

# Python Summer Party Challenge

by Interview Master

Day 11 of 15

## Stripe

You are a data analyst in Stripe's risk management team investigating transaction patterns to identify potential fraud. The team needs to develop a systematic approach to screen transactions for financial risks. Your goal is to create an initial risk assessment methodology using transaction characteristics.

### Challenge Questions

**Q1:**

How many transactions in October 2024 have a customer email ending with a domain other than 'gmail.com', 'yahoo.com', or 'hotmail.com'? This metric will help us identify transactions associated with less common email providers that may indicate emerging risk patterns.

**Q2:**

For transactions occurring in November 2024, what is the average transaction amount, using 0 as a default for any missing values? This calculation will help us detect abnormal transaction amounts that could be related to fraudulent activity.

**Q3:**

Among transactions flagged as 'High' risk in December 2024, which day of the week recorded the highest number of such transactions? This analysis is intended to pinpoint specific days with concentrated high-risk activity and support the development of our preliminary fraud detection score.



# 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](http://www.interviewmaster.ai/python-party)

Or keep scrolling to see my solutions

# My Solution - Q1

Day 11 Python Challenge

```
# Get transactions in October 2024
fct_transactions['transaction_date'] = pd.to_datetime(fct_transactions['transaction_date'])
october_24_trans = fct_transactions[
    (fct_transactions['transaction_date'].dt.year == 2024) &
    (fct_transactions['transaction_date'].dt.month == 10)
]

# Get transactions with customer emails other than 'gmail.com', 'yahoo.com', or 'hotmail.com'
common_domains = ("gmail.com", "yahoo.com", "hotmail.com")

less_common_emails_trans = october_24_trans[
    ~october_24_trans['customer_email'].str.endswith(common_domains, na=False)
]

print(f"Number of transactions with a less common email providers: {len(less_common_emails_trans)}")
```



# My Solution - Q2

Day 11 Python Challenge

```
# Get November 2024 transactions
nov_24_trans = fct_transactions[
    (fct_transactions['transaction_date'].dt.month == 11) &
    (fct_transactions['transaction_date'].dt.year == 2024)
]

# Replace missing amounts with 0 and calculate average
avg_transaction_amount = nov_24_trans['transaction_amount'].fillna(0).mean()

print(f"Average transaction amount for November 2024: {avg_transaction_amount}")
```



# My Solution - Q3

Day 11 Python Challenge

```
# Get December 2024 high-risk transactions
dec_high_risk = fct_transactions.merge(
    dim_risk_flags,
    on='transaction_id'
).query("risk_level == 'High'")

dec_high_risk = dec_high_risk[
    (dec_high_risk['transaction_date'].dt.month == 12) &
    (dec_high_risk['transaction_date'].dt.year == 2024)
]

# Extract the day of the week and count transactions per day of the week
dec_high_risk["day_of_week"] = dec_high_risk['transaction_date'].dt.day_name()

dow_trans_count = dec_high_risk.groupby('day_of_week')['transaction_id'].count()

print(f"Day of the week with the highest High risk flagged transactions is {dow_trans_count.idxmax()} with {dow_trans_count.max()} transactions.")
```

Ready for your own challenge?

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

[www.interviewmaster.ai/python-party](http://www.interviewmaster.ai/python-party)



InterviewMaster.AI