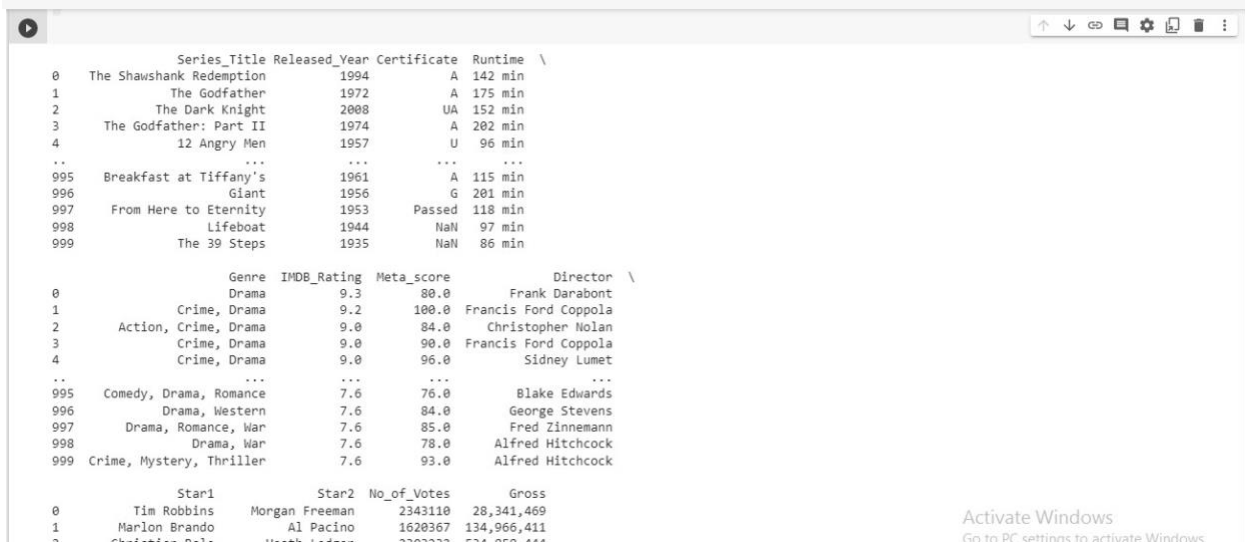


ASSIGNMENT 5

NAME: YASH KATKHADE
ROLL NO: 325
PRN: 202201060046
DIVISION: C2

CODE:

```
import pandas as pd
df=pd.read_csv("/content/imdb data.csv")
print(df)
```



	Series Title	Released Year	Certificate	Runtime	
0	The Shawshank Redemption	1994	A	142 min	
1	The Godfather	1972	A	175 min	
2	The Dark Knight	2008	UA	152 min	
3	The Godfather: Part II	1974	A	202 min	
4	12 Angry Men	1957	U	96 min	
...	
995	Breakfast at Tiffany's	1961	A	115 min	
996	Giant	1956	G	201 min	
997	From Here to Eternity	1953	Passed	118 min	
998	Lifeboat	1944	NaN	97 min	
999	The 39 Steps	1935	NaN	86 min	
	Genre	IMDB_Rating	Meta_score	Director	
0	Drama	9.3	80.0	Frank Darabont	
1	Crime, Drama	9.2	100.0	Francis Ford Coppola	
2	Action, Crime, Drama	9.0	84.0	Christopher Nolan	
3	Crime, Drama	9.0	90.0	Francis Ford Coppola	
4	Crime, Drama	9.0	96.0	Sidney Lumet	
...	
995	Comedy, Drama, Romance	7.6	76.0	Blake Edwards	
996	Drama, Western	7.6	84.0	George Stevens	
997	Drama, Romance, War	7.6	85.0	Fred Zinnemann	
998	Drama, War	7.6	78.0	Alfred Hitchcock	
999	Crime, Mystery, Thriller	7.6	93.0	Alfred Hitchcock	
	Star1	Star2	No_of_Votes	Gross	
0	Tim Robbins	Morgan Freeman	2343110	28,341,469	
1	Marlon Brando	Al Pacino	1620367	134,966,411	
2	Christian Bale	Heath Ledger	7202727	28,050,488	

● PIE CHART:

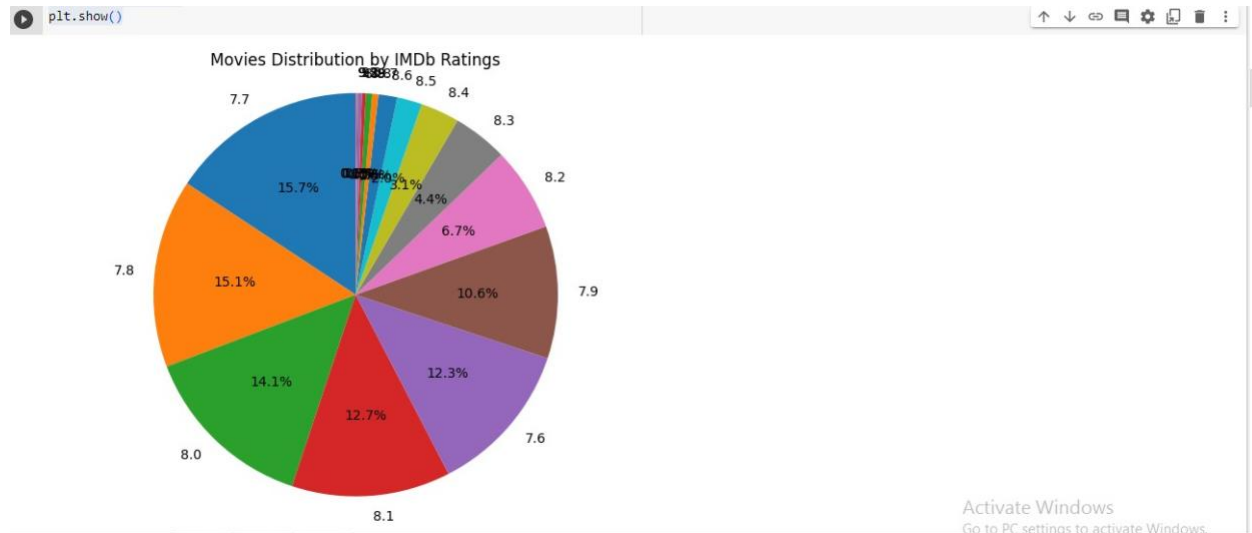
```
import matplotlib.pyplot as plt

# Calculate the count of movies for each IMDb rating
rating_counts = df['IMDB_Rating'].value_counts()
```

```

# Plotting the pie chart
plt.figure(figsize=(8, 6))
plt.pie(rating_counts, labels=rating_counts.index, autopct='%1.1f%%',
startangle=90)
plt.title('Movies Distribution by IMDb Ratings')
plt.axis('equal')
plt.show()

```



- **BAR GRAPH:**

```

import pandas as pd
import matplotlib.pyplot as plt

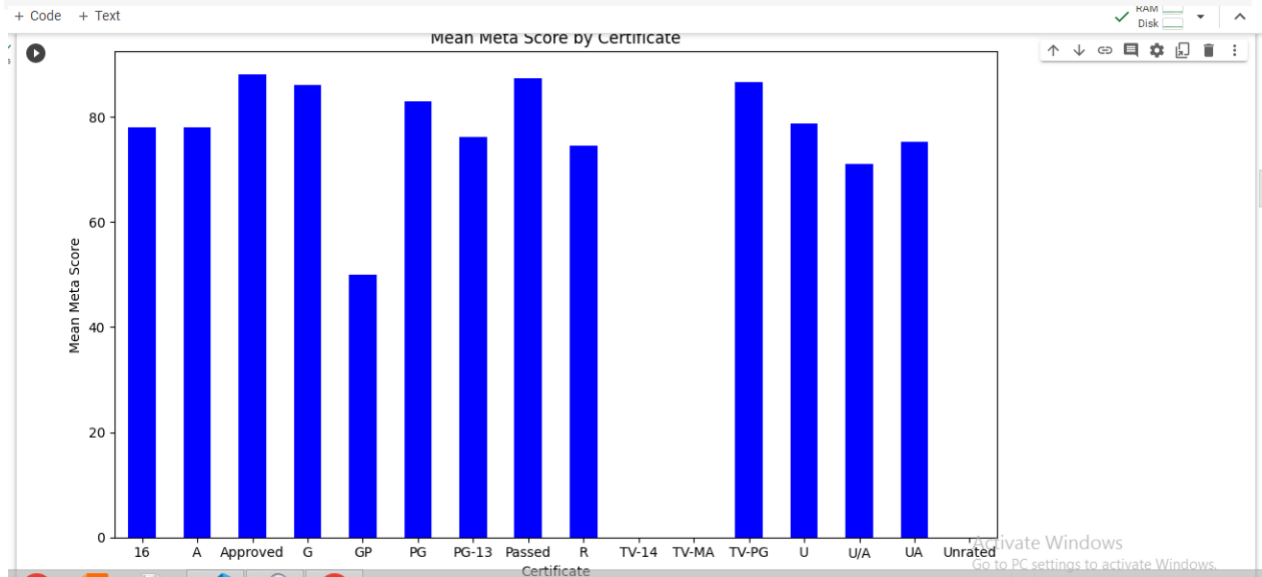
# Group the data by certificate and calculate the mean Meta Score for each
certificate
certificate_scores = df.groupby('Certificate')['Meta_score'].mean()

# Create the bar graph using Matplotlib
plt.figure(figsize=(10, 6))
certificate_scores.plot(kind='bar', color='blue')

# Set plot labels and title
plt.xlabel('Certificate')
plt.ylabel('Mean Meta Score')
plt.title('Mean Meta Score by Certificate')

```

```
# Display the plot
plt.xticks(rotation=0) # Rotate x-axis labels if needed
plt.tight_layout() # Adjust plot layout
plt.show()
```



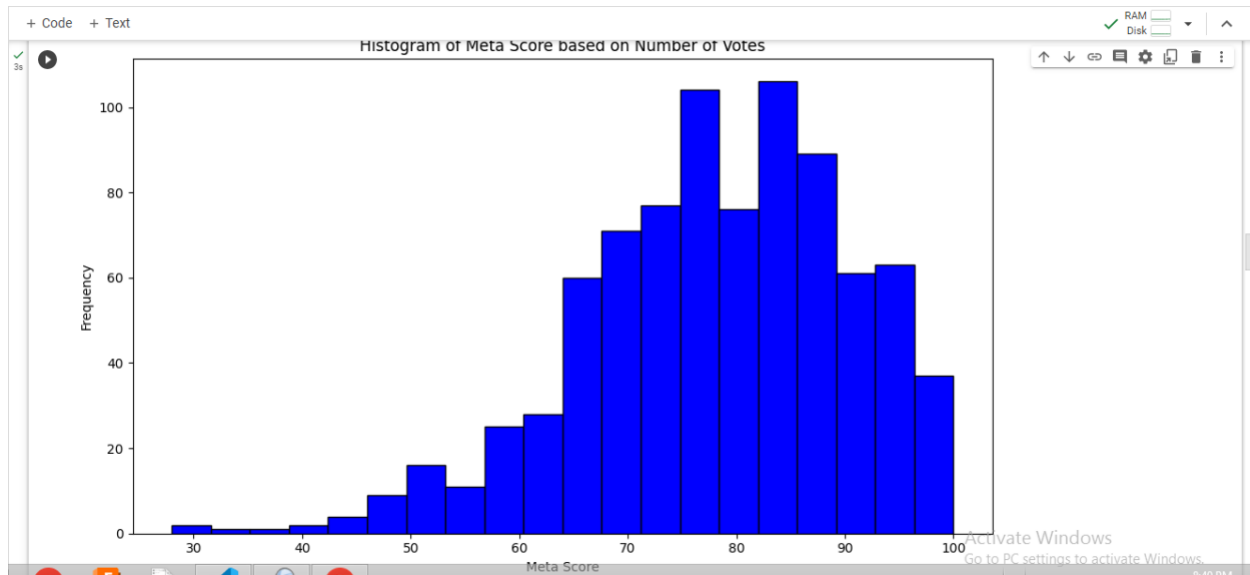
- **HISTOGRAM:**

```
import pandas as pd
import matplotlib.pyplot as plt

# Create the histogram using Matplotlib
plt.figure(figsize=(10, 6))
plt.hist(df['Meta_score'], bins=20, color='blue', edgecolor='black')

# Set plot labels and title
plt.xlabel('Meta Score')
plt.ylabel('Frequency')
plt.title('Histogram of Meta Score based on Number of Votes')

# Display the plot
plt.tight_layout() # Adjust plot layout
plt.show()
```



- **LINE CHART:**

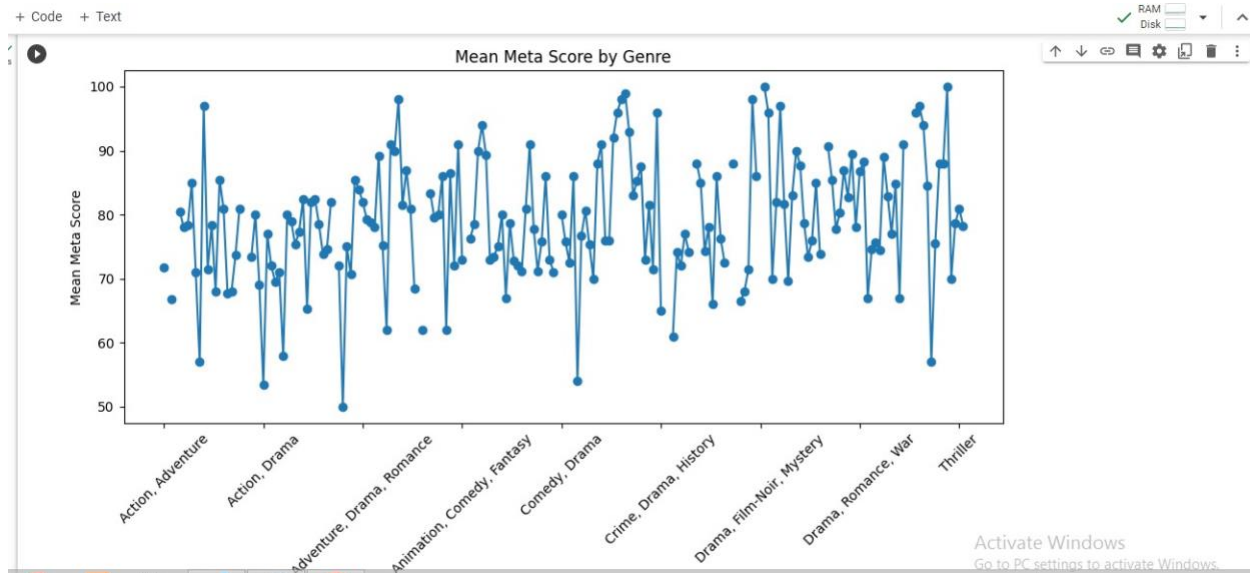
```
import pandas as pd
import matplotlib.pyplot as plt

# Group the data by genre and calculate the mean Meta Score for each genre
genre_scores = df.groupby('Genre')['Meta_score'].mean()

# Create the line chart using Matplotlib
plt.figure(figsize=(10, 6))
genre_scores.plot(kind='line', marker='o')

# Set plot labels and title
plt.xlabel('Genre')
plt.ylabel('Mean Meta Score')
plt.title('Mean Meta Score by Genre')

# Display the plot
plt.xticks(rotation=45) # Rotate x-axis labels for better visibility
plt.tight_layout() # Adjust plot layout
plt.show()
```



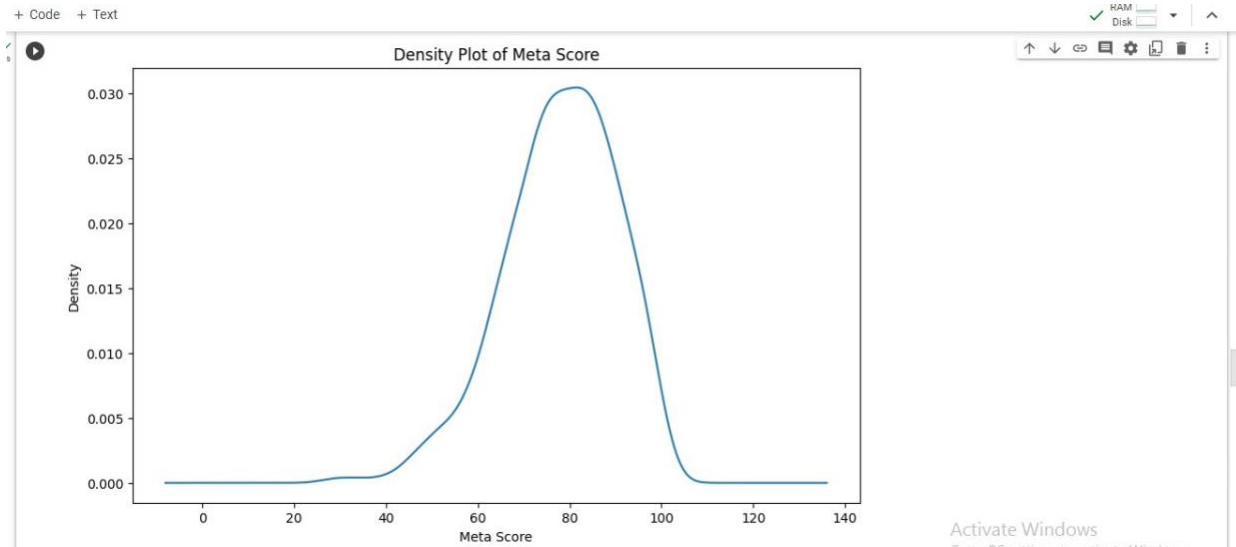
- **DENSITY PLOT:**

```
import pandas as pd
import matplotlib.pyplot as plt

# Create the density plot using Matplotlib
plt.figure(figsize=(10, 6))
df['Meta_score'].plot(kind='density')

# Set plot labels and title
plt.xlabel('Meta Score')
plt.ylabel('Density')
plt.title('Density Plot of Meta Score')

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



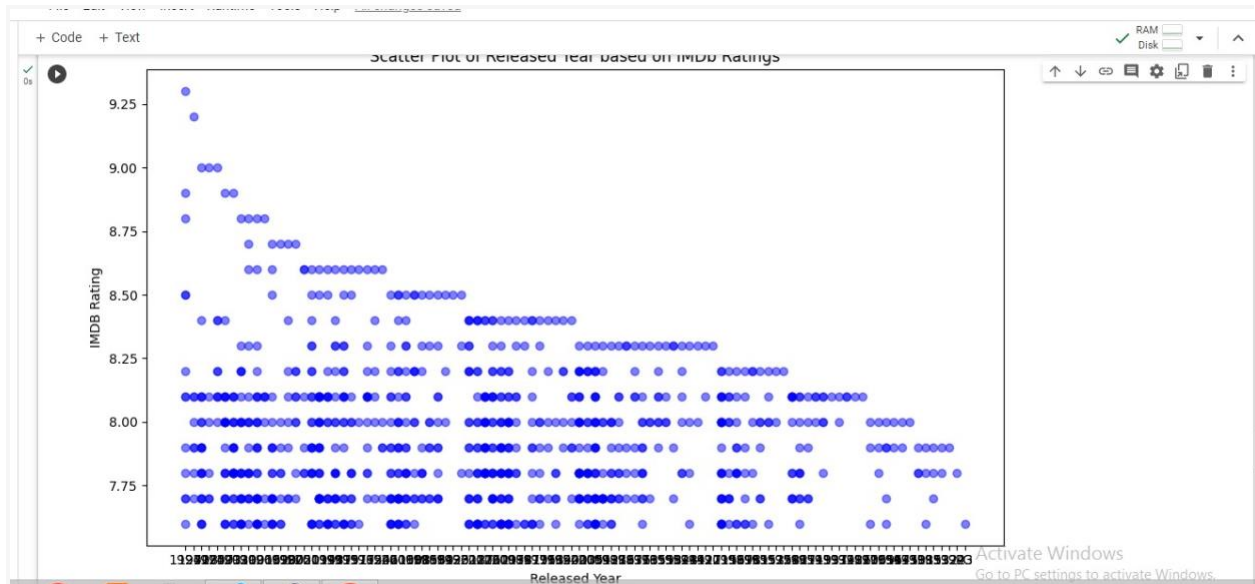
- **SCATTER PLOT:**

```
import pandas as pd
import matplotlib.pyplot as plt

# Create the scatter plot using Matplotlib
plt.figure(figsize=(10, 6))
plt.scatter(df['Released_Year'], df['IMDB_Rating'], color='blue',
alpha=0.5)

# Set plot labels and title
plt.xlabel('Released Year')
plt.ylabel('IMDB Rating')
plt.title('Scatter Plot of Released Year based on IMDb Ratings')

# Display the plot
plt.tight_layout() # Adjust plot layout
plt.show()
```



- FACET GRID:

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

# Create the facet grid using Seaborn
g = sns.FacetGrid(df, col='IMDB_Rating', col_wrap=4, height=4, aspect=1.2)
g.map(sns.scatterplot, 'IMDB_Rating', 'Meta_score', color='blue',
alpha=0.5)

# Set plot labels and title
g.set_axis_labels('IMDB Rating', 'Meta Score')
g.fig.suptitle('Facet Grid of Meta Score based on IMDb Ratings')

# Adjust plot layout
plt.tight_layout()
plt.subplots_adjust(top=0.9)

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

