IMDB Movie Analysis

IMDB Movie Analysis is a process of analyzing and understanding the data of movies available on the IMDB platform. It involves using various statistical and data analysis techniques to extract useful insights and information from the data.
The IMDB Movie Analysis can be performed on various aspects of the movie data, such as the director, actors, genre, year of release, budget, revenue, ratings, and reviews. By analyzing these aspects, we can gain insights into the preferences of the audience and the factors that contribute to the success or failure of a movie.
The IMDB Movie Analysis is useful for movie studios, production houses, and distributors who want to make data-driven decisions about their movies. They can use the insights gained from the analysis to optimize their strategies for creating and marketing movies that are more likely to be successful. It is also useful for movie enthusiasts who want to explore and understand the trends in the movie industry.

I am using Google Collab and Python Language to find the answers for the questions asked and here is a report on my findings in the dataset given for the 5th project-

A: Cleaning the data -

Cleaning data with missing values in Excel can be done in several ways. Here are a few common techniques:

1. Delete Rows or Columns: One way to handle missing values is to simply delete the rows or columns that contain them. This approach may be appropriate if the missing values are a small

percentage of the total data. However, if too much data is removed, it may negatively impact your analysis. To delete rows or columns with missing values, select the appropriate rows or columns, right-click, and choose "Delete."

2. Fill Missing Values with Mean or Median: Another way to handle missing values is to fill them in with the mean or median value of the column. This approach can help preserve the overall distribution of the data. To fill in missing values with the mean or median, select the appropriate column, click on the "Home" tab, select "Find & Select," and then choose "Go To Special." In the dialog box that appears, select "Blanks" and click "OK." This will select all the blank cells in the column. Next, enter the formula "=AVERAGE(A1:A10)" or "=MEDIAN(A1:A10)" (replacing A1:A10 with the range of values in your column) and press Ctrl + Enter.
3. Interpolate Missing Values: Interpolation is a statistical technique that estimates missing values based on the values of neighboring data points. This approach can be useful if there are patterns in the missing values. Excel has several interpolation functions, including "FORECAST" and "TREND." To use these functions, you'll need to select the appropriate range of data and enter the function formula, specifying the missing values as the "x" values.

B: Movies with highest profit-

	color	director_name	num_critic_for_reviews	actor_2_name	gross	movie_title	country	content_rating	budget	imdb_score	aspect_ratio	movie_facebook_likes	Profit
0	Color	James Cameron	723.0	Joel David Moore	760505847.0	Avatar	USA	PG-13	2.370000e+08	7.9	1.78	33000	5.235058e+08
29	Color	Colin Trevorrow	644.0	Judy Greer	652177271.0	Jurassic World	USA	PG-13	1.500000e+08	7.0	2.00	150000	5.021773e+08
26	Color	James Cameron	315.0	Kate Winslet	658672302.0	Titanic	USA	PG-13	2.000000e+08	7.7	2.35	26000	4.586723e+08
3024	Color	George Lucas	282.0	Peter Cushing	460935665.0	Star Wars: Episode IV - A New Hope	USA	PG	1.100000e+07	8.7	2.35	33000	4.499357e+08
3080	Color	Steven Spielberg	215.0	Dee Wallace	434949459.0	E.T. the Extra-Terrestrial	USA	PG	1.050000e+07	7.9	1.85	34000	4.244495e+08
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2334	Color	Katsuhiro Ôtomo	105.0	Robin Atkin Downes	410388.0	Steamboy	Japan	PG-13	2.127520e+09	6.9	1.85	973	-2.127110e+09
2323	Color	Hayao Miyazaki	174.0	Jada Pinkett Smith	2298191.0	Princess Mononoke	Japan	PG-13	2.400000e+09	8.4	1.85	11000	-2.397702e+09
3005	Color	Lajos Koltai	73.0	Péter Fancsikai	195888.0	Fateless	Hungary	R	2.500000e+09	7.1	2.35	607	-2.499804e+09
3859	Color	Chan-wook Park	202.0	Yeong-ae Lee	211667.0	Lady Vengeance	South Korea	R	4.200000e+09	7.7	2.35	4000	-4.199788e+09
2988	Color	Joon-ho Bong	363.0	Kang-ho Song	2201412.0	The Host	South Korea	R	1.221550e+10	7.0	1.85	7000	-1.221330e+10

C: Top 250 Movies -

	Rank	movie_title	imdb_score	language
1937	1	The Shawshank Redemption	9.3	English
3466	2	The Godfather	9.2	English
3481	3	Fargo	9.0	English
66	4	The Dark Knight	9.0	English
2837	5	The Godfather: Part II	9.0	English
...	...	...	...	...
4266	246	Before Sunset	8.0	English
602	247	Big Fish	8.0	English
1603	248	Mystic River	8.0	English
4261	249	The Hustler	8.0	English
1601	250	District 9	8.0	English

