Task 5

Name : Mukesh Kumar mahto

Reg ID: **GO_STP_9639**

Pandas - Data Analysis of IMDB movies data

As we have a basics understanding of the different data structures in pandas, lets explore the fun and interesting 'IMDB-movies-datasets' and get our hands dirty by performing practical data analysis of real data.

Importing Pandas & Numpy

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

Loading the data set of IMDB

```
# import the google drive and save the dataset from kaggle to drive
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Mounted at /content/gdrive
```

```
#connecting the google drive to kaggle through API
import os
os.environ['KAGGLE_CONFIG_DIR'] = "/content/gdrive/My Drive/Kaggle"
```

```
# change to Kaggle directories in the drive
%cd /content/gdrive/My Drive/Kaggle
```

```
/content/gdrive/My Drive/Kaggle
```

```
# download the kaggle datasets
!kaggle datasets download -d PromptCloudHQ/imdb-data
```

```
imdb-data.zip: Skipping, found more recently modified local copy (use --for
```

✓ 0s completed at 6:25 AM

```
!unzip \*.zip  && rm *.zip
```

```
Archive:  imdb-data.zip
replace IMDB-Movie-Data.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename: n
```

```
movies = pd.read_csv("IMDB-Movie-Data.csv")
```

Understand the basic information about the dataset and inspect the dataframe's columns, shapes, variable types etc.

```
## Viewing the columns

movies.columns
```

```
Index(['Rank', 'Title', 'Genre', 'Description', 'Director', 'Actors', 'Year
       'Runtime (Minutes)', 'Rating', 'Votes', 'Revenue (Millions)',
       'Metascore'],
      dtype='object')
```

```
## Inspecting shapes
movies.shape
```

```
(1000, 12)
```

```
## Inspecting Data Types

print(movies.dtypes)
```

```
Rank                    int64
Title                  object
Genre                  object
Description            object
Director               object
Actors                 object
Year                    int64
Runtime (Minutes)       int64
Rating                float64
```

```
movies.size
```

```
12000
```

Basic Statistics about this Dataset

```
movies.describe()
```

	Rank	Year	Runtime (Minutes)	Rating	Votes	(M
count	1000.000000	1000.000000	1000.000000	1000.000000	1.000000e+03	87
mean	500.500000	2012.783000	113.172000	6.723200	1.698083e+05	8
std	288.819436	3.205962	18.810908	0.945429	1.887626e+05	10
min	1.000000	2006.000000	66.000000	1.900000	6.100000e+01	
25%	250.750000	2010.000000	100.000000	6.200000	3.630900e+04	1
50%	500.500000	2014.000000	111.000000	6.800000	1.107990e+05	4
75%	750.250000	2016.000000	123.000000	7.400000	2.399098e+05	11
max	1000.000000	2016.000000	191.000000	9.000000	1.791916e+06	93

```
movies.head(30).mean()
```

```
Rank                     15.500000
Year                   2015.766667
Runtime (Minutes)       116.500000
Rating                    7.020000
Votes                154962.633333
Revenue (Millions)      145.189259
Metascore                63.851852
dtype: float64
```

Data Selection -- Indexing and Slicing the data

996	5.5
997	6.2
998	5.6
999	5.3

1000 rows × 1 columns

```
## Selecting only movies Titles

movie_title = movies[['Title']]
movie_title
```

	Title
0	Guardians of the Galaxy
1	Prometheus
2	Split
3	Sing
4	Suicide Squad
...	...
995	Secret in Their Eyes
996	Hostel: Part II
997	Step Up 2: The Streets
998	Search Party
999	Nine Lives

Groupby Operations

```
genres = movies.groupby('Genre')['Revenue (Millions)'].mean().reset_index()
print(genres)
```

```
                           Genre  Revenue (Millions)
    0                      Action          131.560000
    1           Action,Adventure          223.740000
    2    Action,Adventure,Biography           16.500000
    3       Action,Adventure,Comedy           95.733571
    4        Action,Adventure,Crime           90.620000
    ..                          ...                 ...
    202     Romance,Sci-Fi,Thriller           62.450000
    203                      Sci-Fi           20.760000
    204           Sci-Fi,Thriller           64.510000
    205                    Thriller            0.320000
    206               Thriller,War                 NaN

    [207 rows x 2 columns]
```

Sorting Operations

```
top_rated = movies.sort_values(["Title","Genre","Revenue (Millions)"], ascending
top_rated.head(10)
```

```
Director              False
Actors                False
Year                  False
Runtime (Minutes)     False
Rating                False
Votes                 False
Revenue (Millions)     True
Metascore              True
dtype: bool
```

```
print(movies.isna().sum())
```

```
Rank                    0
Title                   0
Genre                   0
Description             0
Director                0
Actors                  0
Year                    0
Runtime (Minutes)       0
Rating                  0
Votes                   0
Revenue (Millions)    128
Metascore              64
```

	Title	Director	Rating	Metascore	Revenue (Millions)
1	The Dark Knight	Christopher Nolan	9.0	82.0	533.32
2	Inception	Christopher Nolan	8.8	74.0	292.57
3	Kimi no na wa	Makoto Shinkai	8.6	79.0	4.68
4	Interstellar	Christopher Nolan	8.6	74.0	187.99
5	The Intouchables	Olivier Nakache	8.6	57.0	13.18
6	The Lives of Others	Florian Henckel von Donnersmarck	8.5	89.0	11.28
7	Whiplash	Damien Chazelle	8.5	88.0	13.09