**Michael Cabas**

Data Analyst | Data Scientist | Consultant | Python | SQL | PowerBI

Medellín, COL

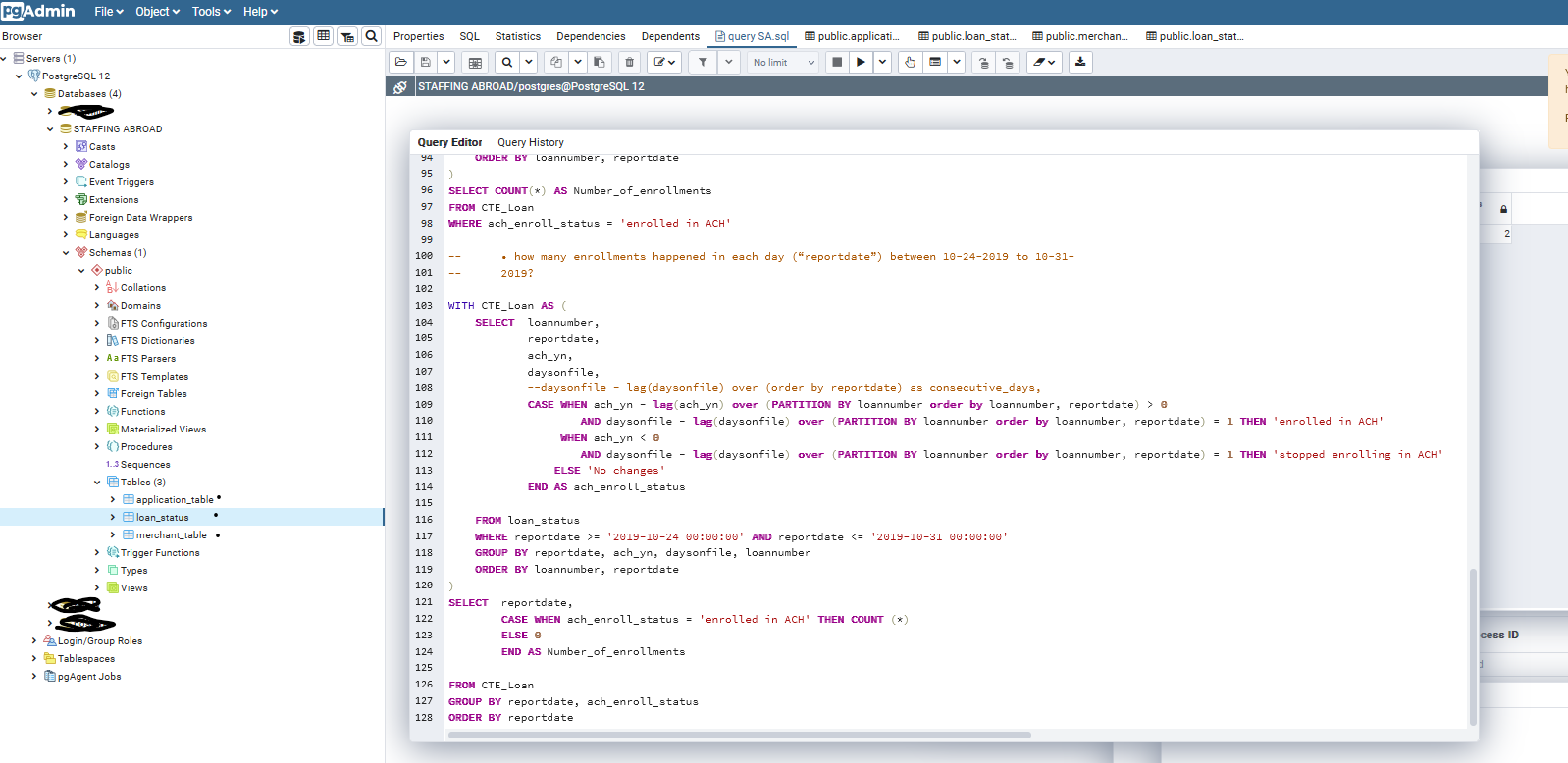
+57 3013748109

mcabas.ingbio@gmail.com

[www.linkedin.com/in/michael-cabas](http://www.linkedin.com/in/michael-cabas)

<https://github.com/mcabas>

*-- Test: STAFFING ABROAD | SQL Data Analyst process technical questions*



Img 1. The exercise was performed on pgAdmind 4 using PostgreSQL

*-- Q1*

*-- 1. Create tables*

*--create table application table and merchant table*

CREATE TABLE application\_table

(applicationid integer,

merchantid integer,

applicationdate timestamp

);