Top Foreign Films –

	Rank	movie_title	imdb_score	language
4498	6	The Good, the Bad and the Ugly	8.9	Italian
4747	23	Seven Samurai	8.7	Japanese
4029	25	City of God	8.7	Portuguese
2373	27	Spirited Away	8.6	Japanese
3870	42	Airlift	8.5	Hindi
4259	55	The Lives of Others	8.5	German
4921	60	Children of Heaven	8.5	Persian
4105	64	Oldboy	8.4	Korean
4659	73	A Separation	8.4	Persian
3685	75	Rang De Basanti	8.4	Hindi
2970	77	Das Boot	8.4	German
2323	79	Princess Mononoke	8.4	Japanese
1329	82	Baahubali: The Beginning	8.4	Telugu
1298	83	Amélie	8.4	French
2829	87	Downfall	8.3	German
4033	99	The Hunt	8.3	Danish
2734	100	Metropolis	8.3	German
2551	118	Pan's Labyrinth	8.2	Spanish
3550	119	Incendies	8.2	French
2047	128	Howl's Moving Castle	8.2	Japanese
4000	130	The Secret in Their Eyes	8.2	Spanish
4160	146	Lage Raho Munna Bhai	8.2	Hindi
1061	157	Solaris	8.1	Russian
4461	161	The Celebration	8.1	Danish
3553	178	Elite Squad	8.1	Portuguese
2830	181	The Sea Inside	8.1	Spanish
3423	191	Akira	8.1	Japanese
2914	192	Tae Guk Gi: The Brotherhood of War	8.1	Korean
4267	195	Amores Perros	8.1	Spanish
2802	206	The Diving Bell and the Butterfly	8.0	French
3344	221	My Name Is Khan	8.0	Hindi
2739	224	The Return	8.0	Russian
3456	227	Persepolis	8.0	French
4144	238	Central Station	8.0	Portuguese
4284	243	Waltz with Bashir	8.0	Hebrew

D: Best Directors-

```
director_name
John Blanchard        9.5
Cary Bell             8.7
Mitchell Altieri      8.7
Sadyk Sher-Niyaz      8.7
Charles Chaplin       8.6
Mike Mayhall          8.6
Damien Chazelle       8.5
Majid Majidi          8.5
Raja Menon            8.5
Ron Fricke            8.5
```

E: Popular Generes -

```
Drama          173
Adventure       66
Thriller        56
Crime           51
Action          50
Comedy          41
Sci-Fi          40
Romance         35
Biography       30
Fantasy         30
War             29
Mystery         28
Family          24
Animation       20
History         18
Sport            9
Horror           8
Western          8
Musical          7
Documentary      5
Music            3
Film-Noir        1
```

