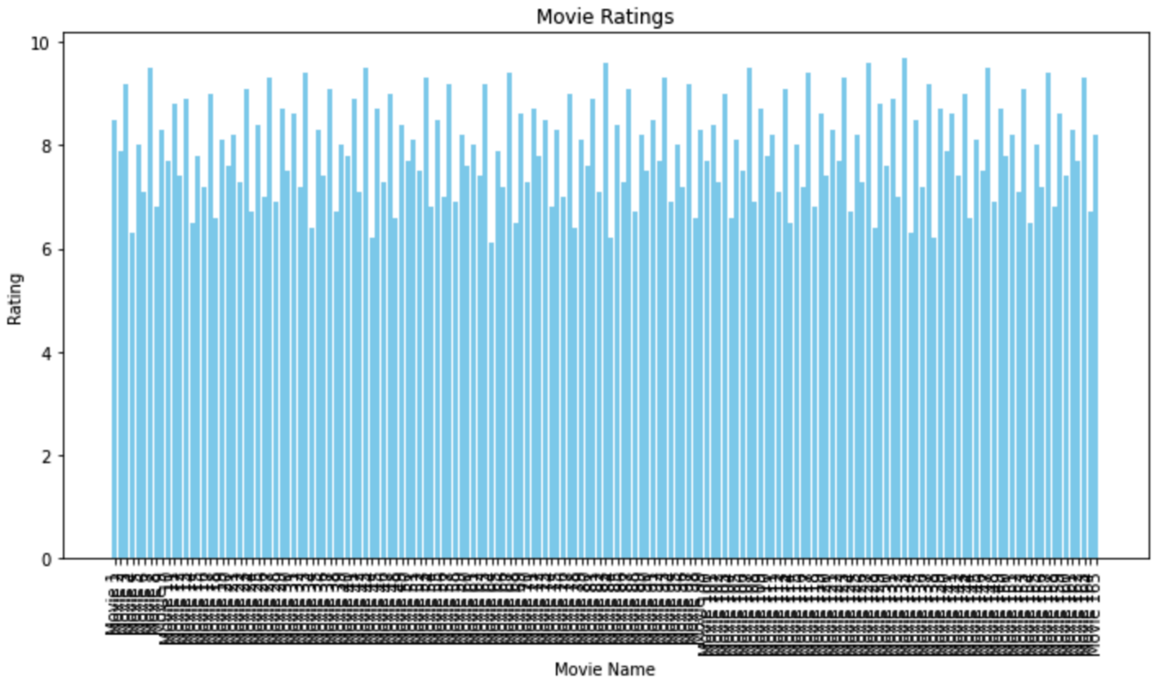
DATA VISUALIZATION USING PYTHON

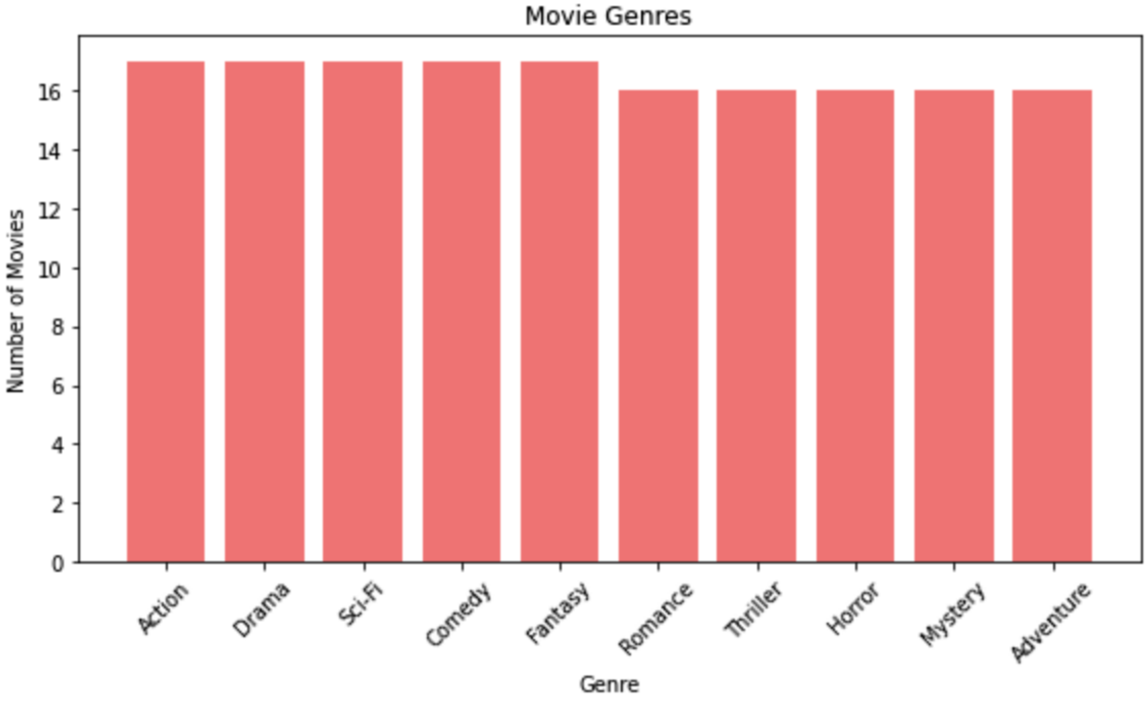
Aim: To visually explore and present insights from a movie dataset, highlighting movie ratings, genres, and their distribution.