CREATE TABLE merchant\_table

(merchantid integer,

businessname varchar

);

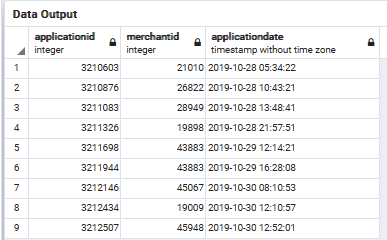
*-- 2. Import data into the tables*

*--Step 1. Refresh Database*

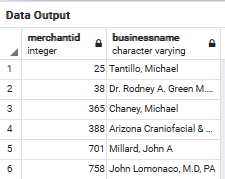
*--Step 2. Go to the application\_table table and click to open import/export, in options select import, and look for the application\_table.csv file in downloads, update the header as ‘yes’ and delimiter as ‘,’, then go to the columns tab and validate that all columns are selected.*

*--Step 3. Click on OK.*

*--Step 4. Repeat the steps for the merchant\_table.csv.*



Img 2. application\_table



Img 3. merchant\_table

*-- 3. Question: Please construct SQL queries to answer the two questions below:*

*-- 3.1. How many applications were received from each merchant between 2019-10-28 and 2019-10-31?*

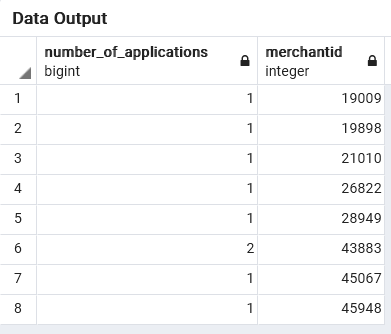
SELECT COUNT(\*) AS Number\_of\_applications,

merchantid

FROM application\_table

WHERE applicationdate > '2019-10-28 00:00:00' AND applicationdate < '2019-10-31 00:00:00'

GROUP BY merchantid



Img 4. Output Q1 3.1.

*-- 3.2. What are the top three merchant names by applications received between 2019-10-28 and 2019-10-31?*

SELECT COUNT(\*) AS Number\_of\_applications,

me.businessname

FROM application\_table AS ap

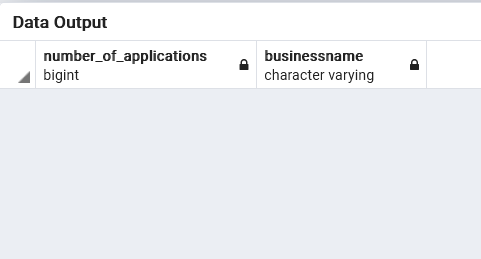
INNER JOIN merchant\_table AS me ON ap.merchantid = me.merchantid

WHERE applicationdate > '2019-10-28 00:00:00' AND applicationdate < '2019-10-31 00:00:00'

GROUP BY me.businessname

ORDER BY Number\_of\_applications DESC

LIMIT 3



Img5. Output Q1 3.2. Output is empty due it was found no relation between the merchant ID from the application table and the merchant ID from the merchant table in the dataset provided for the exercise. But, in the case of having a complete database, the query shown is accurate.

*-- Q2*

*-- 1. Create Loan\_status table*

CREATE TABLE loan\_status

(loannumber integer,

ach\_yn integer,

daysonfile integer,

reportdate timestamp

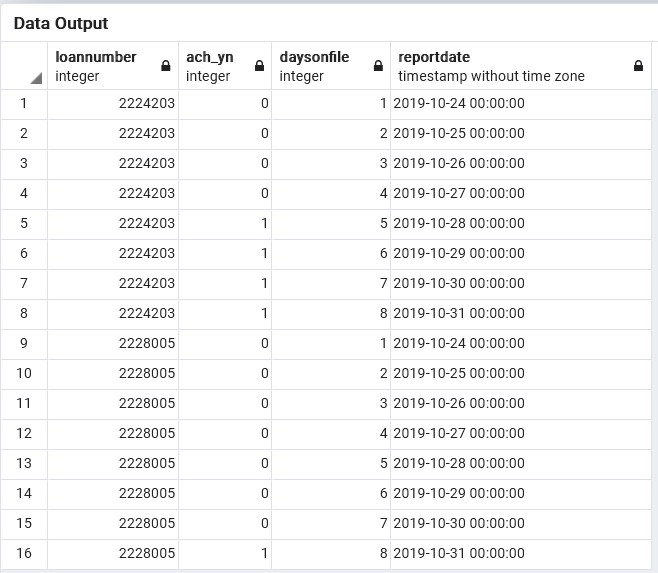
);

*-- 2. import data into the tables*

*--Step 1. Refresh Database*

*--Step 2. Go to the loan\_status table and click to open import/export, in options select import, and look for the loan\_status.csv file in downloads, update the header as ‘yes’ and delimiter as ‘,’, then go to the columns tab and validate that all columns are selected.*

*--Step 3. Click on OK.*



Img5. loan\_status table

*-- Here is a table containing ACH status of loans from 10-24-2019 to 10-31-2019. “ach\_yn” is the flag for*

*-- the daily ACH status of a loan (1 = Yes, 0 = No). “ach\_yn” = 1 means the borrower has an active*

*-- automatic payment on that day.*

*-- • When “ach\_yn” changes from 0 to 1 between two consecutive days, the loan has enrolled in*

*-- ACH (enrollment).*

*-- • When “ach\_yn” changes from 1 to 0 between two consecutive days, the loan has stopped*

*-- enrolling in ACH.*

*-- Question: Please construct SQL queries to answer the two questions below:*

*-- • how many enrollments happened within this time period (10-24-2019 to 10-31-2019)?*

WITH CTE\_Loan AS (

SELECT loannumber,

reportdate,

ach\_yn,

daysonfile,

CASE WHEN ach\_yn - lag(ach\_yn) over (PARTITION BY loannumber order by loannumber, reportdate) > 0

AND daysonfile - lag(daysonfile) over (PARTITION BY loannumber order by loannumber, reportdate) = 1 THEN 'enrolled in ACH'

WHEN ach\_yn < 0

AND daysonfile - lag(daysonfile) over (PARTITION BY loannumber order by loannumber, reportdate) = 1 THEN 'stopped enrolling in ACH'

ELSE 'No changes'

END AS ach\_enroll\_status

FROM loan\_status

WHERE reportdate >= '2019-10-24 00:00:00' AND reportdate <= '2019-10-31 00:00:00'

GROUP BY reportdate, ach\_yn, daysonfile, loannumber

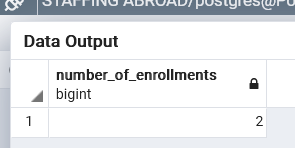
ORDER BY loannumber, reportdate

)

SELECT COUNT(\*) AS Number\_of\_enrollments

FROM CTE\_Loan

WHERE ach\_enroll\_status = 'enrolled in ACH'



Img6. Q2 first output. A Common Table Expressions ´CTE\_Loan ´ is used to build the query.

*-- • how many enrollments happened in each day (“reportdate”) between 10-24-2019 to 10-31-2019?*

WITH CTE\_Loan AS (

SELECT loannumber,

reportdate,

ach\_yn,

daysonfile,

CASE WHEN ach\_yn - lag(ach\_yn) over (PARTITION BY loannumber order by loannumber, reportdate) > 0

AND daysonfile - lag(daysonfile) over (PARTITION BY loannumber order by loannumber, reportdate) = 1 THEN 'enrolled in ACH'

WHEN ach\_yn < 0

AND daysonfile - lag(daysonfile) over (PARTITION BY loannumber order by loannumber, reportdate) = 1 THEN 'stopped enrolling in ACH'

ELSE 'No changes'

END AS ach\_enroll\_status

FROM loan\_status

WHERE reportdate >= '2019-10-24 00:00:00' AND reportdate <= '2019-10-31 00:00:00'

GROUP BY reportdate, ach\_yn, daysonfile, loannumber

ORDER BY loannumber, reportdate

)

SELECT reportdate,

CASE WHEN ach\_enroll\_status = 'enrolled in ACH' THEN COUNT (\*)

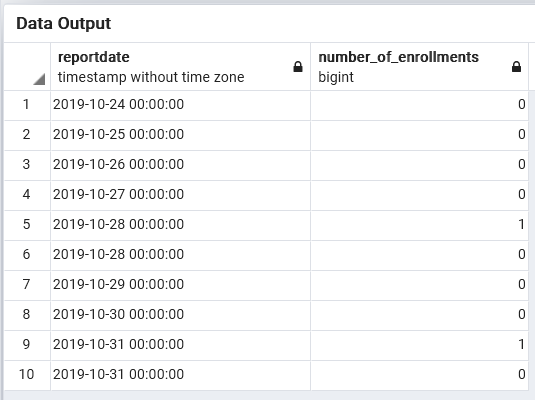
ELSE 0

END AS Number\_of\_enrollments

FROM CTE\_Loan

GROUP BY reportdate, ach\_enroll\_status

ORDER BY reportdate



Img7. Q2 second output.