F: Charts : Critic Favorite and Audience Favorite Actors -

	director_name	num_critic_for_reviews	gross	genres	actor_1_name	movie_title	num_voted_users	num_user_for_reviews	language	budget	title_year	imdb_score	Profit
410	Nancy Meyers	187.0	112703470.0	Comedy|Drama|Romance	Meryl Streep	It's Complicated	69860	214	English	85000000.0	2009.0	6.6	27703470.0
1106	Curtis Hanson	42.0	46815748.0	Action|Adventure|Crime|Thriller	Meryl Streep	The River Wild	32544	69	English	45000000.0	1994.0	6.3	1815748.0
1204	Nora Ephron	252.0	94125426.0	Biography|Drama|Romance	Meryl Streep	Julie & Julia	79264	277	English	40000000.0	2009.0	7.0	54125426.0
1408	David Frankel	208.0	124732962.0	Comedy|Drama|Romance	Meryl Streep	The Devil Wears Prada	286178	631	English	35000000.0	2006.0	6.8	89732962.0
1483	Robert Redford	227.0	14998070.0	Drama|Thriller|War	Meryl Streep	Lions for Lambs	41170	298	English	35000000.0	2007.0	6.2	-20001930.0
1575	Sydney Pollack	66.0	87100000.0	Biography|Drama|Romance	Meryl Streep	Out of Africa	52339	200	English	31000000.0	1985.0	7.2	56100000.0
1618	David Frankel	234.0	63536011.0	Comedy|Drama|Romance	Meryl Streep	Hope Springs	34258	178	English	30000000.0	2012.0	6.3	33536011.0
1674	Carl Franklin	64.0	23209440.0	Drama	Meryl Streep	One True Thing	9283	112	English	30000000.0	1998.0	7.0	-6790560.0
1752	Stephen Frears	87.0	NaN	Biography|Comedy|Drama|Music|Romance	Meryl Streep	Florence Foster Jenkins	2167	32	English	29000000.0	2016.0	7.1	NaN
1925	Stephen Daldry	174.0	41597830.0	Drama|Romance	Meryl Streep	The Hours	102123	660	English	25000000.0	2002.0	7.6	16597830.0
2781	Phyllida Lloyd	331.0	29959436.0	Biography|Drama|History	Meryl Streep	The Iron Lady	82327	350	English	13000000.0	2011.0	6.4	16959436.0
3135	Robert Altman	211.0	20338609.0	Comedy|Drama|Music	Meryl Streep	A Prairie Home Companion	19655	280	English	10000000.0	2006.0	6.8	10338609.0
3641	Fred Zinnemann	38.0	NaN	Drama	Meryl Streep	Julia	6454	44	English	7840000.0	1977.0	7.4	NaN
26	James Cameron	315.0	658672302.0	Drama|Romance	Leonardo DiCaprio	Titanic	793059	2528	English	200000000.0	1997.0	7.7	458672302.0
50	Baz Luhrmann	490.0	144812796.0	Drama|Romance	Leonardo DiCaprio	The Great Gatsby	362912	753	English	105000000.0	2013.0	7.3	39812796.0
97	Christopher Nolan	642.0	292568851.0	Action|Adventure|Sci-Fi|Thriller	Leonardo DiCaprio	Inception	1468200	2803	English	160000000.0	2010.0	8.8	132568851.0
179	Alejandro G. Iñárritu	556.0	183635922.0	Adventure|Drama|Thriller|Western	Leonardo DiCaprio	The Revenant	406020	1188	English	135000000.0	2015.0	8.1	48635922.0
257	Martin Scorsese	267.0	102608827.0	Biography|Drama	Leonardo DiCaprio	The Aviator	264318	799	English	110000000.0	2004.0	7.5	-7391173.0
296	Quentin Tarantino	765.0	162804648.0	Drama|Western	Leonardo DiCaprio	Django Unchained	955174	1193	English	100000000.0	2012.0	8.5	62804648.0
307	Edward Zwick	166.0	57366262.0	Adventure|Drama|Thriller	Leonardo DiCaprio	Blood Diamond	400292	657	English	100000000.0	2006.0	8.0	-42633738.0
308	Martin Scorsese	606.0	116866727.0	Biography|Comedy|Crime|Drama	Leonardo DiCaprio	The Wolf of Wall Street	780588	1138	English	100000000.0	2013.0	8.2	16866727.0
326	Martin Scorsese	233.0	77679638.0	Crime|Drama	Leonardo DiCaprio	Gangs of New York	314033	1166	English	100000000.0	2002.0	7.5	-22320362.0
361	Martin Scorsese	352.0	132373442.0	Crime|Drama|Thriller	Leonardo DiCaprio	The Departed	873649	2054	English	90000000.0	2006.0	8.5	42373442.0
452	Martin Scorsese	490.0	127968405.0	Mystery|Thriller	Leonardo DiCaprio	Shutter Island	786092	964	English	80000000.0	2010.0	8.1	47968405.0
641	Ridley Scott	238.0	39380442.0	Action|Drama|Thriller	Leonardo DiCaprio	Body of Lies	174248	263	English	70000000.0	2008.0	7.1	-30619558.0
911	Steven Spielberg	194.0	164435221.0	Biography|Crime|Drama	Leonardo DiCaprio	Catch Me If You Can	525801	667	English	52000000.0	2002.0	8.0	112435221.0
990	Danny Boyle	118.0	39778599.0	Adventure|Drama|Thriller	Leonardo DiCaprio	The Beach	176169	548	English	50000000.0	2000.0	6.6	-10221401.0
1114	Sam Mendes	323.0	22877808.0	Drama|Romance	Leonardo DiCaprio	Revolutionary Road	152591	414	English	35000000.0	2008.0	7.3	-12122192.0
1422	Randall Wallace	83.0	56876365.0	Action|Adventure	Leonardo DiCaprio	The Man in the Iron Mask	125219	244	English	35000000.0	1998.0	6.4	21876365.0
1453	Clint Eastwood	392.0	37304950.0	Biography|Crime|Drama	Leonardo DiCaprio	J. Edgar	102728	279	English	35000000.0	2011.0	6.6	2304950.0
1560	Sam Raimi	63.0	18636537.0	Action|Thriller|Western	Leonardo DiCaprio	The Quick and the Dead	69197	216	English	32000000.0	1995.0	6.4	-13363463.0
2067	Jerry Zaks	45.0	12782508.0	Drama	Leonardo DiCaprio	Marvin's Room	20163	71	English	23000000.0	1996.0	6.7	-10217492.0
2757	Baz Luhrmann	106.0	46338728.0	Drama|Romance	Leonardo DiCaprio	Romeo + Juliet	167750	506	English	14500000.0	1996.0	6.8	31838728.0
3476	Baz Luhrmann	490.0	144812796.0	Drama|Romance	Leonardo DiCaprio	The Great Gatsby	362933	753	English	105000000.0	2013.0	7.3	39812796.0
26	James Cameron	315.0	658672302.0	Drama|Romance	Leonardo DiCaprio	Titanic	793059	2528	English	200000000.0	1997.0	7.7	458672302.0
50	Baz Luhrmann	490.0	144812796.0	Drama|Romance	Leonardo DiCaprio	The Great Gatsby	362912	753	English	105000000.0	2013.0	7.3	39812796.0
97	Christopher Nolan	642.0	292568851.0	Action|Adventure|Sci-Fi|Thriller	Leonardo DiCaprio	Inception	1468200	2803	English	160000000.0	2010.0	8.8	132568851.0
179	Alejandro G. Iñárritu	556.0	183635922.0	Adventure|Drama|Thriller|Western	Leonardo DiCaprio	The Revenant	406020	1188	English	135000000.0	2015.0	8.1	48635922.0
257	Martin Scorsese	267.0	102608827.0	Biography|Drama	Leonardo DiCaprio	The Aviator	264318	799	English	110000000.0	2004.0	7.5	-7391173.0
296	Quentin Tarantino	765.0	162804648.0	Drama|Western	Leonardo DiCaprio	Django Unchained	955174	1193	English	100000000.0	2012.0	8.5	62804648.0
307	Edward Zwick	166.0	57366262.0	Adventure|Drama|Thriller	Leonardo DiCaprio	Blood Diamond	400292	657	English	100000000.0	2006.0	8.0	-42633738.0
308	Martin Scorsese	606.0	116866727.0	Biography|Comedy|Crime|Drama	Leonardo DiCaprio	The Wolf of Wall Street	780588	1138	English	100000000.0	2013.0	8.2	16866727.0
326	Martin Scorsese	233.0	77679638.0	Crime|Drama	Leonardo DiCaprio	Gangs of New York	314033	1166	English	100000000.0	2002.0	7.5	-22320362.0
361	Martin Scorsese	352.0	132373442.0	Crime|Drama|Thriller	Leonardo DiCaprio	The Departed	873649	2054	English	90000000.0	2006.0	8.5	42373442.0
452	Martin Scorsese	490.0	127968405.0	Mystery|Thriller	Leonardo DiCaprio	Shutter Island	786092	964	English	80000000.0	2010.0	8.1	47968405.0
641	Ridley Scott	238.0	39380442.0	Action|Drama|Thriller	Leonardo DiCaprio	Body of Lies	174248	263	English	70000000.0	2008.0	7.1	-30619558.0
911	Steven Spielberg	194.0	164435221.0	Biography|Crime|Drama	Leonardo DiCaprio	Catch Me If You Can	525801	667	English	52000000.0	2002.0	8.0	112435221.0
990	Danny Boyle	118.0	39778599.0	Adventure|Drama|Thriller	Leonardo DiCaprio	The Beach	176169	548	English	50000000.0	2000.0	6.6	-10221401.0
1114	Sam Mendes	323.0	22877808.0	Drama|Romance	Leonardo DiCaprio	Revolutionary Road	152591	414	English	35000000.0	2008.0	7.3	-12122192.0
1422	Randall Wallace	83.0	56876365.0	Action|Adventure	Leonardo DiCaprio	The Man in the Iron Mask	125219	244	English	35000000.0	1998.0	6.4	21876365.0
1453	Clint Eastwood	392.0	37304950.0	Biography|Crime|Drama	Leonardo DiCaprio	J. Edgar	102728	279	English	35000000.0	2011.0	6.6	2304950.0
1560	Sam Raimi	63.0	18636537.0	Action|Thriller|Western	Leonardo DiCaprio	The Quick and the Dead	69197	216	English	32000000.0	1995.0	6.4	-13363463.0
2067	Jerry Zaks	45.0	12782508.0	Drama	Leonardo DiCaprio	Marvin's Room	20163	71	English	23000000.0	1996.0	6.7	-10217492.0
2757	Baz Luhrmann	106.0	46338728.0	Drama|Romance	Leonardo DiCaprio	Romeo + Juliet	167750	506	English	14500000.0	1996.0	6.8	31838728.0
3476	Baz Luhrmann	490.0	144812796.0	Drama|Romance	Leonardo DiCaprio	The Great Gatsby	362933	753	English	105000000.0	2013.0	7.3	39812796.0

