CS7593 DATA STRUCTURES WITH PYTHON
ASSIGNMENT-3

REG NO: 2017504062
NAME : NIKKILESHH M

TITLE:

IMDb rating Data Analysis

DATASET NAME AND SOURCE:

IMDb ratings from Github
Source- **http://bit.ly/imdbratings**

ABOUT THE DATASET:

- IMDb (also known as the Internet Movie Database) is an online database of information related to films, television programs, home videos, video games, and streaming content online – including cast, production crew and personal biographies, plot summaries, trivia, ratings, and fan and critical reviews

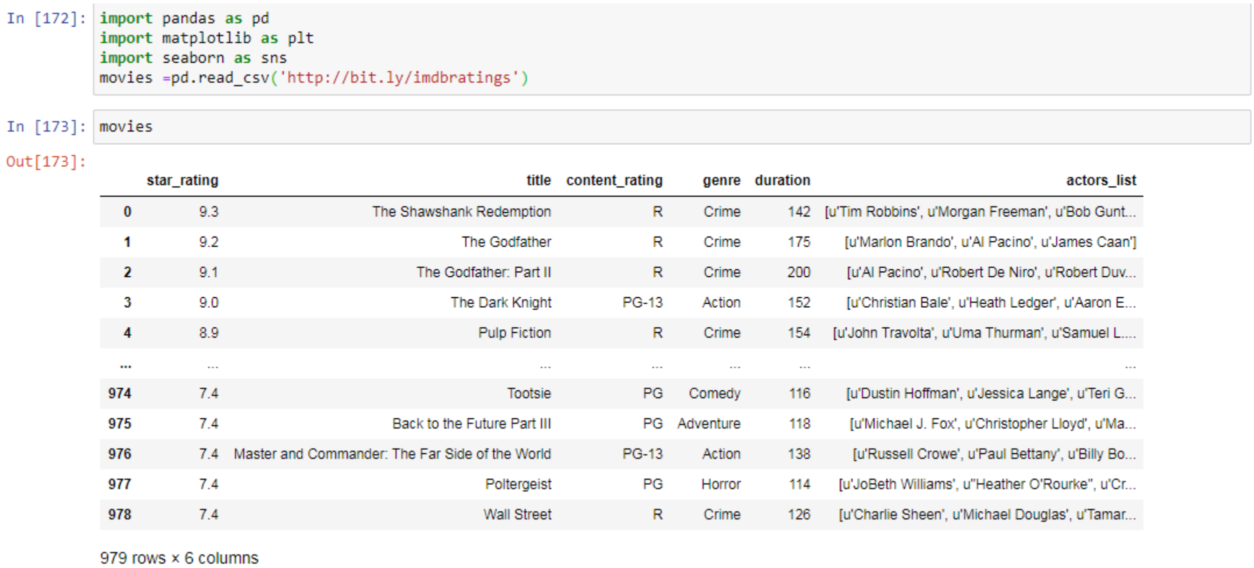
- IMDb registered users can cast a vote (from 1 to 10) on every released title in the database. Individual votes are then aggregated and summarized as a single IMDb rating

LIST OF QUERIES THAT ARE TO BE ANSWERED USING DATA ANALYSIS:

1) What are the movies in crime genre that are considered to be an all-time hit with respect to star rating (greater than 9)?
2) What are the mean, variance and standard deviation of duration column?
3) Find the mean of duration and star rating against each genre of movie.
4) Find all instances when:
   a. Rating is greater than 8.5 and content rating is equal to 'PG'
   (Or)
   b. Duration is greater than 240

5) Find all instances when:
   a. Content rating is “APPROVED” and genre is “Action”
   (Or)
   b. Rating is greater than 9
6) Examine the data set with the following conditions :
   i) Only with columns title, genre and duration
   ii) Star rating less than 9
7) Compare the IMDb star rating for films in Western and Film-Noir genres. What inferences can be drawn from it?
8) Compare the IMDb star rating for films in Sci-Fi and Thriller genres. What inferences can be drawn from it?
9) Compare the star rating and duration of the movies present in the dataset. Mention the similarities noted.
10) Graphically analyze the number of movies in each star rating. Also find the rating which occurs maximum in the dataset.
11) Graphically plot the relation between star rating and duration with respect to content rating
12) Graphically analyze the linear relation between the columns star rating and duration. What type of linear relation is observed?

DATA ANALYSIS:



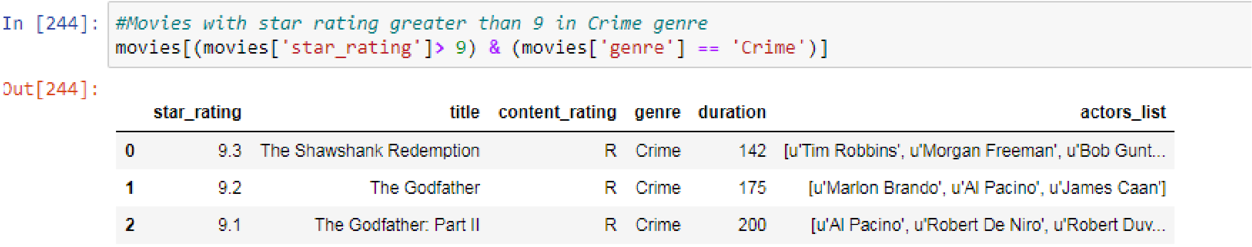
```
In [172]: import pandas as pd
          import matplotlib as plt
          import seaborn as sns
          movies =pd.read_csv('http://bit.ly/imdbratings')

In [173]: movies
Out[173]:
```

	star_rating	title	content_rating	genre	duration	actors_list
0	9.3	The Shawshank Redemption	R	Crime	142	[u'Tim Robbins', u'Morgan Freeman', u'Bob Gunt...
1	9.2	The Godfather	R	Crime	175	[u'Marlon Brando', u'Al Pacino', u'James Caan']
2	9.1	The Godfather: Part II	R	Crime	200	[u'Al Pacino', u'Robert De Niro', u'Robert Duv...
3	9.0	The Dark Knight	PG-13	Action	152	[u'Christian Bale', u'Heath Ledger', u'Aaron E...
4	8.9	Pulp Fiction	R	Crime	154	[u'John Travolta', u'Uma Thurman', u'Samuel L....
...	...	...	...	...	...	...
974	7.4	Tootsie	PG	Comedy	116	[u'Dustin Hoffman', u'Jessica Lange', u'Teri G...
975	7.4	Back to the Future Part III	PG	Adventure	118	[u'Michael J. Fox', u'Christopher Lloyd', u'Ma...
976	7.4	Master and Commander: The Far Side of the World	PG-13	Action	138	[u'Russell Crowe', u'Paul Bettany', u'Billy Bo...
977	7.4	Poltergeist	PG	Horror	114	[u'JoBeth Williams', u"Heather O'Rourke", u'Cr...
978	7.4	Wall Street	R	Crime	126	[u'Charlie Sheen', u'Michael Douglas', u'Tamar...

979 rows × 6 columns

QUERIES AND OUTPUT:

1) What are the movies in crime genre that are considered to be an all-time hit with respect to star rating (Greater than 9)?



```
In [244]: #Movies with star rating greater than 9 in Crime genre
          movies[(movies['star_rating']> 9) & (movies['genre'] == 'Crime')]
```

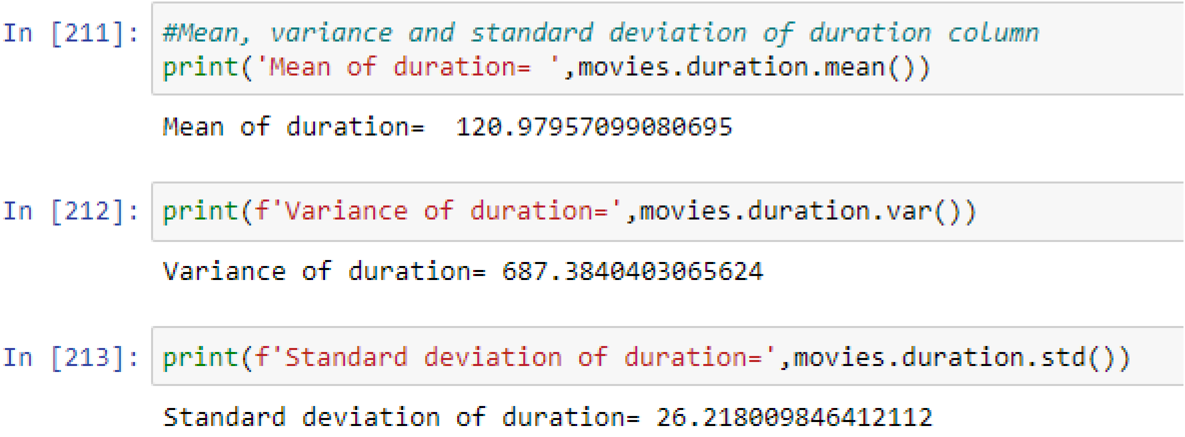
Out[244]:

	star_rating	title	content_rating	genre	duration	actors_list
0	9.3	The Shawshank Redemption	R	Crime	142	[u'Tim Robbins', u'Morgan Freeman', u'Bob Gunt...
1	9.2	The Godfather	R	Crime	175	[u'Marlon Brando', u'Al Pacino', u'James Caan']
2	9.1	The Godfather: Part II	R	Crime	200	[u'Al Pacino', u'Robert De Niro', u'Robert Duv...

Inference:

- No crime movie has ever beaten the record of “The Shawshank Redemption” with an IMDb rating of 9.3

2) What are the mean, variance and standard deviation of the ‘duration’ column?



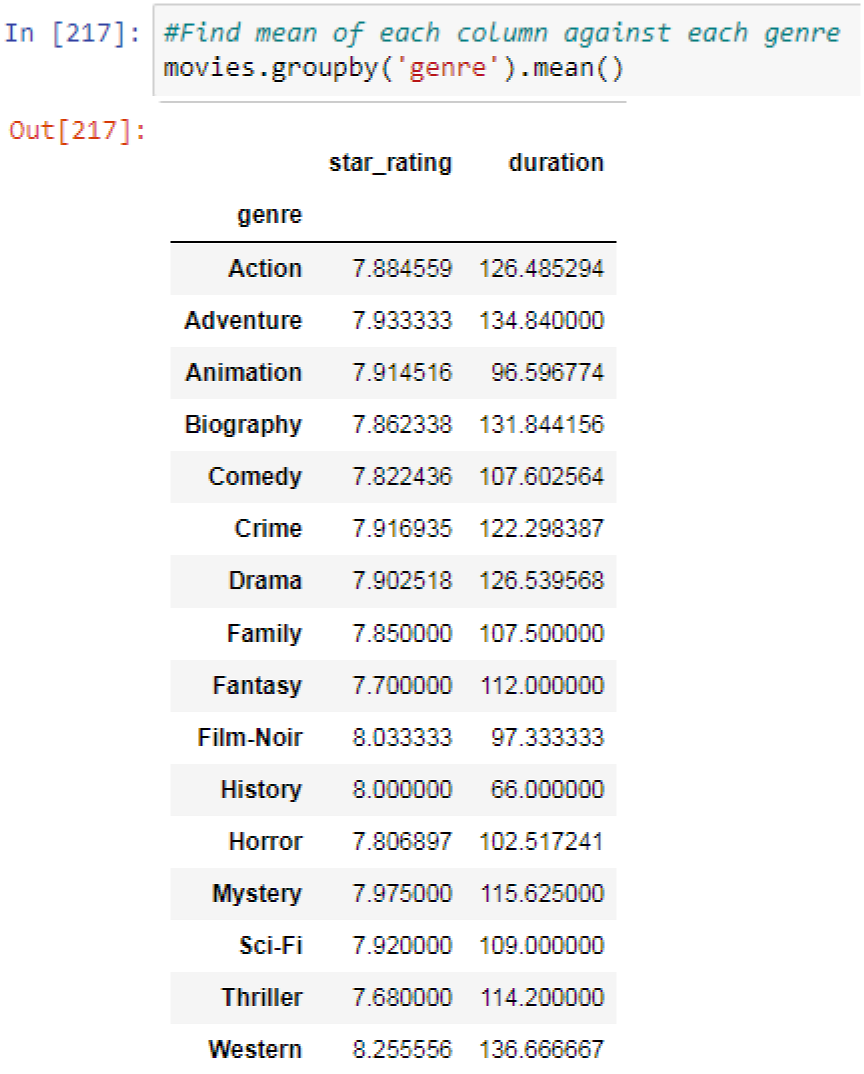
```
In [211]: #Mean, variance and standard deviation of duration column
          print('Mean of duration= ',movies.duration.mean())

          Mean of duration=  120.97957099080695

In [212]: print(f'Variance of duration=',movies.duration.var())

          Variance of duration= 687.3840403065624

In [213]: print(f'Standard deviation of duration=',movies.duration.std())

          Standard deviation of duration= 26.218009846412112
```

Inference:

- Mean, variance and standard deviation of duration column:

MEAN	VARIANCE	STANDARD DEVIATION
120.98	687.38	26.21

3) Find the mean of duration and star rating against each genre of movie. What inferences can be drawn from the output?



```
In [217]: #Find mean of each column against each genre
          movies.groupby('genre').mean()
```

```
Out[217]:
                star_rating    duration
          genre
         Action    7.884559  126.485294
      Adventure    7.933333  134.840000
      Animation    7.914516   96.596774
      Biography    7.862338  131.844156
         Comedy    7.822436  107.602564
          Crime    7.916935  122.298387
          Drama    7.902518  126.539568
         Family    7.850000  107.500000
        Fantasy    7.700000  112.000000
      Film-Noir    8.033333   97.333333
        History    8.000000   66.000000
         Horror    7.806897  102.517241
        Mystery    7.975000  115.625000
         Sci-Fi    7.920000  109.000000
       Thriller    7.680000  114.200000
        Western    8.255556  136.666667
```

Inferences:

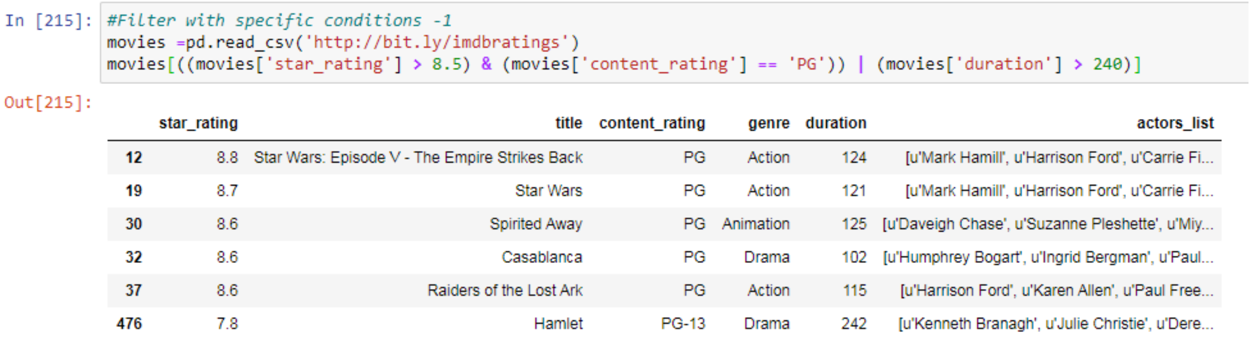
- The Western genre films have the highest average duration and average star rating compared any other genre
- The lowest average star rating is for “Action” genre
- The lowest average duration is for “History” genre

4) Find all instances when:

a. Rating is greater than 8.5 and content rating is equal to 'PG'

(Or)

b. Duration is greater than 240



```
In [215]: #Filter with specific conditions -1
          movies =pd.read_csv('http://bit.ly/imdbratings')
          movies[((movies['star_rating'] > 8.5) & (movies['content_rating'] == 'PG')) | (movies['duration'] > 240)]
```

Out[215]:

	star_rating	title	content_rating	genre	duration	actors_list
12	8.8	Star Wars: Episode V - The Empire Strikes Back	PG	Action	124	[u'Mark Hamill', u'Harrison Ford', u'Carrie Fi...
19	8.7	Star Wars	PG	Action	121	[u'Mark Hamill', u'Harrison Ford', u'Carrie Fi...
30	8.6	Spirited Away	PG	Animation	125	[u'Daveigh Chase', u'Suzanne Pleshette', u'Miy...
32	8.6	Casablanca	PG	Drama	102	[u'Humphrey Bogart', u'Ingrid Bergman', u'Paul...
37	8.6	Raiders of the Lost Ark	PG	Action	115	[u'Harrison Ford', u'Karen Allen', u'Paul Free...
476	7.8	Hamlet	PG-13	Drama	242	[u'Kenneth Branagh', u'Julie Christie', u'Dere...

Inferences:

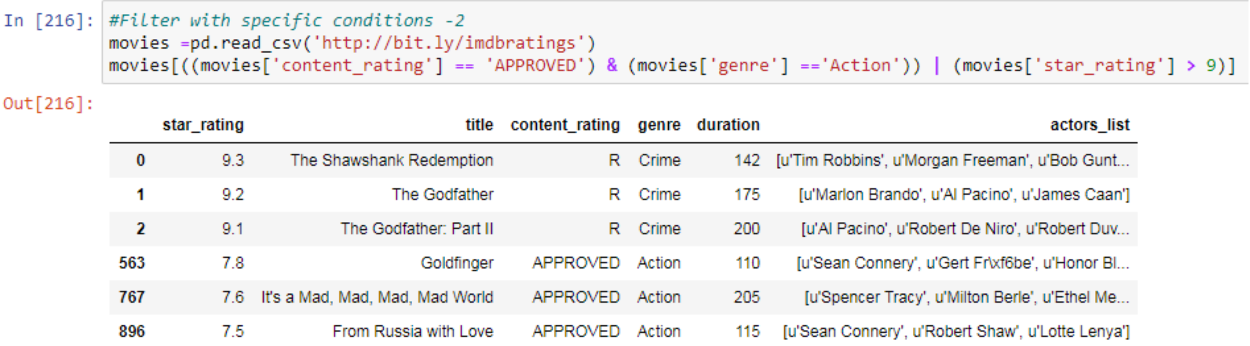
- The movie “Star Wars-Episode V- The Empire Strikes Back” has the highest star rating of 8.8 in action genre with content rating ‘PG’
- There is only one movie “Hamlet” which has a duration greater than 240

5) Find all instances when:

a. Content rating is “APPROVED” and genre is “Action”

(Or)

b. Star rating is greater than 9



```
In [216]: #Filter with specific conditions -2
          movies =pd.read_csv('http://bit.ly/imdbratings')
          movies[((movies['content_rating'] == 'APPROVED') & (movies['genre'] =='Action')) | (movies['star_rating'] > 9)]
```

Out[216]:

	star_rating	title	content_rating	genre	duration	actors_list
0	9.3	The Shawshank Redemption	R	Crime	142	[u'Tim Robbins', u'Morgan Freeman', u'Bob Gunt...
1	9.2	The Godfather	R	Crime	175	[u'Marlon Brando', u'Al Pacino', u'James Caan']
2	9.1	The Godfather: Part II	R	Crime	200	[u'Al Pacino', u'Robert De Niro', u'Robert Duv...
563	7.8	Goldfinger	APPROVED	Action	110	[u'Sean Connery', u'Gert Fr\xf6be', u'Honor Bl...
767	7.6	It's a Mad, Mad, Mad, Mad World	APPROVED	Action	205	[u'Spencer Tracy', u'Milton Berle', u'Ethel Me...
896	7.5	From Russia with Love	APPROVED	Action	115	[u'Sean Connery', u'Robert Shaw', u'Lotte Lenya']

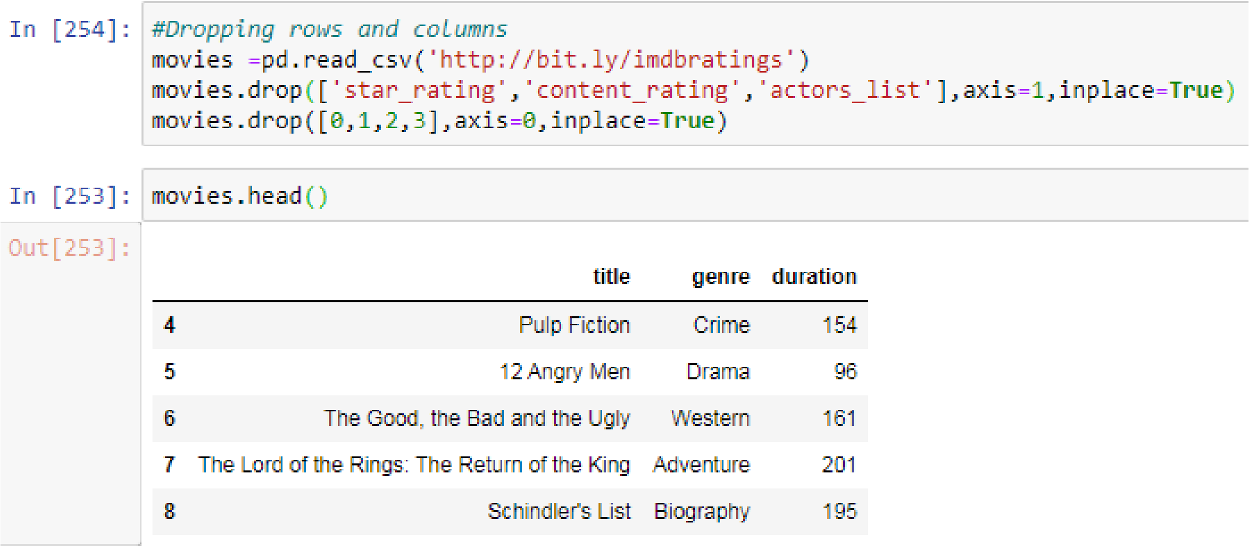
Inferences:

- The movie “Goldfinger” has the highest rating in action genre with content rating APPROVED
- The movie “Godfather: Part II” has the highest duration among movies in crime genre

6) Examine the data set with the following conditions :

   i) Only with columns title, genre and duration

   ii) Star rating less than 9



```
In [254]: #Dropping rows and columns
          movies =pd.read_csv('http://bit.ly/imdbratings')
          movies.drop(['star_rating','content_rating','actors_list'],axis=1,inplace=True)
          movies.drop([0,1,2,3],axis=0,inplace=True)
```

```
In [253]: movies.head()
```

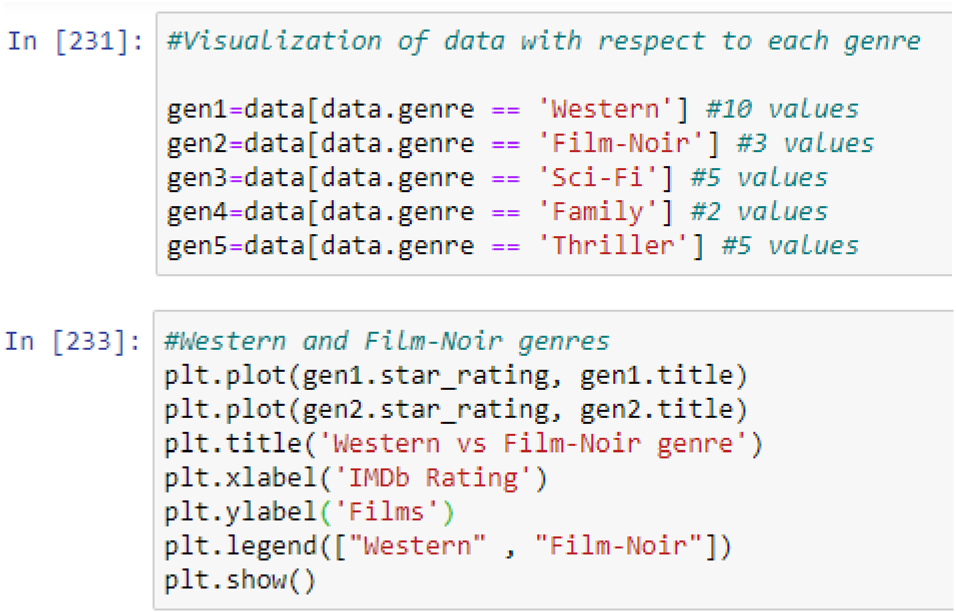
Out[253]:

	title	genre	duration
4	Pulp Fiction	Crime	154
5	12 Angry Men	Drama	96
6	The Good, the Bad and the Ugly	Western	161
7	The Lord of the Rings: The Return of the King	Adventure	201
8	Schindler's List	Biography	195

Inferences:

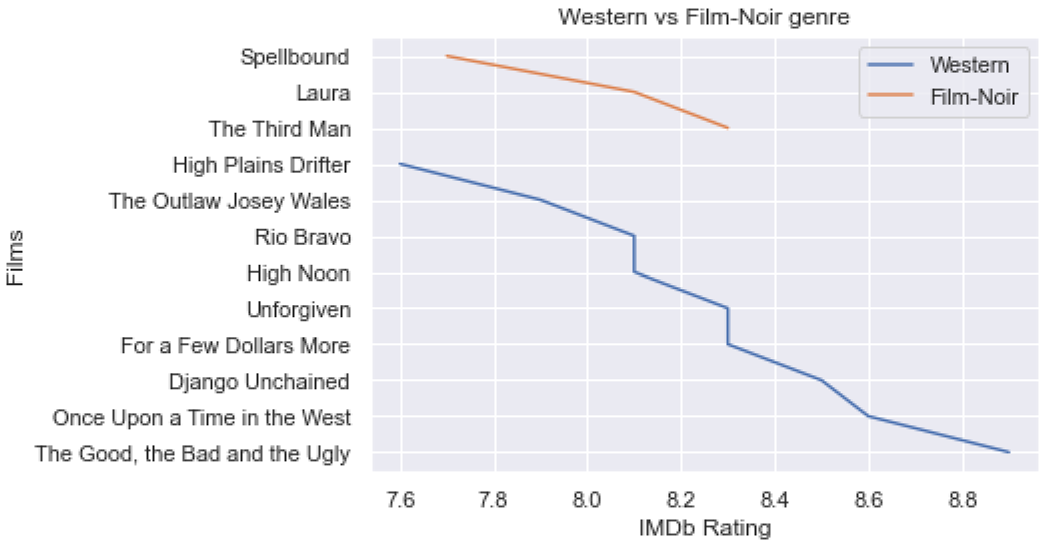
- A new dataset is created for IMDb ratings with only 3 columns and star ratings less than 9
- This dataset be used for a more simpler data analysis
- The movie “Pulp Fiction” tops the new dataset with a duration on 154

7) Compare the IMDb star rating for films in Western and Film-Noir genres. What inferences can be drawn from it?



```
In [231]: #Visualization of data with respect to each genre

          gen1=data[data.genre == 'Western'] #10 values
          gen2=data[data.genre == 'Film-Noir'] #3 values
          gen3=data[data.genre == 'Sci-Fi'] #5 values
          gen4=data[data.genre == 'Family'] #2 values
          gen5=data[data.genre == 'Thriller'] #5 values
```

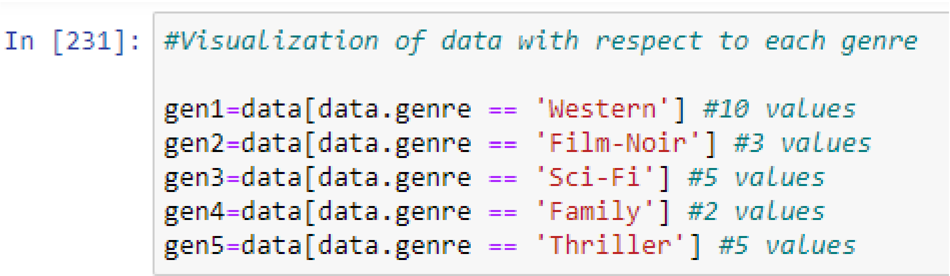
```
In [233]: #Western and Film-Noir genres
          plt.plot(gen1.star_rating, gen1.title)
          plt.plot(gen2.star_rating, gen2.title)
          plt.title('Western vs Film-Noir genre')
          plt.xlabel('IMDb Rating')
          plt.ylabel('Films')
          plt.legend(["Western" , "Film-Noir"])
          plt.show()
```



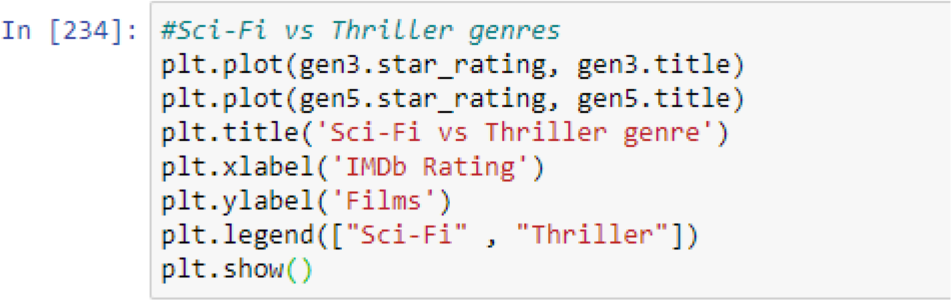
Inferences:

- The range of IMDb ratings for Western genre is higher than Film-Noir genre with a maximum rating of 8.9
- The maximum rating for the genre Film-Noir is 8.3, for the movie “The Third Man”
- The lowest rating for Western genre is 7.6 for the movie “High Plains Drifter”

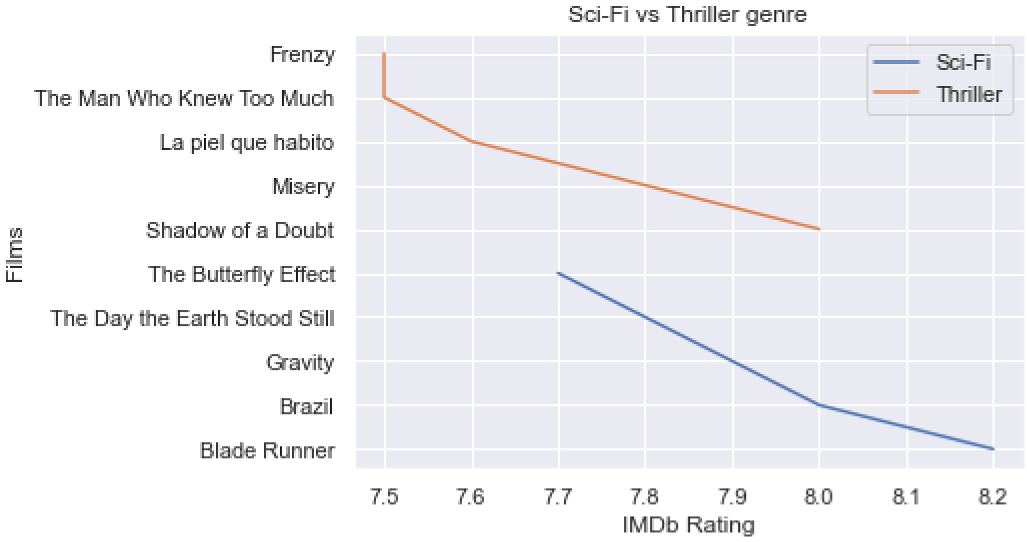
8) Compare the IMDb star ratings for films in Sci-Fi and Thriller genres. What inferences can be drawn from it?



```
In [231]: #Visualization of data with respect to each genre

          gen1=data[data.genre == 'Western'] #10 values
          gen2=data[data.genre == 'Film-Noir'] #3 values
          gen3=data[data.genre == 'Sci-Fi'] #5 values
          gen4=data[data.genre == 'Family'] #2 values
          gen5=data[data.genre == 'Thriller'] #5 values
```



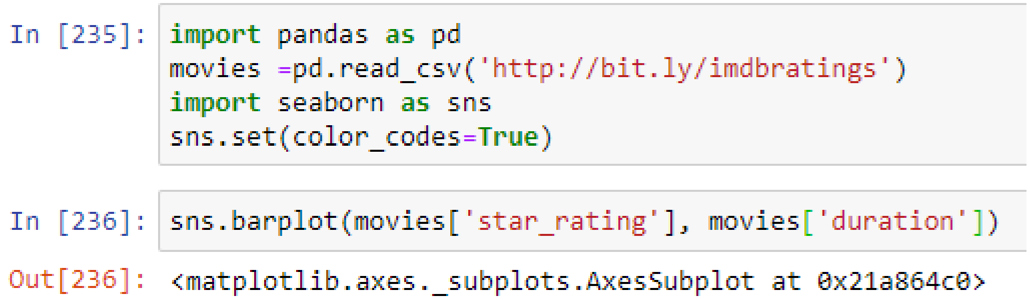
```
In [234]: #Sci-Fi vs Thriller genres
          plt.plot(gen3.star_rating, gen3.title)
          plt.plot(gen5.star_rating, gen5.title)
          plt.title('Sci-Fi vs Thriller genre')
          plt.xlabel('IMDb Rating')
          plt.ylabel('Films')
          plt.legend(["Sci-Fi" , "Thriller"])
          plt.show()
```



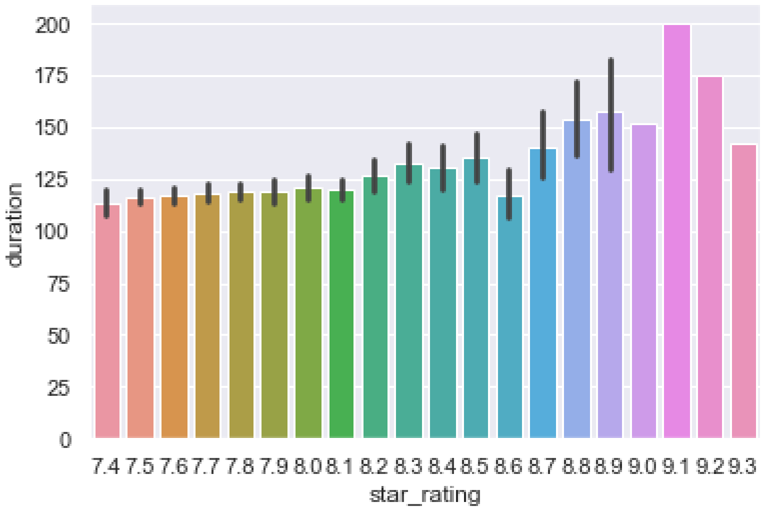
Inferences:

- The movies “Frenzy” and “The Man Who Knew Too Much” have the same rating of 7.5 in Thriller genre
- The maximum rating of Sci-Fi genre is 8.2 for the movie “Blade Runner”
- The maximum rating for “Thriller” movies is 8.0

9) Compare the star rating and duration of the movies present in the dataset. Mention the similarities noted.



