CREATE TABLE customer\_info (

id\_client INTEGER PRIMARY KEY,

total\_amount NUMERIC,

gender VARCHAR(10),

age INTEGER,

count\_city INTEGER,

response\_comm BOOLEAN,

communication BOOLEAN,

tenure INTEGER

);

CREATE TABLE transactions\_info (

date\_new DATE,

id\_check INTEGER,

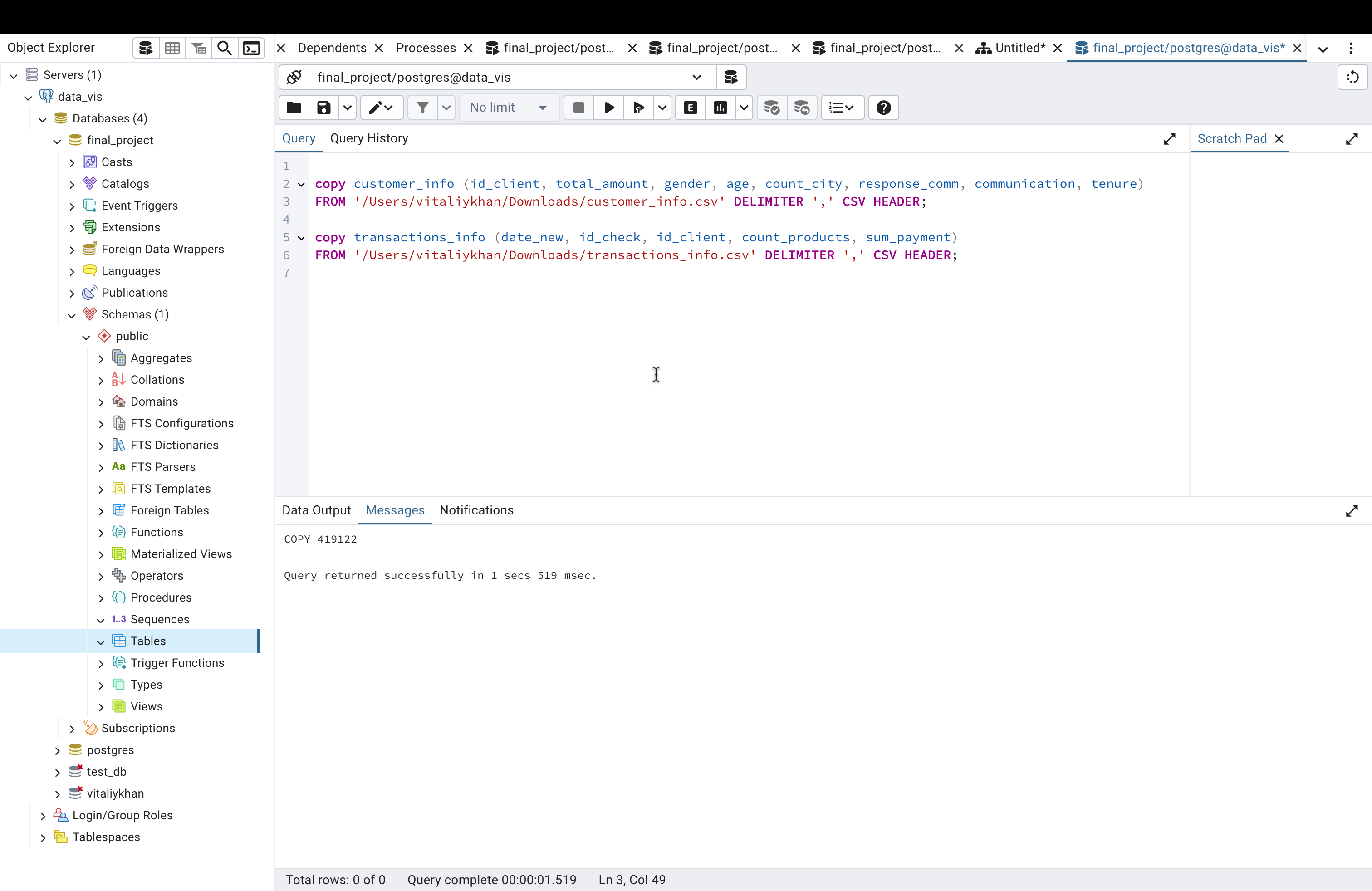
id\_client INTEGER,

count\_products NUMERIC,

sum\_payment NUMERIC,

FOREIGN KEY (id\_client) REFERENCES customer\_info(id\_client)

);



WITH monthly\_activity AS (

SELECT id\_client,

DATE\_TRUNC('month', date\_new) AS month

FROM transactions\_info

WHERE date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY id\_client, month