Code for IMDB Movie Analysis

```
import numpy as np
import pandas as pd
```

Loading Dataset

```
data=pd.read_csv("/content/IMDB_Movies.csv")
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5043 entries, 0 to 5042
Data columns (total 28 columns):
 #   Column                     Non-Null Count  Dtype  
---  ------                     --------------  -----  
 0   color                      5024 non-null   object 
 1   director_name              4939 non-null   object 
 2   num_critic_for_reviews     4993 non-null   float64
 3   duration                   5028 non-null   float64
 4   director_facebook_likes    4939 non-null   float64
 5   actor_3_facebook_likes     5020 non-null   float64
 6   actor_2_name               5030 non-null   object 
 7   actor_1_facebook_likes     5036 non-null   float64
 8   gross                      4159 non-null   float64
 9   genres                     5043 non-null   object 
 10  actor_1_name               5036 non-null   object 
 11  movie_title                5043 non-null   object 
 12  num_voted_users            5043 non-null   int64  
 13  cast_total_facebook_likes  5043 non-null   int64  
 14  actor_3_name               5020 non-null   object 
 15  facenumber_in_poster       5030 non-null   float64
 16  plot_keywords              4890 non-null   object 
 17  movie_imdb_link            5043 non-null   object 
 18  num_user_for_reviews       5023 non-null   object 
 19  language                   5031 non-null   object 
 20  country                    5038 non-null   object 
 21  content_rating             4740 non-null   object 
 22  budget                     4551 non-null   float64
 23  title_year                 4935 non-null   float64
 24  actor_2_facebook_likes     5030 non-null   float64
 25  imdb_score                 5043 non-null   float64
 26  aspect_ratio               4714 non-null   float64
 27  movie_facebook_likes       5043 non-null   int64  
dtypes: float64(12), int64(3), object(13)
memory usage: 1.1+ MB
```

```
data.head
```

```
5039                        NaN                   319.0     Valorie Curry
5040                        0.0                     0.0     Maxwell Moody
5041                        0.0                   489.0     Daniel Henney
5042                       16.0                    16.0  Brian Herzlinger

      actor_1_facebook_likes        gross                           genres  \
0                     1000.0  760505847.0  Action|Adventure|Fantasy|Sci-Fi
1                    40000.0  309404152.0         Action|Adventure|Fantasy
2                    11000.0  200074175.0        Action|Adventure|Thriller
3                    27000.0  448130642.0                  Action|Thriller
4                      131.0          NaN                      Documentary
...                      ...          ...                              ...
```

```
1        2007.0              5000.0      7.1        2.35
2        2015.0               393.0      6.8        2.35
3        2012.0             23000.0      8.5        2.35
4           NaN                12.0      7.1         NaN
...         ...                 ...      ...         ...
5038     2013.0               470.0      7.7         NaN
5039        NaN               593.0      7.5       16.00
5040     2013.0                 0.0      6.3         NaN
5041     2012.0               719.0      6.3        2.35
5042     2004.0                23.0      6.6        1.85

     movie_facebook_likes
0                   33000
1                       0
2                   85000
3                  164000
4                       0
...                   ...
5038                   84
5039                32000
5040                   16
5041                  660
5042                  456

[5043 rows x 28 columns]>
```

```
data.isna().sum()
#through isna() function we can check how many null values are there
```

```
color                         19
director_name                104
num_critic_for_reviews        50
duration                      15
director_facebook_likes      104
actor_3_facebook_likes        23
actor_2_name                  13
actor_1_facebook_likes         7
gross                        884
genres                         0
actor_1_name                   7
movie_title                    0
num_voted_users                0
cast_total_facebook_likes      0
actor_3_name                  23
facenumber_in_poster          13
plot_keywords                153
movie_imdb_link                0
num_user_for_reviews          20
language                      12
country                        5
content_rating               303
budget                       492
title_year                   108
actor_2_facebook_likes        13
imdb_score                     0
aspect_ratio                 329
movie_facebook_likes           0
dtype: int64
```

```
data.describe()
```