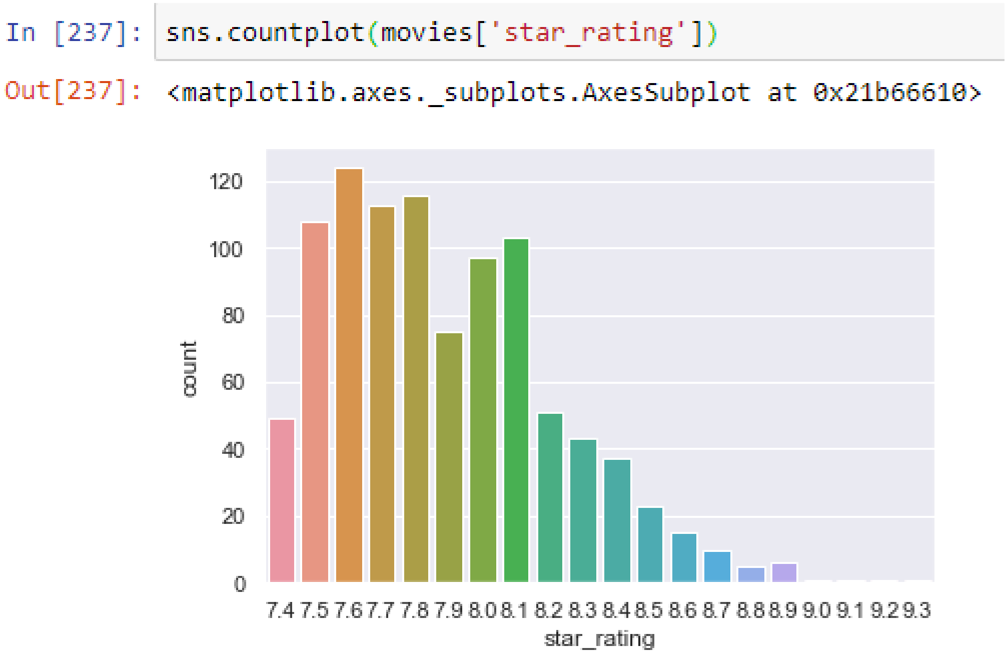
```
In [235]: import pandas as pd
          movies =pd.read_csv('http://bit.ly/imdbratings')
          import seaborn as sns
          sns.set(color_codes=True)

In [236]: sns.barplot(movies['star_rating'], movies['duration'])

Out[236]: <matplotlib.axes._subplots.AxesSubplot at 0x21a864c0>
```



Inferences:

- The duration is almost same for the movies whose star rating range from 7.4 to 8.1
- The highest duration is found for the star rating 9.1
- The lowest duration is found for the star ratings 7.4 and 8.6

10) Graphically analyze the number of movies in each star rating. Also find the rating which occurs maximum in the dataset.

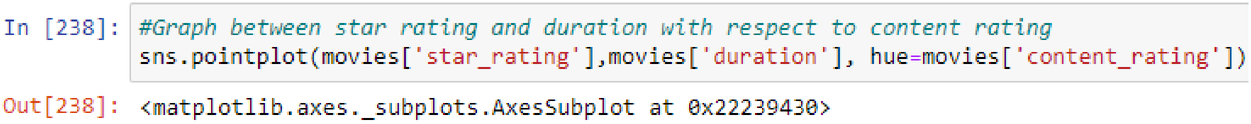


```
In [237]: sns.countplot(movies['star_rating'])
Out[237]: <matplotlib.axes._subplots.AxesSubplot at 0x21b66610>
```

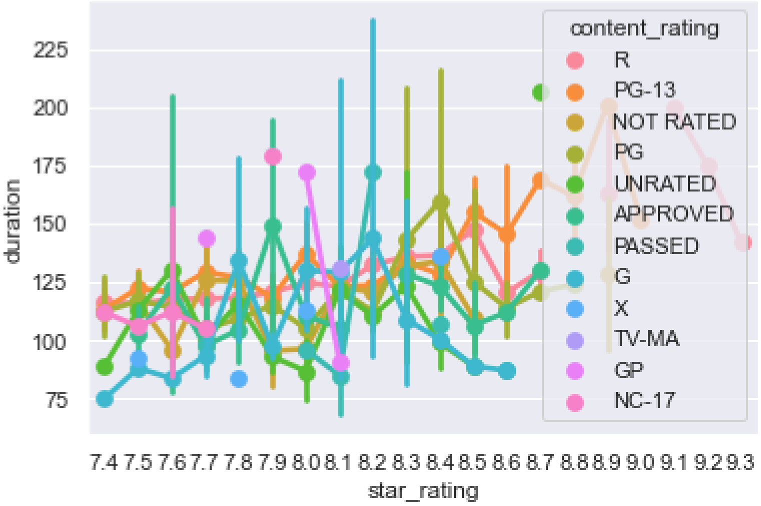
Inferences:

- The rating which occurs maximum in the data set is 7.6, which implies that the number of movies with rating 7.6 are the highest in the dataset
- The ratings which occur minimum in the dataset are in the range of 9 to 9.3

11) Graphically plot the relation between star rating and duration with respect to content rating



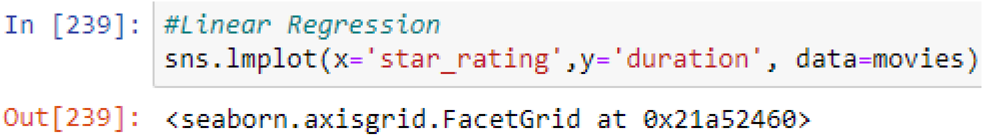
```
In [238]: #Graph between star rating and duration with respect to content rating
          sns.pointplot(movies['star_rating'],movies['duration'], hue=movies['content_rating'])

Out[238]: <matplotlib.axes._subplots.AxesSubplot at 0x22239430>
```



