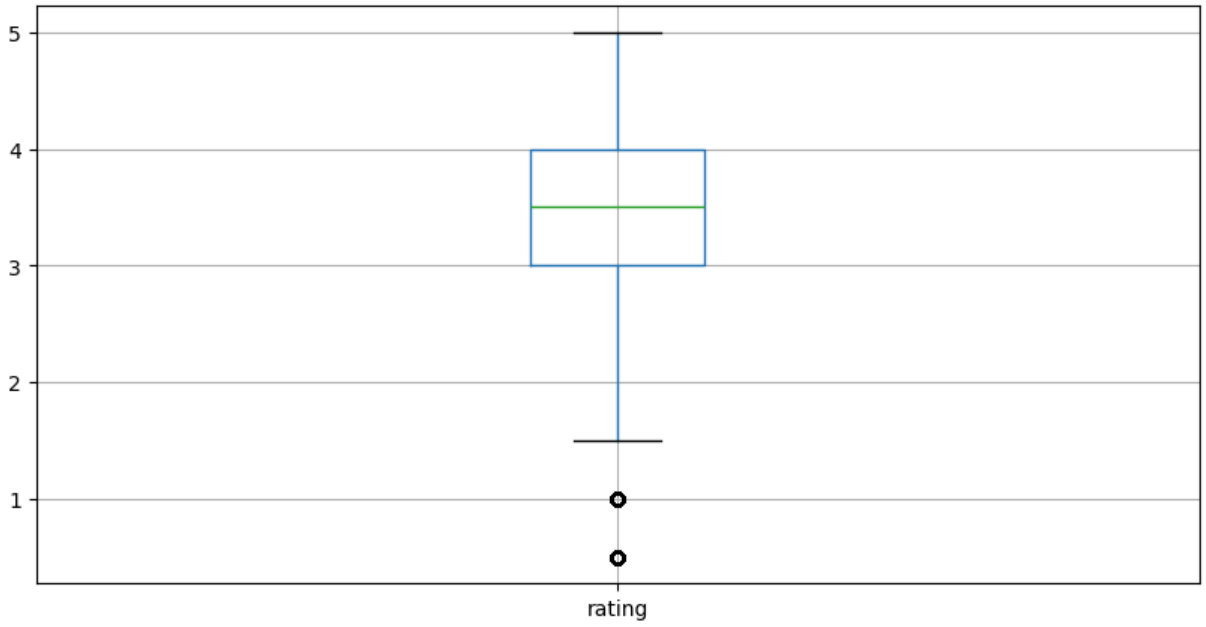
In [69]:
```
plt.show()
```



In [72]:
```
ratings.boxplot(column='rating',figsize=(10,5))
```

Out[72]: <Axes: >

In [73]:
```
plt.show()
```



SLICING OUT COLUMNS

In [75]:
```
tags['tag'].head()
```

Out[75]:
```
0      Mark Waters
1        dark hero
2        dark hero
3    noir thriller
4        dark hero
Name: tag, dtype: object
```

In [76]:
```
movies[['title','genres']].head()
```

Out[76]:

	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 [78]:
```
ratings[-0:1]
```

Out[78]:

	userId	movieId	rating
0	1	2	3.5

In [79]:
```
ratings[-10:]
```

Out[79]:

	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

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

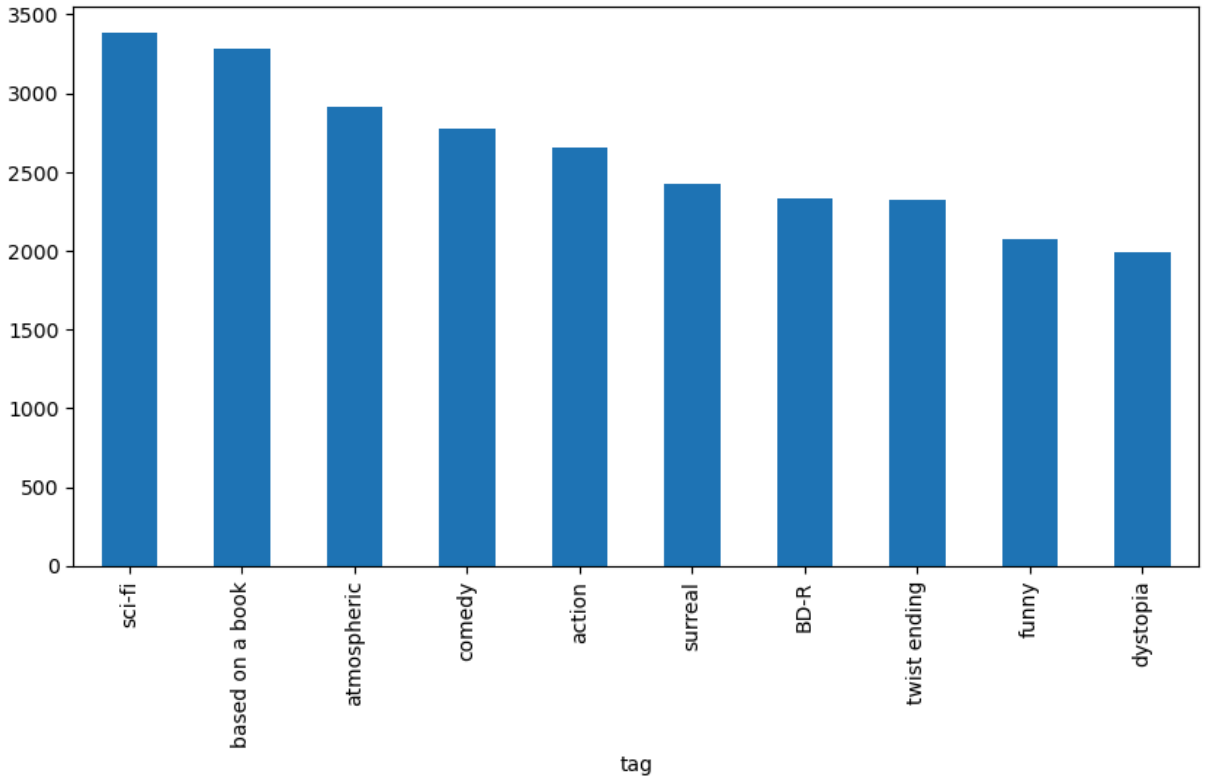
In [80]:
```
tag_counts = tags['tag'].value_counts()
tag_counts[-10:]
```

Out[80]:
```
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 [81]:
```
tag_counts[:10].plot(kind='bar', figsize=(10,5))
```

Out[81]:
```
<Axes: xlabel='tag'>
```

In [82]:
```
plt.show()
```



In [ ]: