

```
In [1]: import pandas as pd
import warnings
warnings.filterwarnings("ignore")
```

```
In [2]: data=pd.read_csv("/home/placenet/Downloads/movies.csv")
```

```
In [3]: data.describe()
```

Out[3]:

	srno	year	rating	time
count	49590.000000	49590.000000	10814.000000	45836.000000
mean	24795.500000	2002.303428	3.451248	2628.445436
std	14315.544261	12.534555	0.495601	1604.646265
min	1.000000	1913.000000	1.400000	52.000000
25%	12398.250000	1999.000000	3.100000	1356.000000
50%	24795.500000	2007.000000	3.500000	2563.000000
75%	37192.750000	2010.000000	3.800000	2877.000000
max	49590.000000	2014.000000	4.500000	28813.000000

```
In [4]: data.head(5)
```

Out[4]:

	srno	movie	year	rating	time
0	1	The Nightmare Before	1993	3.9	4568.0
1	2	The Mummy	1932	3.5	4388.0
2	3	Orphans of the Storm	1921	3.2	9062.0
3	4	The Object of Beauty	1991	2.8	6150.0
4	5	Night Tide	1963	2.8	5126.0

```
In [5]: data1=data.loc[(data.time > 5000)]
data1
```

Out[5]:

	srno	movie	year	rating	time	
	2	3	Orphans of the Storm	1921	3.2	9062.0
	3	4	The Object of Beauty	1991	2.8	6150.0
	4	5	Night Tide	1963	2.8	5126.0
	5	6	One Magic Christmas	1985	3.8	5333.0
	6	7	Muriel's Wedding	1994	3.5	6323.0

49564	49565	American Addict	2013	3.5	5377.0	
49579	49580	Underground: The Julian Assange Story	2012	3.7	5665.0	
49583	49584	Sunset Strip	2012	3.0	5770.0	
49584	49585	Silver Bells	2013	3.5	5287.0	
49586	49587	Top Gear: Series 19: Africa Special	2013	NaN	6822.0	

5897 rows × 5 columns

```
In [6]: data2=data.loc[(data.year > 2000)&(data.time >5000)]
data2
```

Out[6]:

	srno	movie	year	rating	time
438	439	Monkeybone	2001	3.1	5561.0
507	508	Impostor	2001	3.4	6143.0
514	515	Hannibal	2001	3.6	7881.0
523	524	Blow Dry	2001	3.3	5420.0
541	542	Along Came a Spider	2001	3.7	6184.0
...
49564	49565	American Addict	2013	3.5	5377.0
49579	49580	Underground: The Julian Assange Story	2012	3.7	5665.0
49583	49584	Sunset Strip	2012	3.0	5770.0
49584	49585	Silver Bells	2013	3.5	5287.0
49586	49587	Top Gear: Series 19: Africa Special	2013	NaN	6822.0

3954 rows × 5 columns

```
In [7]: data3=data.loc[(data.year >= 2000)&(data.year <= 2010)&(data.rating >=4)]  
data3
```

Out[7]:

	srno	movie	year	rating	time
484	485	Coming to Light: Edward S. Curtis and the Nort...	2000	4.0	5027.0
560	561	The Emperor's New Groove	2000	4.0	4703.0
816	817	Dil Chahta Hai	2001	4.0	11110.0
865	866	Black Hawk Down	2001	4.0	8660.0
920	921	We Were Soldiers	2002	4.0	8309.0
...
48784	48785	King Tut Unwrapped	2010	4.0	NaN
48854	48855	Pit Bulls & Parolees	2009	4.3	NaN
48874	48875	Brew Masters	2010	4.1	NaN
49025	49026	Cake Boss: Next Great Baker	2010	4.1	NaN
49028	49029	Police Women of Dallas	2010	4.0	NaN

725 rows × 5 columns

```
In [8]: datat=data.sort_values('time')
        datat
```

Out[8]:

	srno	movie	year	rating	time
40150	40151	Trailer: Pain	2012	3.6	52.0
41081	41082	Trailer: Get to Work	2012	3.3	55.0
41082	41083	Trailer: Give and Take	2012	3.3	66.0
43166	43167	Trailer: Emperor	2013	3.1	67.0
43330	43331	Trailer: Blood Angel	2013	4.2	69.0
...
49556	49557	Shinobi Girl	2012	2.0	NaN
49561	49562	My Hope America with Billy Graham	2013	3.9	NaN
49565	49566	My Hope America with Billy Graham	2013	3.9	NaN
49588	49589	Kate Plus Ei8ht	2010	2.7	NaN
49589	49590	Kate Plus Ei8ht: Season 1	2010	2.7	NaN

49590 rows × 5 columns

In [9]: `datat.head(10)`

Out[9]:

	srno	movie	year	rating	time
40150	40151	Trailer: Pain	2012	3.6	52.0
41081	41082	Trailer: Get to Work	2012	3.3	55.0
41082	41083	Trailer: Give and Take	2012	3.3	66.0
43166	43167	Trailer: Emperor	2013	3.1	67.0
43330	43331	Trailer: Blood Angel	2013	4.2	69.0
40149	40150	Trailer: Lift the Veil	2012	3.6	69.0
40789	40790	Trailer: Gray Area	2012	3.7	70.0
40154	40155	Trailer: Cleave	2012	3.5	71.0
43168	43169	Trailer: Masks	2013	4.2	73.0
45745	45746	Trailer: Strip Search	2013	3.6	73.0

```
In [10]: data=data.sort_values('rating')
data
```

Out[10]:

	srno	movie	year	rating	time
40934	40935	Lagegi	2007	1.4	NaN
42115	42116	Sun Yaar Chill Maar	2007	1.4	NaN
40826	40827	Lagegi	2007	1.4	NaN
42160	42161	Sun Yaar Chill Maar	2007	1.4	NaN
41396	41397	Meri Toh Lag Gayi Naukri	2011	1.5	NaN
...
49563	49564	My Hope America with Billy Graham: Lose to Gain	2013	NaN	1400.0
49576	49577	Barbie: Life in the Dreamhouse: Barbie Life in...	2013	NaN	1390.0
49577	49578	Barbie: Life in the Dreamhouse: Barbie Life in...	2013	NaN	1458.0
49586	49587	Top Gear: Series 19: Africa Special	2013	NaN	6822.0
49587	49588	Fireplace For Your Home: Crackling Fireplace w...	2010	NaN	3610.0

49590 rows × 5 columns

```
In [11]: data.isna().sum()
```

```
Out[11]: srno      0
movie      0
year       0
rating    38776
time      3754
dtype: int64
```

```
In [12]: data3=data.fillna(data.median())
```

```
In [13]: data.shape
```

```
Out[13]: (49590, 5)
```

```
In [14]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 49590 entries, 0 to 49589  
Data columns (total 5 columns):  
#   Column  Non-Null Count  Dtype  
---  -  
0   srno     49590 non-null    int64  
1   movie    49590 non-null    object  
2   year     49590 non-null    int64  
3   rating   10814 non-null    float64  
4   time     45836 non-null    float64  
dtypes: float64(2), int64(2), object(1)  
memory usage: 1.9+ MB
```

above line is to check types of columns

```
In [15]: data=data.groupby(['year']).sum()
```



```
In [16]: data4=data.groupby(['year']).count()  
data4
```

Out[16]:

	srno	rating	time
year			
1913	1	1	1
1914	1	1	1
1915	1	1	1
1916	1	1	1
1918	1	1	1
...
2010	1	1	1
2011	1	1	1
2012	1	1	1
2013	1	1	1
2014	1	1	1