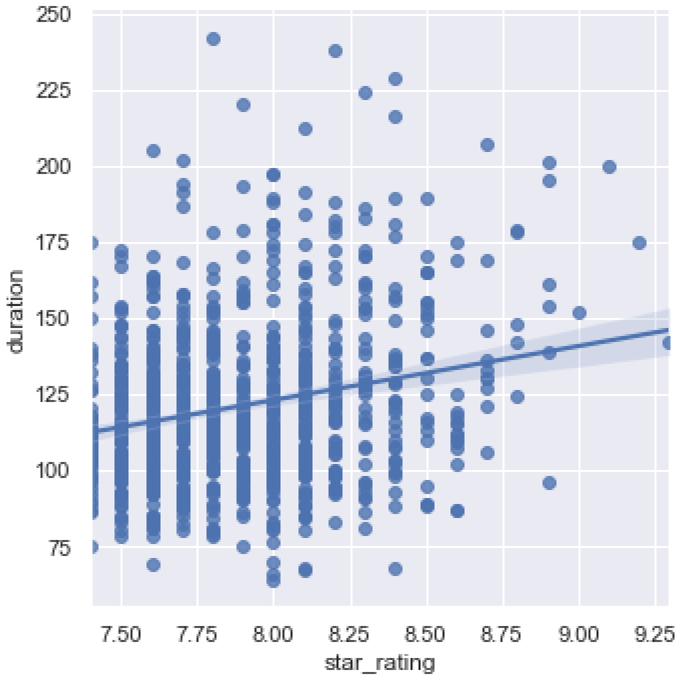
Inferences:

- The content rating ‘G’ has the highest duration which is greater than 225
- The minimum duration for movies with content rating ‘X’ is 75
- The maximum duration for movies with content rating ‘NC-17’ is 179

12)Graphically analyze the linear relation between the columns star rating and duration. What type of linear relation is observed?



```
In [239]: #Linear Regression
          sns.lmplot(x='star_rating',y='duration', data=movies)

Out[239]: <seaborn.axisgrid.FacetGrid at 0x21a52460>
```



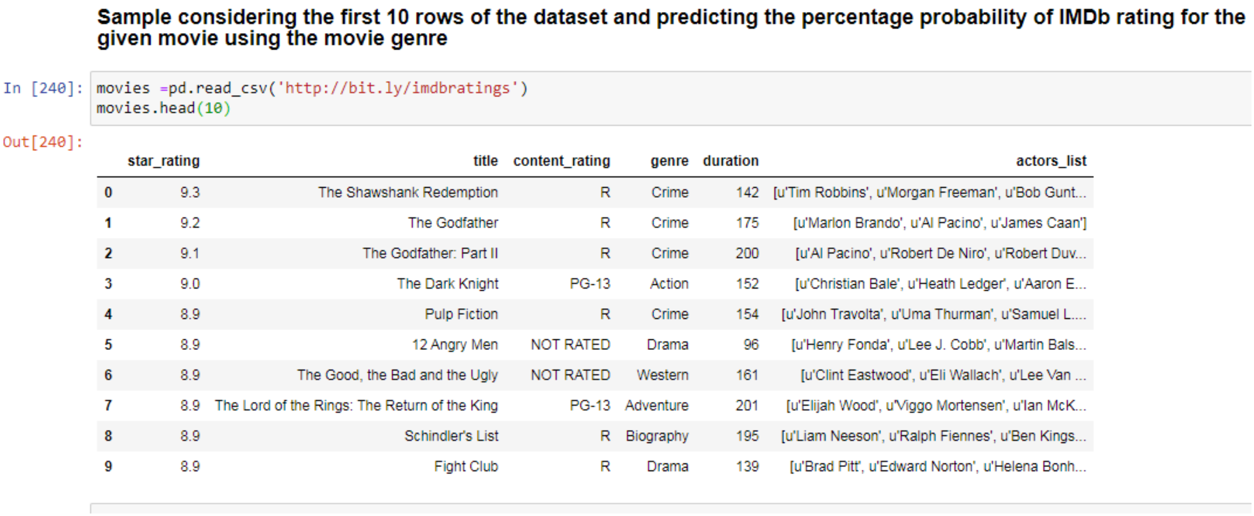
Inferences:

- A positive linear relation is observed between star rating and duration as the slope of linear regression line is positive
- We can conclude that movies with duration between 110 to 150 tend to have a higher rating compared to others

FUTURE PREDICTIONS:

- By studying the patterns in movie genres, we can predict whether a new movie when released will get a high rating in IMDb or not

- For example, considering the first 10 rows of the data set and counting the number of genres of each type, we can predict if a new movie of a given genre will get a high rating in IMDb



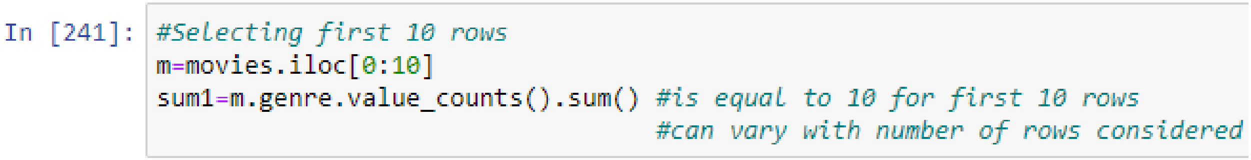
Sample considering the first 10 rows of the dataset and predicting the percentage probability of IMDb rating for the given movie using the movie genre

```
In [240]: movies =pd.read_csv('http://bit.ly/imdbratings')
          movies.head(10)
```

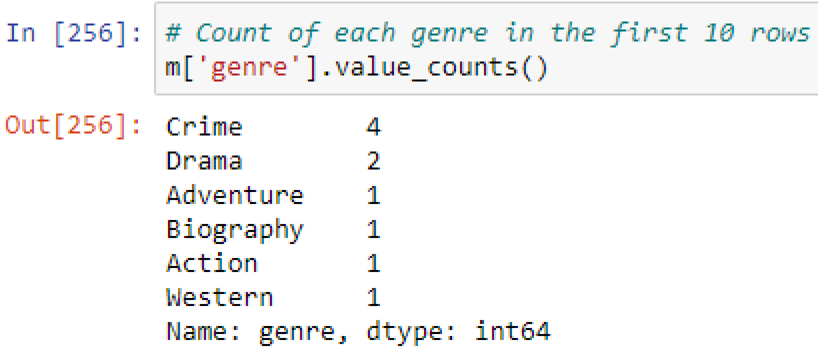
Out[240]:

	star_rating	title	content_rating	genre	duration	actors_list
0	9.3	The Shawshank Redemption	R	Crime	142	[u'Tim Robbins', u'Morgan Freeman', u'Bob Gunt...
1	9.2	The Godfather	R	Crime	175	[u'Marlon Brando', u'Al Pacino', u'James Caan']
2	9.1	The Godfather: Part II	R	Crime	200	[u'Al Pacino', u'Robert De Niro', u'Robert Duv...
3	9.0	The Dark Knight	PG-13	Action	152	[u'Christian Bale', u'Heath Ledger', u'Aaron E...
4	8.9	Pulp Fiction	R	Crime	154	[u'John Travolta', u'Uma Thurman', u'Samuel L....
5	8.9	12 Angry Men	NOT RATED	Drama	96	[u'Henry Fonda', u'Lee J. Cobb', u'Martin Bals...
6	8.9	The Good, the Bad and the Ugly	NOT RATED	Western	161	[u'Clint Eastwood', u'Eli Wallach', u'Lee Van ...
7	8.9	The Lord of the Rings: The Return of the King	PG-13	Adventure	201	[u'Elijah Wood', u'Viggo Mortensen', u'Ian McK...
8	8.9	Schindler's List	R	Biography	195	[u'Liam Neeson', u'Ralph Fiennes', u'Ben Kings...
9	8.9	Fight Club	R	Drama	139	[u'Brad Pitt', u'Edward Norton', u'Helena Bonh...

