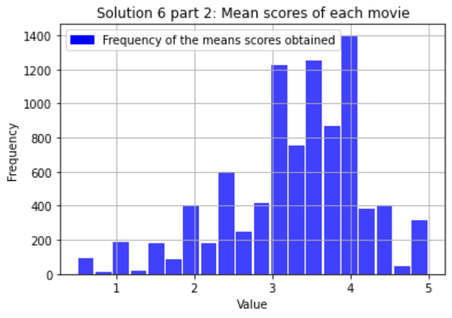
In [76]:
```
"""Download the file called "movie-lens-ratings.csv" (the data is also available at http://grouplens.org/datasets/). The data comes from a movie review website. Each row is a review, and the columns are the id of the author of the review, the id of the movie being r

Answer the following questions:

1. How many reviews are there with missing ratings?

2. How many unique reviewers are there? How many unique movies?

3. How many reviews did each reviewer write on average?

4. How many reviews did each movie receive on average?

5. How many reviews are there with the score of 5? With the score of 1?

6. What is the distribution of mean scores of the movies? First build a dataframe containing mean scores for each movie, then create a histogram from this dataframe (df.hist) with 20 bins.

Click on the title of this section to make your submission


"""

import matplotlib.pyplot as plt   #importing pyplot module of matplotlib package
import matplotlib.patches as mpatches #importing patches module of matplotlib package
import pandas as pd      #importing pandas package


df = pd.read_csv (r'movie-lens-ratings.csv')  #reading data from csv file and storing in df


df.set_index("timestamp",inplace=True)  #setting index to timestamp to make the dataframe unique and inpace=True to make changes in the same dataframe


print("Solution 1: Number of reviews with missing ratings is ", df["rating"].isnull().sum())   #solution 1 - adding how many times the rating value is null



total_unique_users= len(df["userId"].unique())  #finding total of unique users identified by userId in the dataframe
print("Solution 2 part 1: Number of unique reviewers is ", total_unique_users) #solution 2 Part 1



total_unique_movies= len(df["movieId"].unique())  #finding total of unique movies identified by movieId in the dataframe
print("Solution 2 part 2: Number of unique movies is ", total_unique_movies) #solution 2 Part 2


total_unique_user_ratings = df.groupby(["userId"]).rating.count().sum() #grouping the rating according to userId, then getting the count of rating for each userId and adding(sum) the number of count of rating
print("Solution 3: The number of reviews that each reviewer write on an average is ",round((total_user_ratings/total_unique_users),2)) #solution 3


total_unique_movie_ratings= df.groupby(["movieId"]).rating.count().sum() #grouping the rating according to movieId, then getting the count of rating for each movieId and adding(sum) the number of count of rating
print("Solution 4: The number of reviews did each movie receive on average is ",round((total_unique_movie_ratings/total_unique_movies),2))   #solution 4



print("Solution 5 part 1: Number of reviews with the score of 5 is ", len(df[df.rating==5]))  #solution 5 part 1 finding the count  of all the reviews whose rating is 5
print("Solution 5 part 2: Number of reviews with the score of 1 is ", len(df[df.rating==1]))  #solution 5 part 2 finding the count  of all the reviews whose rating is 1



mean_rating = df.groupby(["movieId"]).rating.mean()  #grouping the rating according to movieId and finding the mean
print("Solution 6 part1 : Distribution of mean scores of the movies ", mean_rating)                          #solution 6 part 1



blue_patch = mpatches.Patch(color='blue', label='Frequency of the means scores obtained') #creating patch and labelling it
plt.legend(handles=[blue_patch])
plt.hist(x=mean_rating, bins=20,alpha=0.75, color='blue',
                           rwidth=0.90)  #putting various arguments in histogram where bins=20 is bins size, alpha is the transparency value, rwidth is width of bars in histogram
plt.xlabel('Value')    #Labelling the x-axis of histogram
plt.ylabel('Frequency') #Labelling the y-axis of histogram
plt.grid(True)   #adding grid in histogram
plt.title('Solution 6 part 2: Mean scores of each movie') #adding title of histogram
plt.show()   #solution 6 part 2
```

```
Solution 1: Number of reviews with missing ratings is  0
Solution 2 part 1: Number of unique reviewers is  671
Solution 2 part 2: Number of unique movies is  9066
Solution 3: The number of reviews that each reviewer write on an average is  149.04
Solution 4: The number of reviews did each movie receive on average is  11.03
Solution 5 part 1: Number of reviews with the score of 5 is  15095
Solution 5 part 2: Number of reviews with the score of 1 is  3326
Solution 6 part1 : Distribution of mean scores of the movies  movieId
1         3.872470
```

```
2         3.401869
3         3.161017
4         2.384615
5         3.267857
            ...
161944    5.000000
162376    4.500000
162542    5.000000
162672    3.000000
163949    5.000000
Name: rating, Length: 9066, dtype: float64
```



In [ ]:

In [ ]:

In [ ]:

In [ ]: