

```
In [22]: import pandas as pd
import numpy as np
import matplotlib as plt

In [5]: x = pd.read_csv('C:\\Users\\andre\\Desktop\\data\\d_movies.csv')
x
```

Out[5]:

	movie_title	release_date	genre	mpaa_rating	total_gross	inflation_adjusted_gross
0	Snow White and the Seven Dwarfs	1937-12-21	Musical	G	184,925,485.00	5,228,953,251.00
1	Pinocchio	1940-02-09	Adventure	G	84,300,000.00	2,188,229,052.00
2	Fantasia	1940-11-13	Musical	G	83,320,000.00	2,187,090,808.00
3	Song of the South	1946-11-12	Adventure	G	65,000,000.00	1,078,510,579.00
4	Cinderella	1950-02-15	Drama	G	85,000,000.00	920,608,730.00
...	...	...	...	...	...	...
574	The Light Between Oceans	2016-09-02	Drama	PG-13	12,545,979.00	12,545,979.00
575	Queen of Katwe	2016-09-23	Drama	PG	8,874,389.00	8,874,389.00
576	Doctor Strange	2016-11-04	Adventure	PG-13	232,532,923.00	232,532,923.00
577	Moana	2016-11-23	Adventure	PG	246,082,029.00	246,082,029.00
578	Rogue One: A Star Wars Story	2016-12-16	Adventure	PG-13	529,483,936.00	529,483,936.00

579 rows x 6 columns

```
In [6]: x.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 579 entries, 0 to 578
Data columns (total 6 columns):
#   Column              Non-Null Count  Dtype
---  ---
0   movie_title         579 non-null   object
1   release_date        579 non-null   object
2   genre               562 non-null   object
3   mpaa_rating         579 non-null   object
4   total_gross         579 non-null   object
5   inflation_adjusted_gross 579 non-null   object
dtypes: object(6)
memory usage: 27.3+ KB
```

```
In [7]: x.isnull().sum()
```

Out[7]:

movie_title	0
release_date	0
genre	17
mpaa_rating	0
total_gross	0
inflation_adjusted_gross	0

dtype: int64

```
In [8]: df=x.fillna(method='backfill')
```

```
In [9]: df.isnull().sum()
```

Out[9]:

movie_title	0
release_date	0
genre	0
mpaa_rating	0
total_gross	0
inflation_adjusted_gross	0

dtype: int64

```
In [10]: df
```

Out[10]:

	movie_title	release_date	genre	mpaa_rating	total_gross	inflation_adjusted_gross
0	Snow White and the Seven Dwarfs	1937-12-21	Musical	G	184,925,485.00	5,228,953,251.00
1	Pinocchio	1940-02-09	Adventure	G	84,300,000.00	2,188,229,052.00
2	Fantasia	1940-11-13	Musical	G	83,320,000.00	2,187,090,808.00
3	Song of the South	1946-11-12	Adventure	G	65,000,000.00	1,078,510,579.00
4	Cinderella	1950-02-15	Drama	G	85,000,000.00	920,608,730.00
...	...	...	...	...	...	...
574	The Light Between Oceans	2016-09-02	Drama	PG-13	12,545,979.00	12,545,979.00
575	Queen of Katwe	2016-09-23	Drama	PG	8,874,389.00	8,874,389.00
576	Doctor Strange	2016-11-04	Adventure	PG-13	232,532,923.00	232,532,923.00
577	Moana	2016-11-23	Adventure	PG	246,082,029.00	246,082,029.00
578	Rogue One: A Star Wars Story	2016-12-16	Adventure	PG-13	529,483,936.00	529,483,936.00