a) Considering the first 10 rows of the data set and calculating the sum of count of each genre

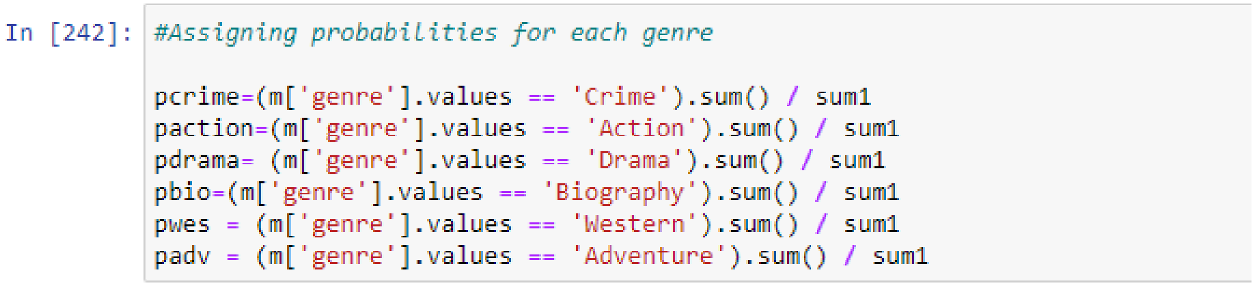


```
In [241]: #Selecting first 10 rows
          m=movies.iloc[0:10]
          sum1=m.genre.value_counts().sum() #is equal to 10 for first 10 rows
                                            #can vary with number of rows considered
```



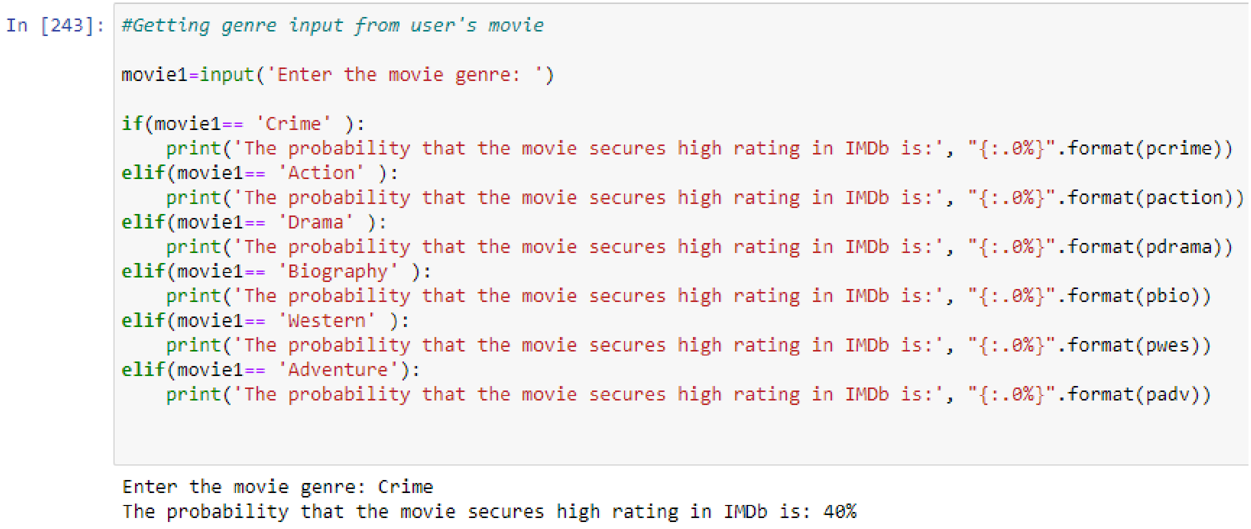
```
In [256]: # Count of each genre in the first 10 rows
          m['genre'].value_counts()

Out[256]: Crime        4
          Drama        2
          Adventure    1
          Biography    1
          Action       1
          Western      1
          Name: genre, dtype: int64
```

b) Calculating the probabilities for each genre- dividing the count of each genre by the sum calculated in step ‘a’



```
In [242]: #Assigning probabilities for each genre

          pcrime=(m['genre'].values == 'Crime').sum() / sum1
          paction=(m['genre'].values == 'Action').sum() / sum1
          pdrama= (m['genre'].values == 'Drama').sum() / sum1
          pbio=(m['genre'].values == 'Biography').sum() / sum1
          pwes = (m['genre'].values == 'Western').sum() / sum1
          padv = (m['genre'].values == 'Adventure').sum() / sum1
```

c) Getting input from the user and based on the genre provided, a prediction is made on the movie to secure high rating in IMDb



```
In [243]: #Getting genre input from user's movie

          movie1=input('Enter the movie genre: ')

          if(movie1== 'Crime' ):
              print('The probability that the movie secures high rating in IMDb is:', "{:.0%}".format(pcrime))
          elif(movie1== 'Action' ):
              print('The probability that the movie secures high rating in IMDb is:', "{:.0%}".format(paction))
          elif(movie1== 'Drama' ):
              print('The probability that the movie secures high rating in IMDb is:', "{:.0%}".format(pdrama))
          elif(movie1== 'Biography' ):
              print('The probability that the movie secures high rating in IMDb is:', "{:.0%}".format(pbio))
          elif(movie1== 'Western' ):
              print('The probability that the movie secures high rating in IMDb is:', "{:.0%}".format(pwes))
          elif(movie1== 'Adventure'):
              print('The probability that the movie secures high rating in IMDb is:', "{:.0%}".format(padv))

          Enter the movie genre: Crime
          The probability that the movie secures high rating in IMDb is: 40%
```

The above prediction model considers the first 10 rows of the dataset, whereas for more accurate and in-depth prediction and analysis, we can use machine learning models to observe the parameters such as:

- Trends in movie ratings based on genre, i.e., which genre is liked by most people and based on that predict the rating of a new movie

- By using supervised learning models such as regression and KNN, we can train and test the data until the model achieves a desired level of accuracy