101 rows × 3 columns

```
In [17]: data2.to_csv('movies1.csv')
```

In [18]: data2

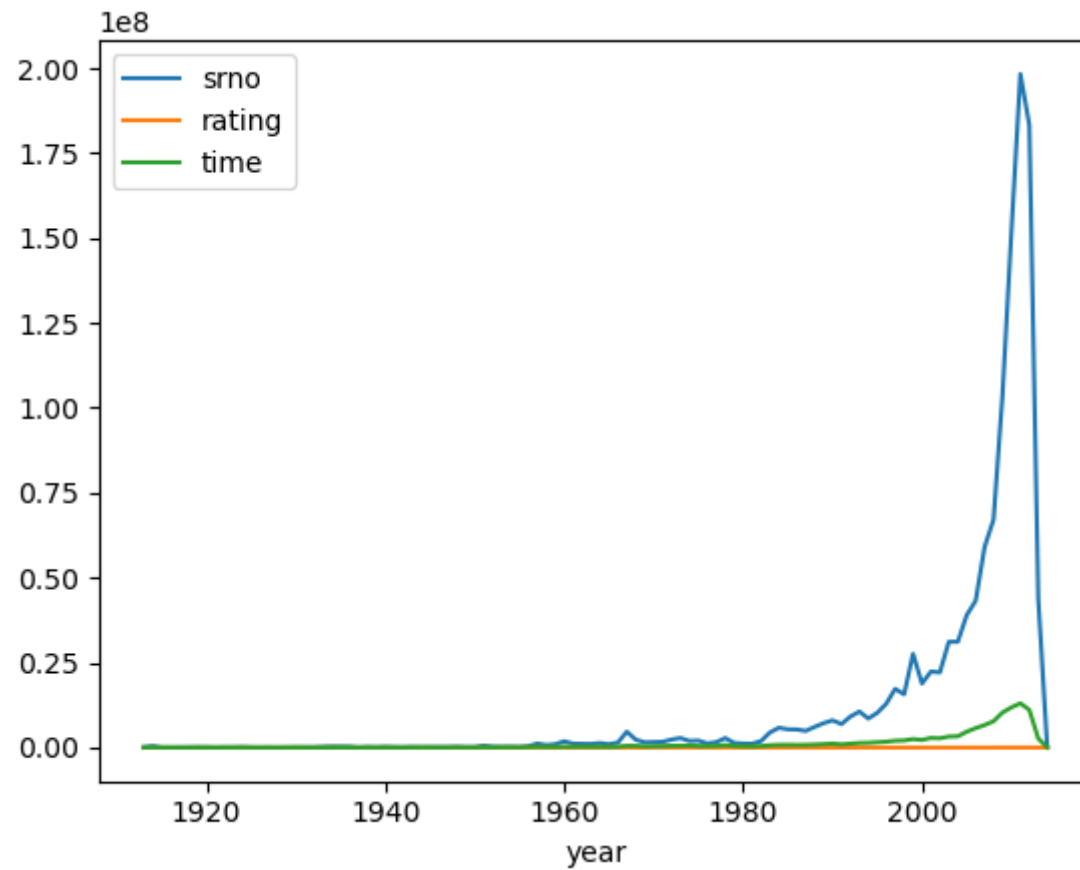
Out[18]:

	srno	movie	year	rating	time
438	439	Monkeybone	2001	3.1	5561.0
507	508	Impostor	2001	3.4	6143.0
514	515	Hannibal	2001	3.6	7881.0
523	524	Blow Dry	2001	3.3	5420.0
541	542	Along Came a Spider	2001	3.7	6184.0
...
49564	49565	American Addict	2013	3.5	5377.0
49579	49580	Underground: The Julian Assange Story	2012	3.7	5665.0
49583	49584	Sunset Strip	2012	3.0	5770.0
49584	49585	Silver Bells	2013	3.5	5287.0
49586	49587	Top Gear: Series 19: Africa Special	2013	NaN	6822.0

3954 rows × 5 columns

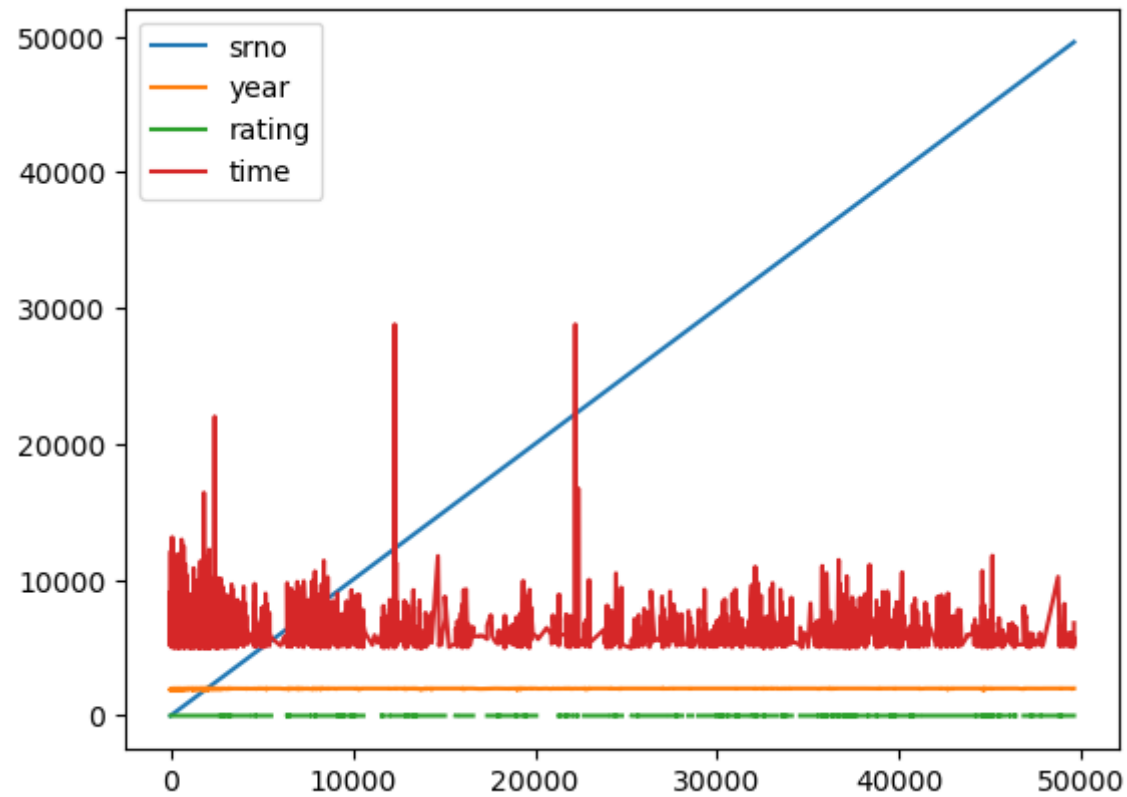
```
In [19]: data.plot()
```

```
Out[19]: <Axes: xlabel='year'>
```