579 rows x 6 columns

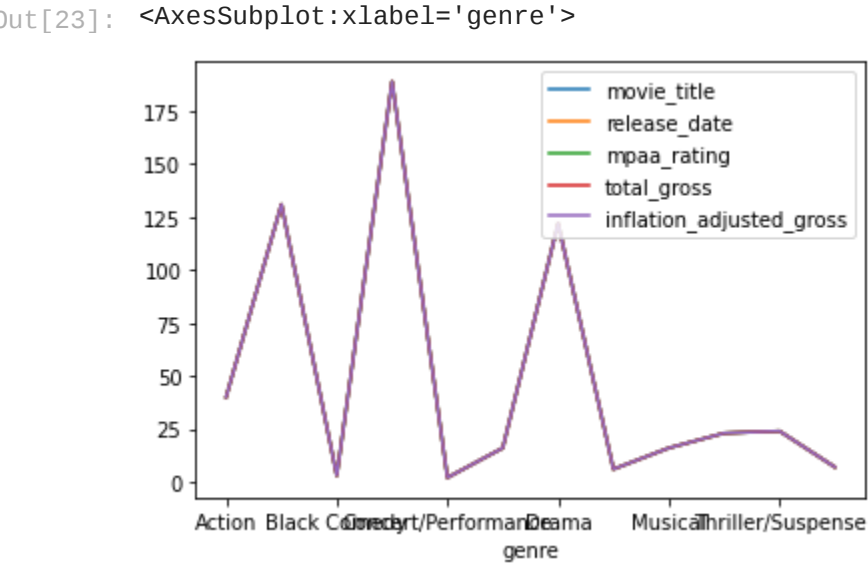
```
In [16]: df1 = df.groupby('genre')
```

```
In [24]: df1.count()
```

Out[24]:

	movie_title	release_date	mpaa_rating	total_gross	inflation_adjusted_gross
genre					
Action	40	40	40	40	40
Adventure	131	131	131	131	131
Black Comedy	3	3	3	3	3
Comedy	189	189	189	189	189
Concert/Performance	2	2	2	2	2
Documentary	16	16	16	16	16
Drama	122	122	122	122	122
Horror	6	6	6	6	6
Musical	16	16	16	16	16
Romantic Comedy	23	23	23	23	23
Thriller/Suspense	24	24	24	24	24
Western	7	7	7	7	7

```
In [23]: df1.count().plot()
```



```
In [33]: df3['d - a'] = df1.apply(lambda x: df1['total_gross'] -df1['inflation_adjusted_gross'])
```

```
-----
TypeError                                 Traceback (most recent call last)
c:\users\andre\appdata\local\programs\python\python39\lib\site-packages\pandas\core\groupby\groupby.py in apply(self, func, *args, **kwargs)
    893         try:
--> 894             result = self._python_apply_general(f, self._selected_obj)
    895         except TypeError:

c:\users\andre\appdata\local\programs\python\python39\lib\site-packages\pandas\core\groupby\groupby.py in _python_apply_general(self, f, data)
    927         """
--> 928         keys, values, mutated = self.grouper.apply(f, data, self.axis)
    929

c:\users\andre\appdata\local\programs\python\python39\lib\site-packages\pandas\core\groupby\ops.py in apply(self, f, data, axis)
    237         group_axes = group.axes
--> 238         res = f(group)
    239         if not _is_indexed_like(res, group_axes, axis):

<ipython-input-33-ec3ec4b4adf5> in <lambda>(x)
----> 1 df3['d - a'] = df1.apply(lambda x: df1['total_gross'] -df1['inflation_adjusted_gross'])

TypeError: unsupported operand type(s) for -: 'SeriesGroupBy' and 'SeriesGroupBy'

During handling of the above exception, another exception occurred:

TypeError                                 Traceback (most recent call last)
<ipython-input-33-ec3ec4b4adf5> in <module>
----> 1 df3['d - a'] = df1.apply(lambda x: df1['total_gross'] -df1['inflation_adjusted_gross'])

c:\users\andre\appdata\local\programs\python\python39\lib\site-packages\pandas\core\groupby\groupby.py in apply(self, func, *args, **kwargs)
    903
    904         with group_selection_context(self):
--> 905             return self._python_apply_general(f, self._selected_obj)
    906
    907         return result

c:\users\andre\appdata\local\programs\python\python39\lib\site-packages\pandas\core\groupby\groupby.py in _python_apply_general(self, f, data)
    926         data after applying f
    927         """
--> 928         keys, values, mutated = self.grouper.apply(f, data, self.axis)
    929
    930         return self._wrap_applied_output(

c:\users\andre\appdata\local\programs\python\python39\lib\site-packages\pandas\core\groupby\ops.py in apply(self, f, data, axis)
    236         # group might be modified
    237         group_axes = group.axes
--> 238         res = f(group)
    239         if not _is_indexed_like(res, group_axes, axis):
    240             mutated = True

<ipython-input-33-ec3ec4b4adf5> in <lambda>(x)
----> 1 df3['d - a'] = df1.apply(lambda x: df1['total_gross'] -df1['inflation_adjusted_gross'])

TypeError: unsupported operand type(s) for -: 'SeriesGroupBy' and 'SeriesGroupBy'
```

```
In [ ]:
```