CODE:

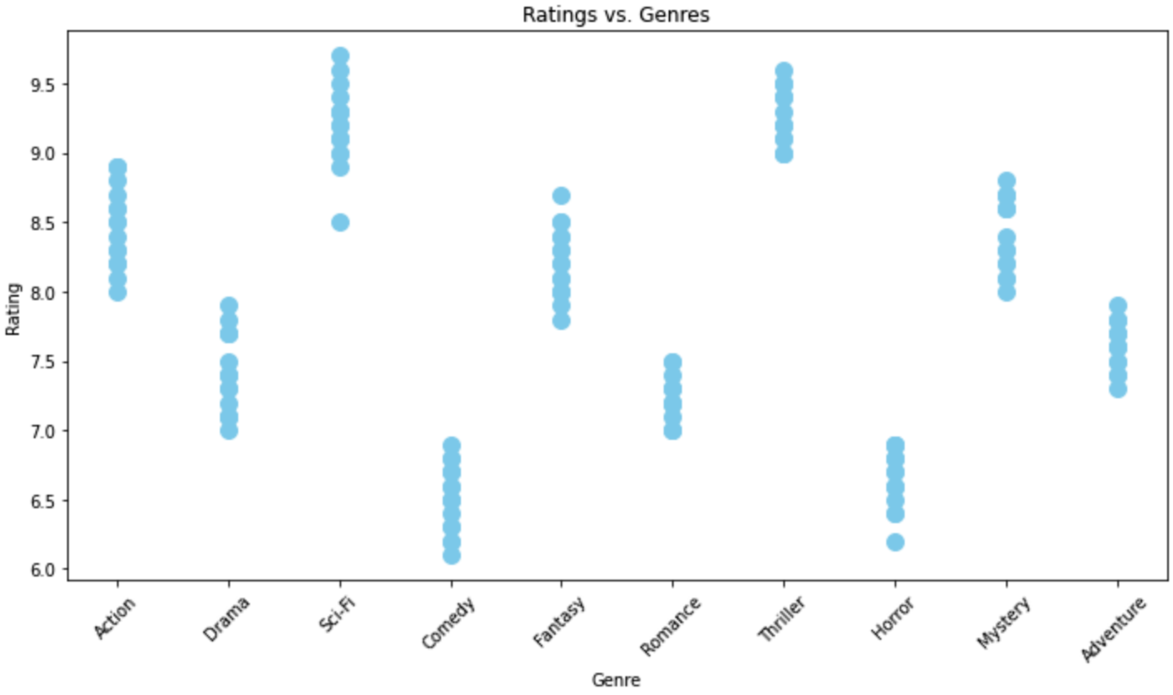
```
import pandas as pd
import matplotlib.pyplot as plt

# Load the CSV data into a Pandas DataFrame
data = pd.read_csv('/Users/dhawalsahu/Downloads/Movie_review.csv')

# Extract data for plotting
movie_names = data['movie_name']
genres = data['genre']
ratings = data['star_ratings']
plt.figure(figsize=(10, 6))
plt.bar(movie_names, ratings, color='skyblue')
plt.xlabel('Movie Name')
plt.ylabel('Rating')
plt.title('Movie Ratings')
plt.xticks(rotation=90)
plt.tight_layout()
```