	num_critic_for_reviews	duration	director_facebook_likes	actor_3_fac
count	4993.000000	5028.000000	4939.000000	
mean	140.194272	107.201074	686.509212	
std	121.601675	25.197441	2813.328607	
min	1.000000	7.000000	0.000000	
25%	50.000000	93.000000	7.000000	
50%	110.000000	103.000000	49.000000	
75%	195.000000	118.000000	194.500000	
max	813.000000	511.000000	23000.000000	

A. Cleaning the Data

```
#droping unnecessary columns
data.drop(["color","director_facebook_likes","actor_2_name","cast_total_facebook_likes","actor_3_name",
          "country","content_rating","aspect_ratio","movie_facebook_likes","actor_1_facebook_likes","duration",
          "actor_3_facebook_likes","facenumber_in_poster",'plot_keywords',"movie_imdb_link",'actor_2_facebook_likes'],
          axis=1,inplace=True)
```

```
data.info()
data.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5043 entries, 0 to 5042
Data columns (total 12 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   director_name           4939 non-null   object 
 1   num_critic_for_reviews  4993 non-null   float64
 2   gross                   4159 non-null   float64
 3   genres                  5043 non-null   object 
 4   actor_1_name            5036 non-null   object 
 5   movie_title             5043 non-null   object 
 6   num_voted_users         5043 non-null   int64  
 7   num_user_for_reviews    5023 non-null   object 
 8   language                5031 non-null   object 
 9   budget                  4551 non-null   float64
 10  title_year              4935 non-null   float64
 11  imdb_score              5043 non-null   float64
dtypes: float64(5), int64(1), object(6)
memory usage: 472.9+ KB
```

<table>
<tr><th></th><th>director_name</th><th>num_critic_for_reviews</th><th>gross</th><th>genres</th><th>actor_1_name</th><th>movie_title</th><th>num_voted_users</th></tr>
<tr><td>0</td><td>James Cameron</td><td>723.0</td><td>760505847.0</td><td>Action|Adventure|Fantasy|Sci-Fi</td><td>CCH Pounder</td><td>Avatar</td><td>886204</td></tr>
<tr><td>1</td><td>Gore Verbinski</td><td>302.0</td><td>309404152.0</td><td>Action|Adventure|Fantasy</td><td>Johnny Depp</td><td>Pirates of the Caribbean: At World's End</td><td>471220</td></tr>
<tr><td>2</td><td>Sam Mendes</td><td>602.0</td><td>200074175.0</td><td>Action|Adventure|Thriller</td><td>Christoph Waltz</td><td>Spectre</td><td>275868</td></tr>
<tr><td>3</td><td>Christopher Nolan</td><td>813.0</td><td>448130642.0</td><td>Action|Thriller</td><td>Tom Hardy</td><td>The Dark Knight Rises</td><td>1144337</td></tr>
<tr><td>4</td><td>Doug Walker</td><td>NaN</td><td>NaN</td><td>Documentary</td><td>Doug Walker</td><td>Star Wars: Episode VII - The Force Awakens ...</td><td>8</td></tr>
</table>

```
data.replace({'director_name':np.NaN},value="None",inplace=True)
data['num_critic_for_reviews']=data['num_critic_for_reviews'].fillna(value=data['num_critic_for_reviews'].mean())
data.drop(data.index[4],inplace=True)
```