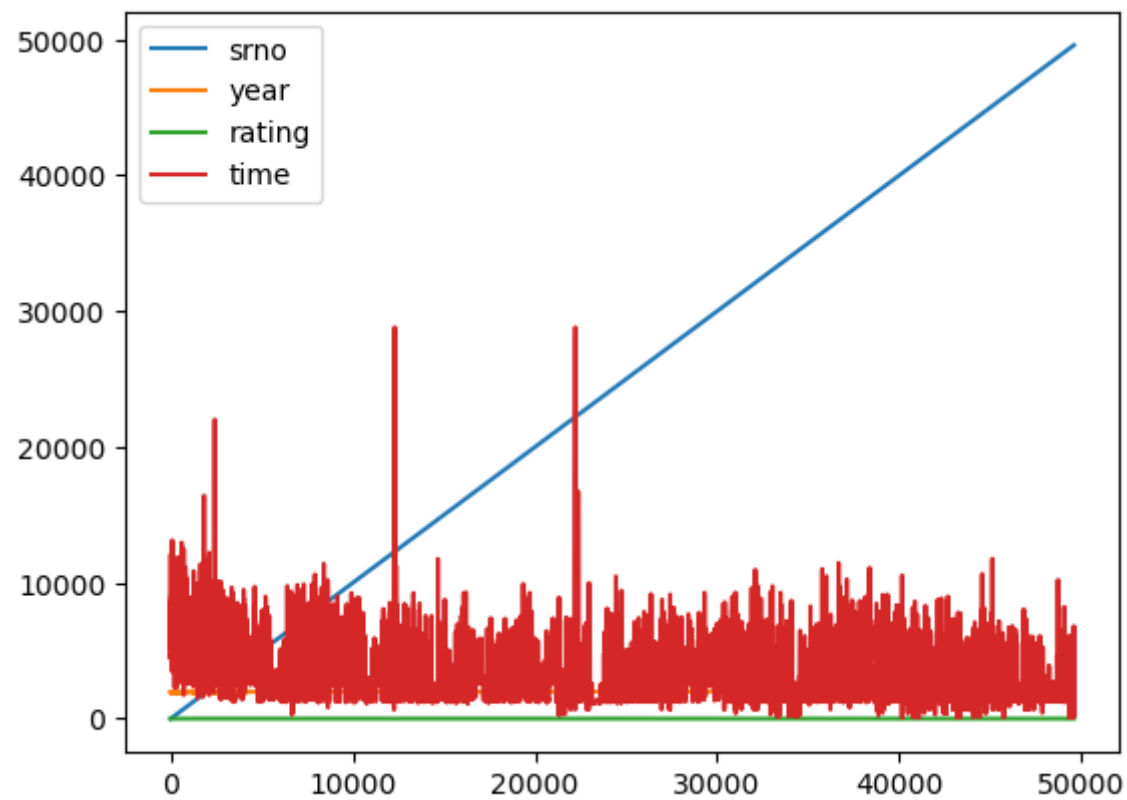
```
In [20]: data1.plot()
```

```
Out[20]: <Axes: >
```



```
In [21]: data3.plot()
```

```
Out[21]: <Axes: >
```



```
In [ ]:
```