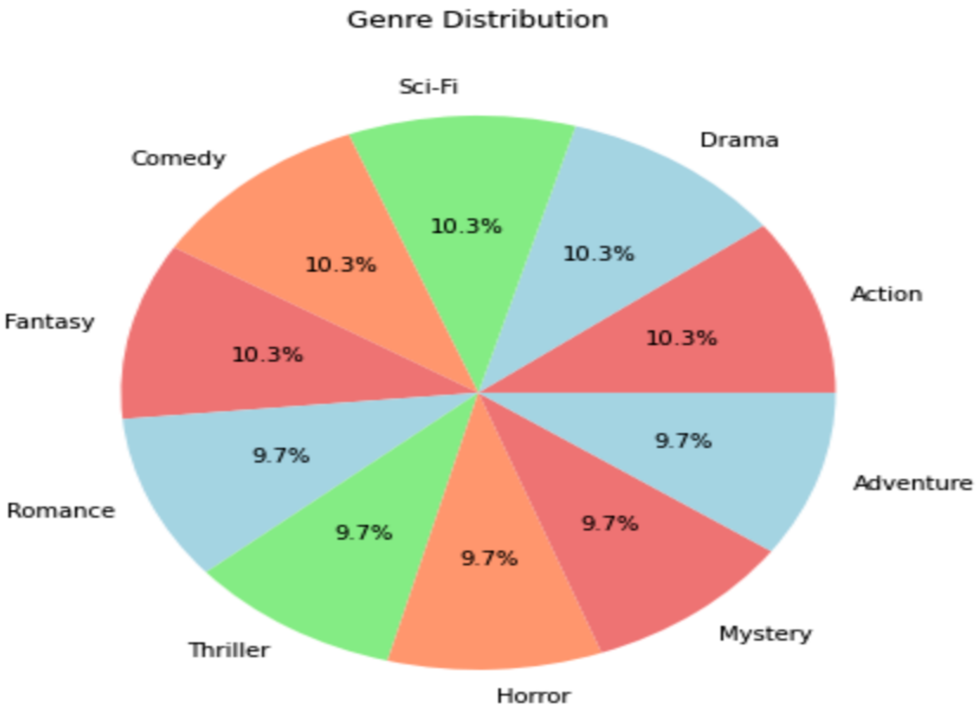
```
# Create a bar chart for movie genres
genre_counts = genres.value_counts()
plt.figure(figsize=(8, 5))
plt.bar(genre_counts.index, genre_counts.values, color='lightcoral')
plt.xlabel('Genre')
plt.ylabel('Number of Movies')
plt.title('Movie Genres')
plt.xticks(rotation=45)
plt.tight_layout()
```



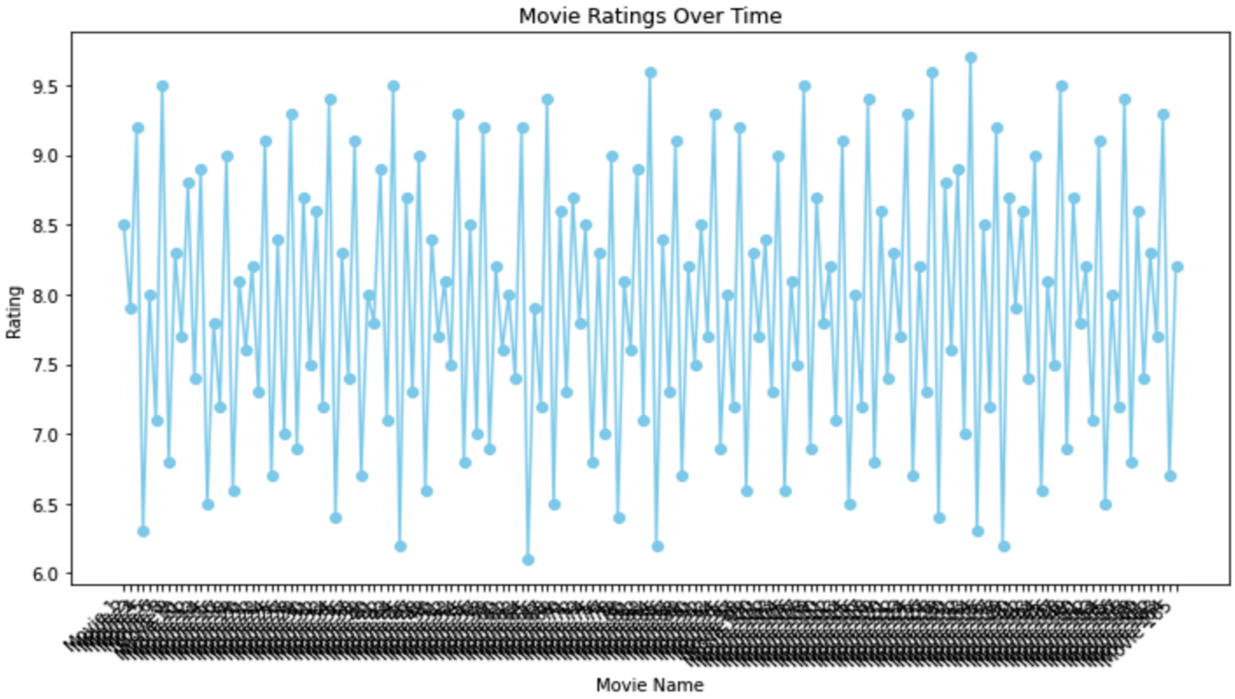
```
# Create a scatter plot for Ratings vs. Genres
plt.figure(figsize=(10, 6))
plt.scatter(data['genre'], data['star_ratings'], c='skyblue', marker='o', s=100)
plt.xlabel('Genre')
plt.ylabel('Rating')
plt.title('Ratings vs. Genres')
plt.xticks(rotation=45)
plt.tight_layout()
```