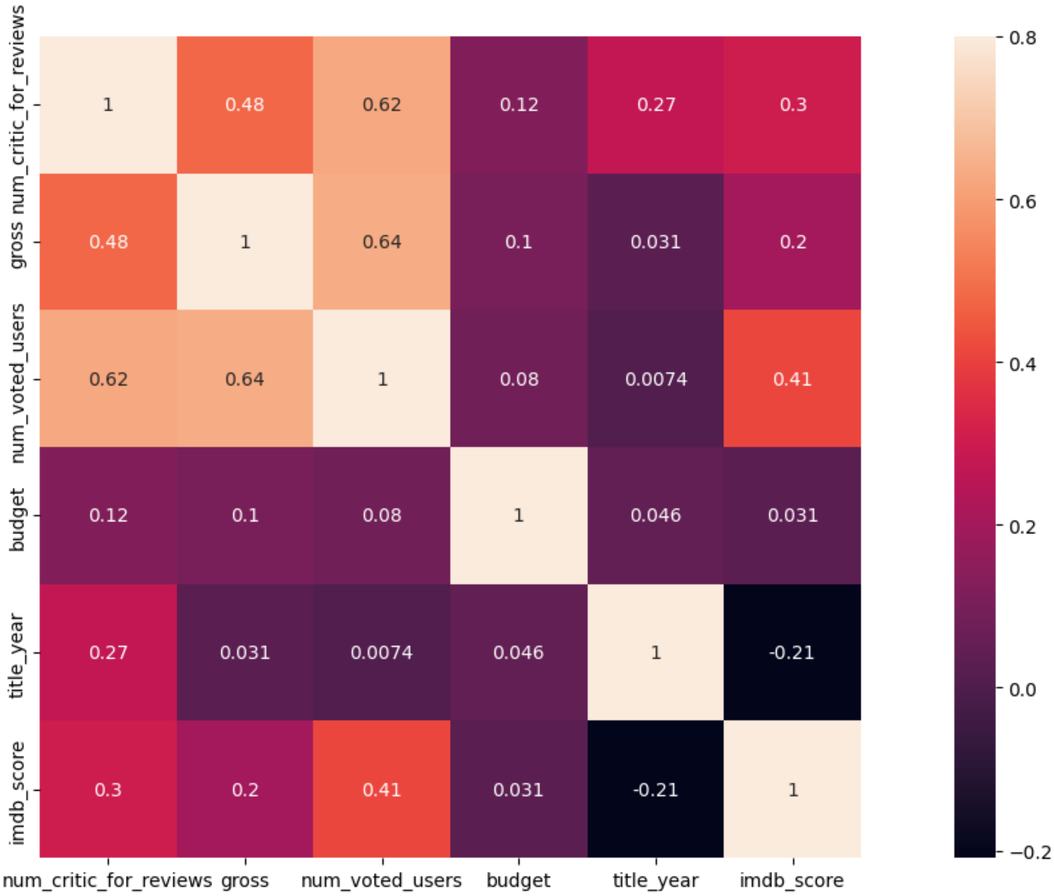
```
data.head(100)
```

<table>
<tr><th></th><th>director_name</th><th>num_critic_for_reviews</th><th>gross</th><th>genres</th><th>actor_1_name</th><th>movie_title</th><th>num_voted_u</th></tr>
<tr><td>0</td><td>James Cameron</td><td>723.0</td><td>760505847.0</td><td>Action|Adventure|Fantasy|Sci-Fi</td><td>CCH Pounder</td><td>Avatar</td><td>8</td></tr>
<tr><td>1</td><td>Gore Verbinski</td><td>302.0</td><td>309404152.0</td><td>Action|Adventure|Fantasy</td><td>Johnny Depp</td><td>Pirates of the Caribbean: At World's End</td><td>4</td></tr>
</table>

```
import matplotlib.pyplot as plt
import seaborn as sns

#plotting heat map:
plt.figure(figsize=(18,8),dpi=100,)
plt.subplots(figsize=(18,8))
sns.heatmap(data=data.corr(),square=True,vmax=0.8,annot=True)
```

```
<Axes: >
<Figure size 1800x800 with 0 Axes>
```




```
data['budget']=data['budget'].fillna(data['budget'].mean())
data.head(10)
```

	director_name	num_critic_for_reviews	gross	genres	actor_1_name	movie_title
0	James Cameron	723.0	760505847.0	Action|Adventure|Fantasy|Sci-Fi	CCH Pounder	Avatar
1	Gore Verbinski	302.0	309404152.0	Action|Adventure|Fantasy	Johnny Depp	Pirates of the Caribbean: At World's End

```
#data.to_csv(r'CleanedData.csv', index=None)
```

Christopher The Dark

B. Movies with highest profit

6 Sam Raimi 392.0 336530303.0 Action|Adventure|Romance J.K. Simmons Spider-Man 3

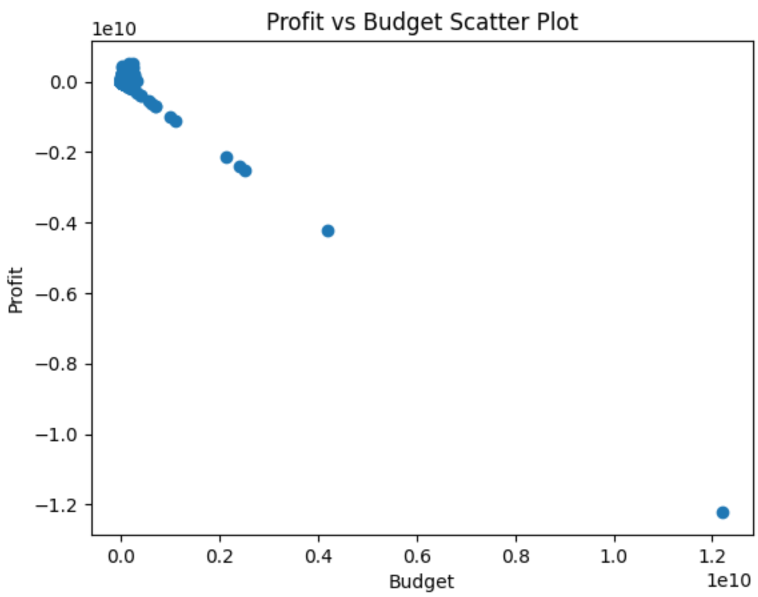
```
data['Profit'] = data['gross'] - data['budget']

# Sort the dataset based on the "Profit" column
df = data.sort_values(by='Profit', ascending=False)
```

