



N26 KYC Challenge

Task 1

As a financial institution regulated by the FCA, N26 has the obligation to verify the identity of all customers who want to open a N26 account. Each prospective customer has to go through a Know Your Customer (KYC) process by submitting a government-issued photo ID and a facial picture of themselves to our partner, Veritas. Veritas then would perform 2 checks:

- Document check: To verify that the photo ID is valid and authentic;
- Facial Similarity check: To verify that the face in the picture is the same with that on the submitted ID.

The customer will 'pass' the KYC process and get aboard if the results of both Document and Facial Similarity checks are 'clear'. If the result of any check is not 'clear', the customer has to submit all the photos again.

The pass rate is defined as the number of customers who pass both the KYC process divided by the number of customers who attempt the process. Each customer has up to 2 attempts.

The pass rate has decreased substantially for the last few weeks. Please write a report that outlines the root causes and solutions.

Relevant files:

- [facial_similarity_reports.csv](#) - Reports of all Facial Similarity checks
- [doc_reports.csv](#) - Reports of all Document checks
- [veritas.html](#) - The API documentation of Veritas explaining some terms used in the reports.

The candidate is free to use SQL or any scripting language to parse and analyse the data. Please show all your work (including your code if applicable) and assumptions.

Bonus Task

You've asked and they have delivered. The data team, that is. You've noticed something odd happening over the last few days and have requested information to investigate further. You're not sure what you're looking for but you're sure that once you get your hands on that data, you'll be able to figure it out in no time!

In the [folder](#), the data team have provided six files and a message for you...

1. Communication and SQL familiarity:
 - a. Examine the following SQL query, and explain clearly and succinctly what it means. Will the query work? Explain why or why not.

```
WITH processed_users AS (  
    SELECT left(u.phone_country, 2) AS short_phone_country, u.id  
    FROM users u  
)  
SELECT t.user_id,  
t.merchant_country,  
sum(t.amount / fx.rate / power(10, cd.exponent)) AS amount  
FROM transactions t  
JOIN fx_rates fx ON (fx.ccy = t.currency AND fx.base_ccy = 'EUR')  
JOIN currency_details cd ON cd.currency = t.currency  
JOIN processed_users pu ON pu.id = t.user_id  
WHERE t.source = 'GAIA'  
AND pu.short_phone_country = t.merchant_country  
GROUP BY t.user_id, t.merchant_country  
ORDER BY amount DESC;
```

- b. Now it's your turn! Write a query to identify users whose first transaction was a successful card payment over \$10 USD

Good luck!