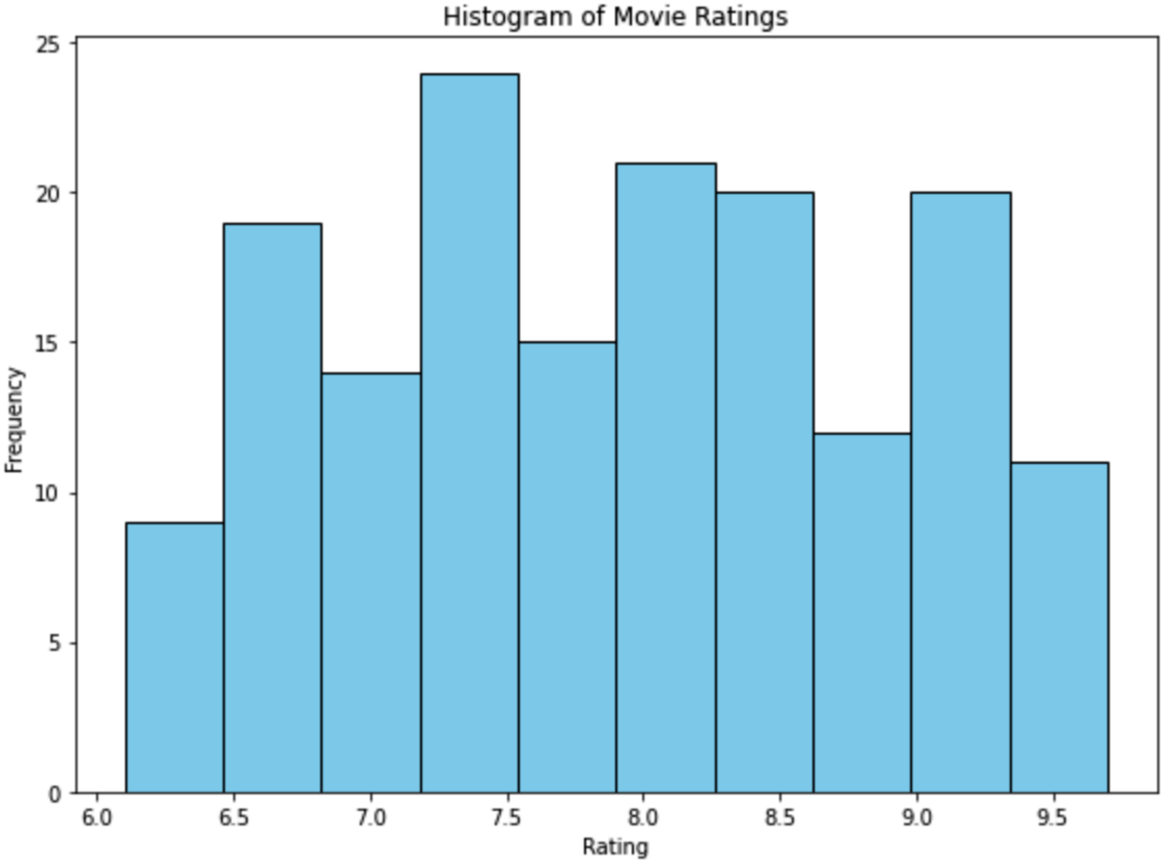
```
# Create a pie chart for genre distribution
genre_counts = data['genre'].value_counts()
plt.figure(figsize=(6, 6))
plt.pie(genre_counts, labels=genre_counts.index, autopct='%1.1f%%', colors=['lightcoral',
'lightblue', 'lightgreen', 'lightsalmon'])
plt.title('Genre Distribution')
plt.axis('equal')  # Equal aspect ratio ensures that the pie chart is a circle.
plt.tight_layout()

# Show the pie chart
plt.show()
```



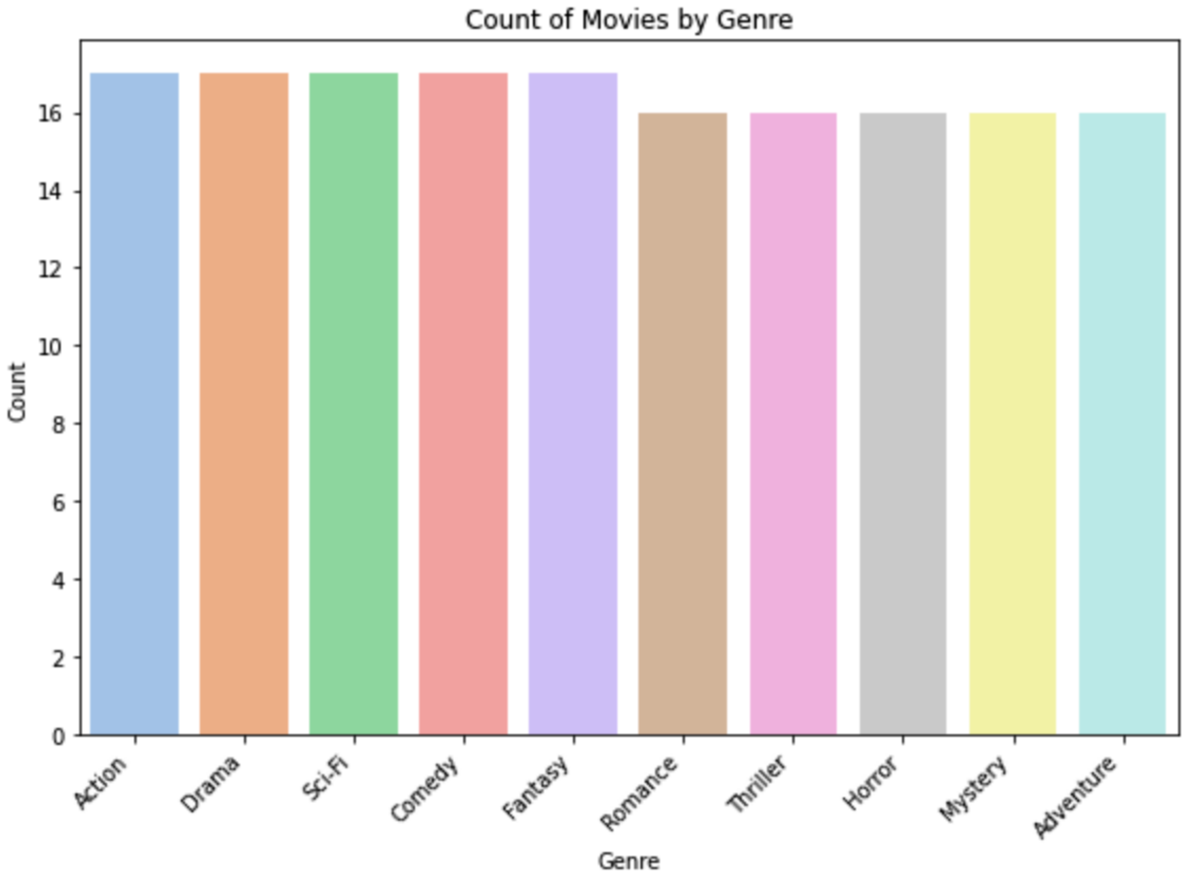
```
plt.figure(figsize=(10, 6))
plt.plot(data['movie_name'], data['star_ratings'], marker='o', color='skyblue')
plt.xlabel('Movie Name')
plt.ylabel('Rating')
plt.title('Movie Ratings Over Time')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()

# Show movie names on the X-axis
plt.show()
```



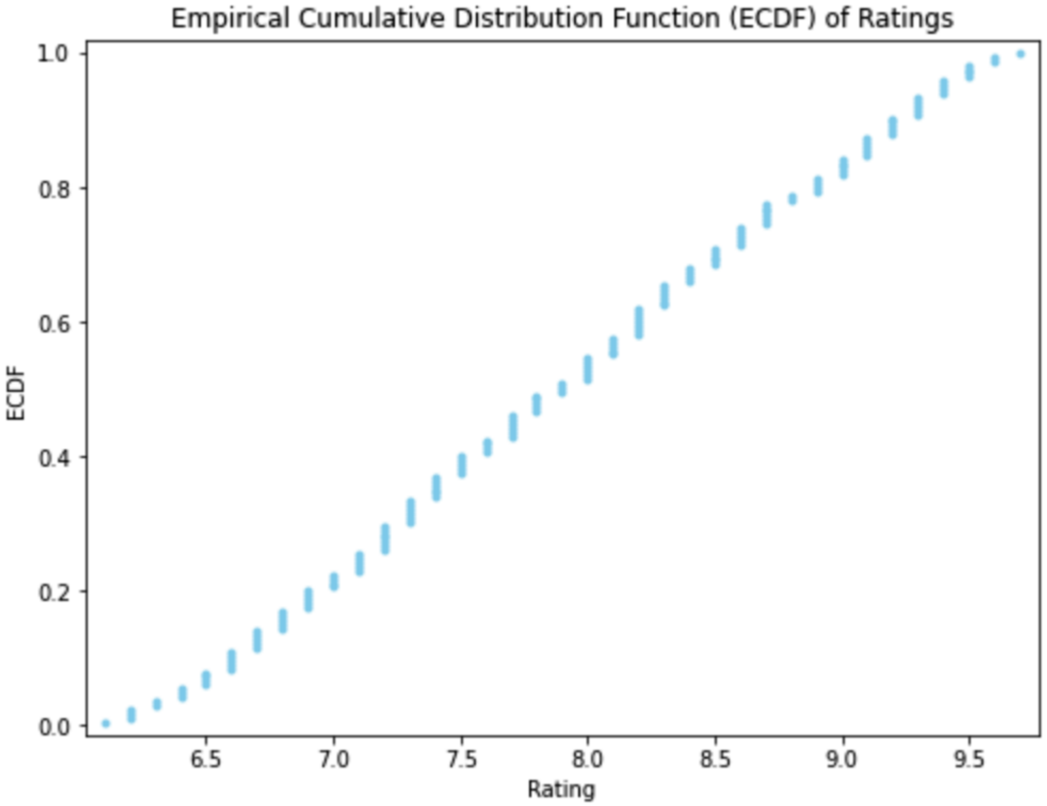
```
# Create a histogram for movie ratings
plt.figure(figsize=(8, 6))
plt.hist(data['star_ratings'], bins=10, color='skyblue', edgecolor='black')
plt.xlabel('Rating')
plt.ylabel('Frequency')
plt.title('Histogram of Movie Ratings')
plt.tight_layout()

# Show the histogram
plt.show()
```



```
plt.figure(figsize=(8, 6))
sns.countplot(x='genre', data=data, palette='pastel')
plt.xlabel('Genre')
plt.ylabel('Count')
plt.title('Count of Movies by Genre')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



```
import numpy as np
# Extract the 'Rating' column
ratings = data['star_ratings']

# Sort the ratings in ascending order
ratings_sorted = np.sort(ratings)

# Calculate the ECDF
n = len(ratings_sorted)
y = np.arange(1, n + 1) / n

# Create the ECDF plot
plt.figure(figsize=(8, 6))
plt.plot(ratings_sorted, y, marker='.', linestyle='none', color='skyblue')
plt.xlabel('Rating')
plt.ylabel('ECDF')
plt.title('Empirical Cumulative Distribution Function (ECDF) of Ratings')
plt.margins(0.02)  # Add a small margin to the plot
plt.show()
```



RESULT-Hence, the movie rating was analysed and successfully visualised using different tools in python