Harry Potter

```
# Plot a scatter plot with "Budget" on the x-axis and "Profit" on the y-axis
plt.scatter(df['budget'], df['Profit'])
plt.xlabel('Budget')
plt.ylabel('Profit')
plt.title('Profit vs Budget Scatter Plot')

# Display the plot
plt.show()
```



C. Top 250

```
top_250=data.copy()
# Filter movies with more than 25,000 votes
top_250 = top_250[top_250['num_voted_users'] > 25000]

# Sort the dataset by IMDb rating in descending order
top_250 = top_250.sort_values('imdb_score', ascending=False)

# Select the top 250 movies
top_250 = top_250[:250]

# Create a new column "Rank" and set it to the rank of the corresponding film
top_250['Rank'] = range(1, 251)

# Display the top 250 movies with their rank
top_250[['Rank', 'movie_title', 'imdb_score', 'language']]
```

	Rank	movie_title	imdb_score	language
1937	1	The Shawshank Redemption	9.3	English
3466	2	The Godfather	9.2	English
3481	3	Fargo	9.0	English
66	4	The Dark Knight	9.0	English
2837	5	The Godfather: Part II	9.0	English
...	...	...	...	...
4266	246	Before Sunset	8.0	English
602	247	Big Fish	8.0	English

```
# create a new column Top_Foreign_Lang_Film
Top_Foreign_Lang_Film = top_250[top_250['language']!="English"]
Top_Foreign_Lang_Film[['Rank', 'movie_title', 'imdb_score', 'language']]
```

2373	27	Spirited Away	8.6	Japanese
3870	42	Airlift	8.5	Hindi
4259	55	The Lives of Others	8.5	German
4921	60	Children of Heaven	8.5	Persian
4105	64	Oldboy	8.4	Korean
4659	73	A Separation	8.4	Persian
3685	75	Rang De Basanti	8.4	Hindi
2970	77	Das Boot	8.4	German
2323	79	Princess Mononoke	8.4	Japanese
1329	82	Baahubali: The Beginning	8.4	Telugu
1298	83	Amélie	8.4	French
2829	87	Downfall	8.3	German
4033	99	The Hunt	8.3	Danish
2734	100	Metropolis	8.3	German
2551	118	Pan's Labyrinth	8.2	Spanish
3550	119	Incendies	8.2	French
2047	128	Howl's Moving Castle	8.2	Japanese
4000	130	The Secret in Their Eyes	8.2	Spanish
4160	146	Lage Raho Munna Bhai	8.2	Hindi
1061	157	Solaris	8.1	Russian
4461	161	The Celebration	8.1	Danish
3553	178	Elite Squad	8.1	Portuguese
2830	181	The Sea Inside	8.1	Spanish
3423	191	Akira	8.1	Japanese
2914	192	Tae Guk Gi: The Brotherhood of War	8.1	Korean
4267	195	Amores Perros	8.1	Spanish
2802	206	The Diving Bell and the Butterfly	8.0	French
3344	221	My Name Is Khan	8.0	Hindi
2739	224	The Return	8.0	Russian
3456	227	Persepolis	8.0	French
4144	238	Central Station	8.0	Portuguese
4284	243	Waltz with Bashir	8.0	Hebrew

D. Best Directors

```
# Group the dataset by director_name column
director_group = data.groupby("director_name")

# Calculate the mean of imdb_score for each director and sort them in descending order
mean_scores = director_group["imdb_score"].mean().sort_values(ascending=False)

# Break ties by sorting directors alphabetically
top_directors = mean_scores.sort_index().nlargest(10)

# Print the top 10 directors with the highest mean imdb_score
top_directors
```

```
director_name
John Blanchard      9.5
Cary Bell           8.7
Mitchell Altieri    8.7
Sadyk Sher-Niyaz    8.7
Charles Chaplin     8.6
Mike Mayhall        8.6
Damien Chazelle     8.5
Majid Majidi        8.5
Raja Menon          8.5
Ron Fricke          8.5
Name: imdb_score, dtype: float64
```

E. Popular Genres

```
genres = top_250['genres'].str.split('|', expand=True).stack().reset_index(level=1, drop=True)

popular_genres = genres.value_counts().sort_values(ascending=False)
popular_genres
```

```
Drama          173
Adventure       66
Thriller        56
Crime           51
Action          50
Comedy          41
Sci-Fi          40
Romance         35
Biography       30
Fantasy         30
War             29
Mystery         28
Family          24
Animation       20
History         18
Sport            9
Horror           8
Western          8
Musical          7
Documentary      5
Music            3
Film-Noir        1
dtype: int64
```

