

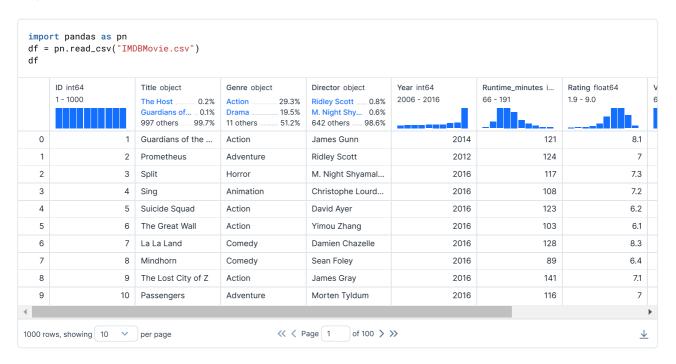




# Project\_1

# **Analyzing IMDB Movie Dataset**

The Objective of this project is to analyze the IMDb Movie Dataset to uncover trends in ratings, genres and Revenues and Analyzing Based on the given Questions ..



# TASK-1

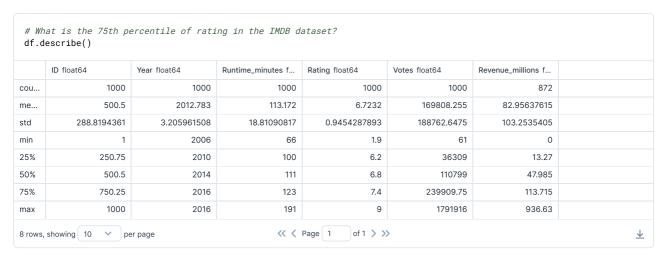
```
#How many rows are there in the IMDB dataset?

df.shape[0]

1000
```

Code Commentary: df.shape[0] retrieves the number of rows present in the Dataframe named as df.

# TASK-2



**Code Commentary :** The function df.describe() provides summary statistics for numerical columns in the DataFrame df, including count, mean, standard deviation, minimum, quartiles, and maximum values.

# TASK-3

```
# How many NA values are there in the field 'Revenue'?
df['Revenue_millions'].isnull().sum()
```

Code Commentary: Using sum function calculates the total number of missing values in the 'Revenue\_millions' column of the DataFrame df.

### TASK-4

```
#How many movies have revenue higher than 75 million?
(df['Revenue_millions'] > 75).sum()
```

Code Commentary: Count the Sum of occurrences where the 'Revenue\_millions' column in DataFrame df has a value greater than 75.

# TASK-5

```
# How many movies have revenue great"er than 50 million but rating less than 7?
df[(df["Revenue_millions"] > 50) & (df["Rating"] < 7)]["Title"].count()</pre>
211
```

**Code Commentary:** Counts the number of movies in DataFrame df Using count function that have a revenue greater than 50 million and a rating less than 7.

# TASK-6

```
# What is the total revenue generated by movies in the year 2015?
df[df["Year"] == 2015]["Revenue_millions"].sum()

8854.119999999999
```

Code Commentary: Calculates the total revenue generated by movies released in the year 2015 in the DataFrame df.

### TASK-7

```
# What is the average rating for the genre adventure in the year 2015?
df[(df["Genre"] == "Adventure") & (df["Year"] == 2015)]["Rating"].mean()
6.8
```

Code Commentary: Computes the average rating of adventure genre movies released in the year 2015 in the DataFrame df.

# TASK-8

```
# What is the average duration of movies in rows 75 to 150? Please note that the rows in python start from 0.

df.iloc[75:151]["Runtime_minutes"].mean()

#OR

cond = df.iloc[75:151]

cond ["Runtime_minutes"].mean()

127.47368421052632
```

Code Commentary: Using mean function Calculates the average runtime in minutes for the movies indexed from 75 to 150 in the DataFrame df.

# TASK-9

```
# Which year generated the highest revenue?
df.groupby(by = "Year")["Revenue_millions"].sum().sort_values(ascending = False)
Year
       11211.65
2016
2015
        8854.12
        7997.40
2014
2013
        7666.72
2012
         6910.29
2010
         5989.65
2011
         5431.96
         5292.26
2009
2008
         5053.22
2007
         4306.23
2006
         3624.46
Name: Revenue_millions, dtype: float64
```

Code Commentary: Here Using groupby function groups the DataFrame df by the "Year" column, calculates the sum of "Revenue\_millions" for each year, and show the results in descending order based on the total revenue.

