

Python Summer Party Challenge

by Interview Master

Day 8 of 15

Lululemon

You are a Product Analyst for the Lululemon Online Store team investigating how alternative payment methods might influence sales performance. The team wants to understand the potential impact of introducing a new installment payment option. Your analysis will predict sales lift and customer conversion for the proposed payment method.

Challenge Questions

Q1:

Between April 1st and June 30th, 2025, what is the count of transactions for each payment method? This analysis will establish the baseline distribution of how customers currently pay.

Q2:

Between April 1st and June 30th, 2025, what is the average order value for each payment method? This metric will help us assess which payment methods are tied to higher spending levels.

Q3:

Between April 1st and June 30th, 2025, what would be the predicted sales lift if a 'pay over time' option were introduced? Assume that 20% of credit card transactions during this period would switch to using the 'pay over time' option. And that for these switched transactions, the order value is expected to increase by 15% based on the average order value of all credit card transactions in that same time period.



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 8 Python Challenge

```
# Get April 1 - June 30, 2025 transactions
april_june = fct_transactions[
    (fct_transactions['transaction_date'].dt.year == 2025) &
    (fct_transactions['transaction_date'].dt.month.isin([4,5,
6]))]
]

# Count transactions per payment payment_method
transaction_count = april_june.groupby('payment_method')['transaction_id'].count()
print(transaction_count)
```



My Solution - Q2

Day 8 Python Challenge

```
# Get April 1 - June 30, 2025 transactions
april_june = fct_transactions[
    (fct_transactions['transaction_date'].dt.year == 2025) &
    (fct_transactions['transaction_date'].dt.month.isin([4,5,
6]))]
]

# Get average order value per payment method
avg_order_value = april_june.groupby('payment_method')['order_
value'].mean()
print(avg_order_value)
```



My Solution - Q3

Day 8 Python Challenge

```
# Get April 1 - June 30, 2025 transactions
april_june = fct_transactions[
    (fct_transactions['transaction_date'].dt.year == 2025) &
    (fct_transactions['transaction_date'].dt.month.isin([4,5,
6]))]
]

# Get credit card transactions
credit_card_trans = april_june[april_june['payment_method'] == "credit_card"]

# Calculate the number of switched transactions
credit_card_trans_count = len(credit_card_trans)
pay_over_time = (20 / 100) * credit_card_trans_count

# Calculate the average order value for credit card transactions
avg_order_value_credit = credit_card_trans['order_value'].mean()

# Calculate the expected increase in order value for switched transactions
expected_increase = (15 / 100) * avg_order_value_credit

# Calculate the predicted sales lift
predicted_sales_lift = pay_over_time * expected_increase
print(f"Predicted sales lift: {predicted_sales_lift:.2f}")
```

Ready for your own challenge?

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

www.interviewmaster.ai/python-party

