

Python Summer Party Challenge

by Interview Master

Day 13 of 15

Shake Shack

You are a Product Analyst working with the Shake Shack R&D team to evaluate customer ratings for experimental milkshake flavors. Your team has collected ratings data from a small sampling test. Your task is to systematically analyze and clean the ratings data to identify top-performing flavors.

Challenge Questions

Q1:

There was an error in our data collection process, and we unknowingly introduced duplicate rows into our data. Remove any duplicate entries in the customer ratings data to ensure the accuracy of the analysis.

Q2:

For each milkshake flavor, calculate the average customer rating and append this as a new column to the milkshake_ratings DataFrame. Don't forget to clean the DataFrame first by dropping duplicate values.

Q3:

For each row in dataset, calculate the difference between that customer's rating and the average rating for the flavor. Don't forget to clean the DataFrame first by dropping duplicate values.



Want to try this yourself?

Join the Challenge

Sign up for the Python Summer Party Challenge and solve 21 days
of data science problems

www.interviewmaster.ai/python-party

Or keep scrolling to see my solutions

My Solution - Q1

Day 13 Python Challenge

```
# Removing duplicate rows
without_duplicates = milkshake_ratings.drop_duplicates().reset
_index(drop=True)
print(without_duplicates)
```



My Solution - Q2

Day 13 Python Challenge

```
# Removing duplicate rows
without_duplicates = milkshake_ratings.drop_duplicates().reset_index(drop=True)

# Calculating average customer rating per flavor
avg_ratings = without_duplicates.groupby('flavor')['rating'].mean()

without_duplicates['avg_rating'] = without_duplicates['flavor'].map(avg_ratings)
print(without_duplicates)
```



My Solution - Q3

Day 13 Python Challenge

```
# Removing duplicate rows
without_duplicates = milkshake_ratings.drop_duplicates().reset_index(drop=True)

# Calculating average customer rating per flavor
avg_ratings = without_duplicates.groupby('flavor')['rating'].mean()

without_duplicates['avg_rating'] = without_duplicates['flavor'].map(avg_ratings)

# Calculating the difference between the customer's rating and the flavor rating
without_duplicates['rating_diff'] = without_duplicates['rating'] - without_duplicates['avg_rating']
print(without_duplicates)
```

Ready for your own challenge?

Try this yourself by signing up for the Python Summer Party challenge:

www.interviewmaster.ai/python-party



InterviewMaster.AI