### TASK-10

```
# What is the maximum revenue out of (10,20,30,40,50) rows?
df.iloc[10:60:10]["Revenue_millions"].max()

936.63
```

**Code Commentary:** By using iloc function it selects every 10th row from index 10 to index 59 in the DataFrame df, and then calculates the maximum value of the "Revenue\_millions" column within this subset.

### TASK-11

```
# How many movies with the genres 'Adventure', 'Action', 'Horror', and 'Crime' exist in the IMDB dataset?
df[df["Genre"].isin(["Adventure", "Action", "Horror", "Crime"])]["Title"].count()
```

Code Commentary: Using Count Function counts the number of movies in the DataFrame df that belong to the genres Adventure, Action, Horror, or Crime.

### **TASK-12**

```
# Create a genre-level report with metrics average rating, the average number of votes, and average revenue.
# What is the average rating of the 'Horror' genre?
df[df["Genre"] == "Horror"]["Rating"].mean()
5.867391304347826
```

Code Commentary: Calculates the mean (average) rating of movies belonging to the Horror genre in the DataFrame df.

#### TASK-13

```
# How many movies has Billy Ray directed and find the year of release of those movies
cond = df[df["Director"] == "Billy Ray"]["Title"].count()
print("The no of movies has Billy Ray directed is",cond)
year = df[df["Director"] == "Billy Ray"]["Year"]
print("The year of movie that billy ray directed is",year)
The no of movies has Billy Ray directed is 1
The year of movie that billy ray directed is 995 2015
Name: Year, dtype: int64
```

**Code Commentary:** This code first counts the number of movies directed by "Billy Ray" and stores it in the variable cond. It then prints this count. Next, it retrieves the years of the movies directed by "Billy Ray" and prints them.

### TASK-14

```
# How many movies were released in the year 2012 - 14. What type of genre were released the most
x = df[df["Year"].isin([2012,2013,2014])]["Title"].count()
print("The no of movies were released in the year 2012 - 14 is",x)
y= df[df["Year"].isin([2012,2013,2014])]["Genre"].min()
print("Type of Genre were released the most is",y)
The no of movies were released in the year 2012 - 14 is 253
Type of Genre were released the most is Action
```

**Code Commentary:** In 1st Line of code counts the number of movies released between 2012 and 2014 and i stored it in variable x, then prints it out. Subsequently, it determines the genre that appears first alphabetically among the movies released in that time frame, storing it in variable y, and prints out the most released genre.

#### TASK-15

```
# Which Movie had the highest vote and what genre it belongs to.
print(df[df["Votes"] == df["Votes"].max()]["Title"])
df[df["Votes"] == df["Votes"].max()]["Genre"]

54    The Dark Knight
Name: Title, dtype: object

54    Action
Name: Genre, dtype: object
```

Code Commentary: This code prints the titles of the movie(s) with the maximum number of votes and retrieves their corresponding genres.

#### TASK-16

**Code Commentary:** First I identifies the directors whose movies grossed the highest revenue and stores the director names in variable x, then prints it out. I got the answer as "j.j.Abrams" And I calculates the total revenue generated by the director "J.J. Abrams" and stores it in variable y, and prints out the total revenue.

#### TASK-17

```
#Create a report to showcase the revenue of each movie, as % revenue concerning the total revenue of the respective genre
#For example if a movie 'ABC' has genre 'Action' and released in 2015, then % revenue will be
#(Revenue of the movie 'ABC' *100)/ (Total revenue of the genre 'Action' in 2015)

#What is the % revenue of the movie 'Split' in its respective genre and year?

df['%_Revenue'] = (df['Revenue_millions'] / df.groupby(['Genre', 'Year'])['Revenue_millions'].transform('sum')) * 100
split_revenue = df[df['Title'] == 'Split']['%_Revenue'].values[0]
print(f'The % revenue of the movie 'Split' in its respective genre and year is : {split_revenue:}')

The % revenue of the movie 'Split' in its respective genre and year is : 29.42041024985622
```

**Code Commentary:** This code snippet calculates the percentage of revenue generated by each movie within its respective genre and year, assigns it to a new column named '%\_Revenue', and then extracts the percentage revenue of the movie 'Split' using its respective genre and year. Finally, it prints out the percentage revenue of the movie 'Split'.