F. Charts - The Critic Favorite and Audience Favorite Actors

```
movies=data.copy()
# create dataframes for each actor
meryl_df = movies[movies['actor_1_name'] == 'Meryl Streep']
leo_df = movies[movies['actor_1_name'] == 'Leonardo DiCaprio']
brad_df = movies[movies['actor_1_name'] == 'Leonardo DiCaprio']

# Create Combined column
combined = pd.concat([movies[movies['actor_1_name'] == 'Meryl Streep'], movies[movies['actor_1_name'] == 'Leonardo DiCaprio']

#display
combined
```

	director_name	num_critic_for_reviews	gross	
410	Nancy Meyers	187.0	112703470.0	ComedylDr
1106	Curtis Hanson	42.0	46815748.0	ActionlAdventure
1204	Nora Ephron	252.0	94125426.0	BiographylDr
1408	David Frankel	208.0	124732962.0	ComedylDr
1483	Robert Redford	227.0	14998070.0	Dran
1575	Sydney Pollack	66.0	87100000.0	BiographylDr
1618	David Frankel	234.0	63536011.0	ComedylDr
1674	Carl Franklin	64.0	23209440.0	
1752	Stephen Frears	87.0	NaN	BiographylComedylDramalM
1925	Stephen Daldry	174.0	41597830.0	Dr
2781	Phyllida Lloyd	331.0	29959436.0	Biographyl
3135	Robert Altman	211.0	20338609.0	Comedy
3641	Fred Zinnemann	38.0	NaN	
26	James Cameron	315.0	658672302.0	Dr
50	Baz Luhrmann	490.0	144812796.0	Dr
97	Christopher Nolan	642.0	292568851.0	ActionlAdventure
179	Alejandro G. Iñárritu	556.0	183635922.0	AdventurelDramalT
257	Martin Scorsese	267.0	102608827.0	Bio
296	Quentin Tarantino	765.0	162804648.0	D
307	Edward Zwick	166.0	57366262.0	Adventurel
308	Martin Scorsese	606.0	116866727.0	BiographylComedy
326	Martin Scorsese	233.0	77679638.0	

```
actor_group = data.groupby('actor_1_name')

actor_mean = actor_group[['num_critic_for_reviews', 'num_user_for_reviews']].mean()
actor_mean
```

actor_1_name	num_critic_for_reviews	num_user_for_reviews
50 Cent	98.000000	284.000000
A.J. Buckley	298.000000	345.000000
Aaliyah	137.000000	695.000000
Aasif Mandvi	210.000000	147.000000
Abbie Cornish	270.333333	184.666667
...	...	...
Zoë Kravitz	114.666667	93.666667
Zuhair Haddad	5.000000	1.000000
Álex Angulo	9.000000	7.000000
Ólafur Darri Ólafsson	16.000000	19.000000
Óscar Jaenada	186.000000	139.000000

2097 rows × 2 columns

```
critic_df = actor_mean.sort_values('num_critic_for_reviews', ascending=False)
print(critic_df["num_critic_for_reviews"])
```

```
actor_1_name
Phaldut Sharma          738.0
Peter Capaldi           654.0
Craig Stark             596.0
Bérénice Bejo           576.0
Suraj Sharma            552.0
                        ...
Mike Stanley              1.0
Mike Beckingham           1.0
Marcello Mastroianni      1.0
Manny Perez               1.0
Carrie Bradstreet         1.0
Name: num_critic_for_reviews, Length: 2097, dtype: float64
```

```
audience_df = actor_mean.sort_values('num_user_for_reviews', ascending=False)
audience_df["num_user_for_reviews"]
```

```
actor_1_name
Heather Donahue         3400.0
Christo Jivkov          2814.0
Steve Bastoni           2789.0
Phaldut Sharma          1885.0
Keir Dullea             1736.0
                         ...
Jon Brion                  1.0
Patrick O'Donnell          1.0
Mary Kate Wiles            1.0
Paul Hickert               1.0
Claire Gordon-Harper       1.0
Name: num_user_for_reviews, Length: 2097, dtype: float64
```

```
total_df= actor_mean["num_critic_for_reviews"].add(actor_mean['num_user_for_reviews'],fill_value=0).to_frame()
total_df=total_df.sort_values(0,ascending=False)
result = total_df.head(10)
result
```

actor_1_name	0
Heather Donahue	3760.0
Christo Jivkov	3220.0
Steve Bastoni	3064.0

```
chartdata=data.copy()
bins = [1920, 1930, 1940, 1950, 1960, 1970, 1980, 1990, 2000, 2010, 2020]
labels = ['1920s', '1930s', '1940s', '1950s', '1960s', '1970s', '1980s', '1990s', '2000s', '2010s']
chartdata['decade'] = pd.cut(chartdata['title_year'], bins=bins, labels=labels)


df_by_decade = chartdata.groupby('decade')['num_voted_users'].sum().reset_index()
df_by_decade
```

	decade	num_voted_users
0	1920s	132420
1	1930s	1233065
2	1940s	962634
3	1950s	2175102
4	1960s	4819970
5	1970s	13740773
6	1980s	24616391
7	1990s	80028936
8	2000s	186323739
9	2010s	104763014

```
plt.bar(df_by_decade['decade'], df_by_decade['num_voted_users'])
plt.xlabel('Decade')
plt.ylabel('Number of Voted Users')
plt.title('Change in Number of Voted Users over Decades')
plt.show()
```