),

client\_activity AS (

SELECT id\_client, COUNT(DISTINCT month) AS active\_months

FROM monthly\_activity

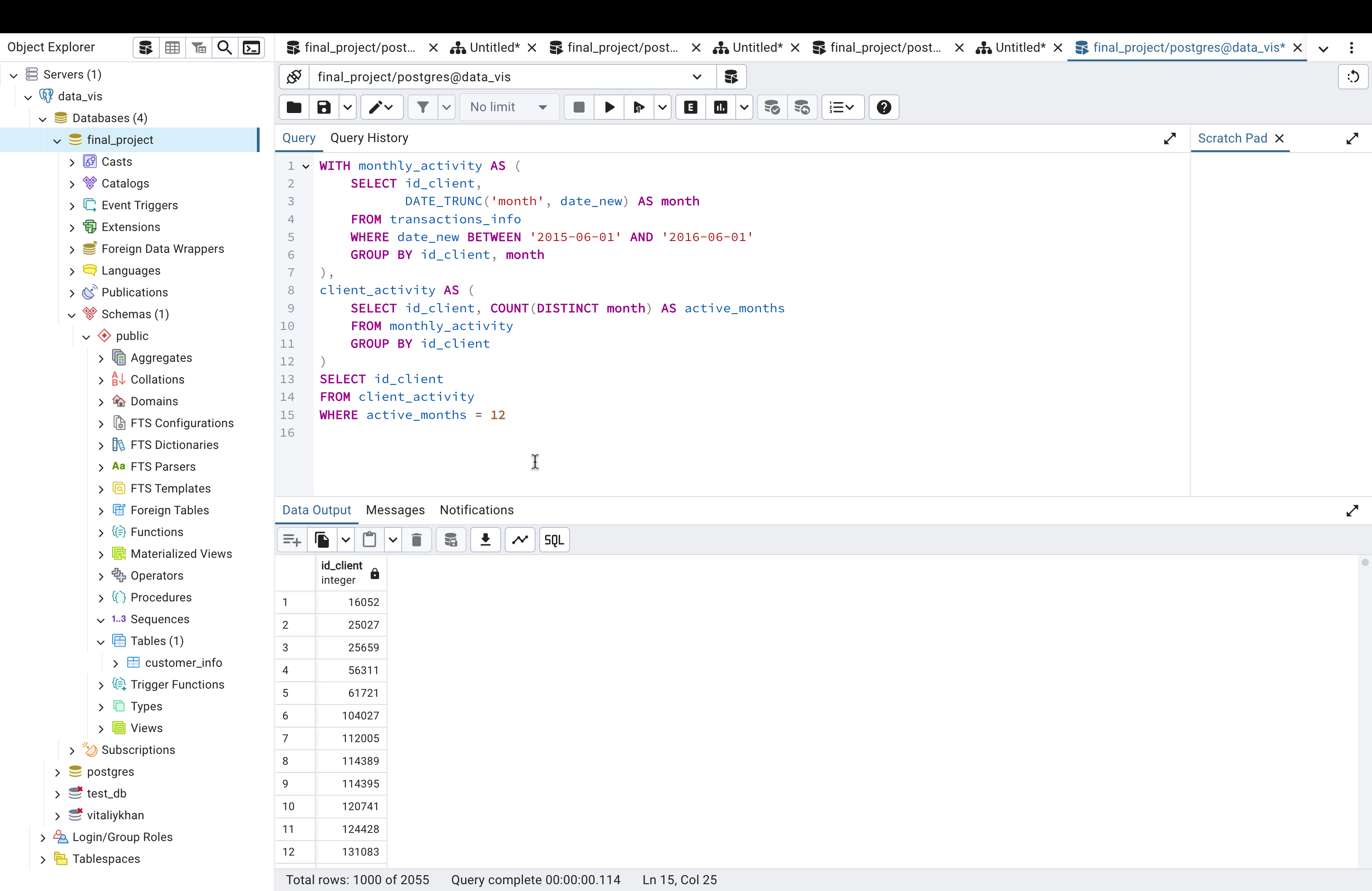
GROUP BY id\_client

)

SELECT id\_client

FROM client\_activity

WHERE active\_months = 12



Average Amount of the Check Per Month

WITH monthly\_summary AS (

SELECT

DATE\_TRUNC('month', date\_new) AS month,

SUM(sum\_payment) AS total\_amount,

COUNT(DISTINCT id\_check) AS total\_transactions

FROM

transactions\_info

WHERE

date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY

month

)

SELECT

month,

AVG(total\_amount / NULLIF(total\_transactions, 0)) AS avg\_check\_per\_month

FROM

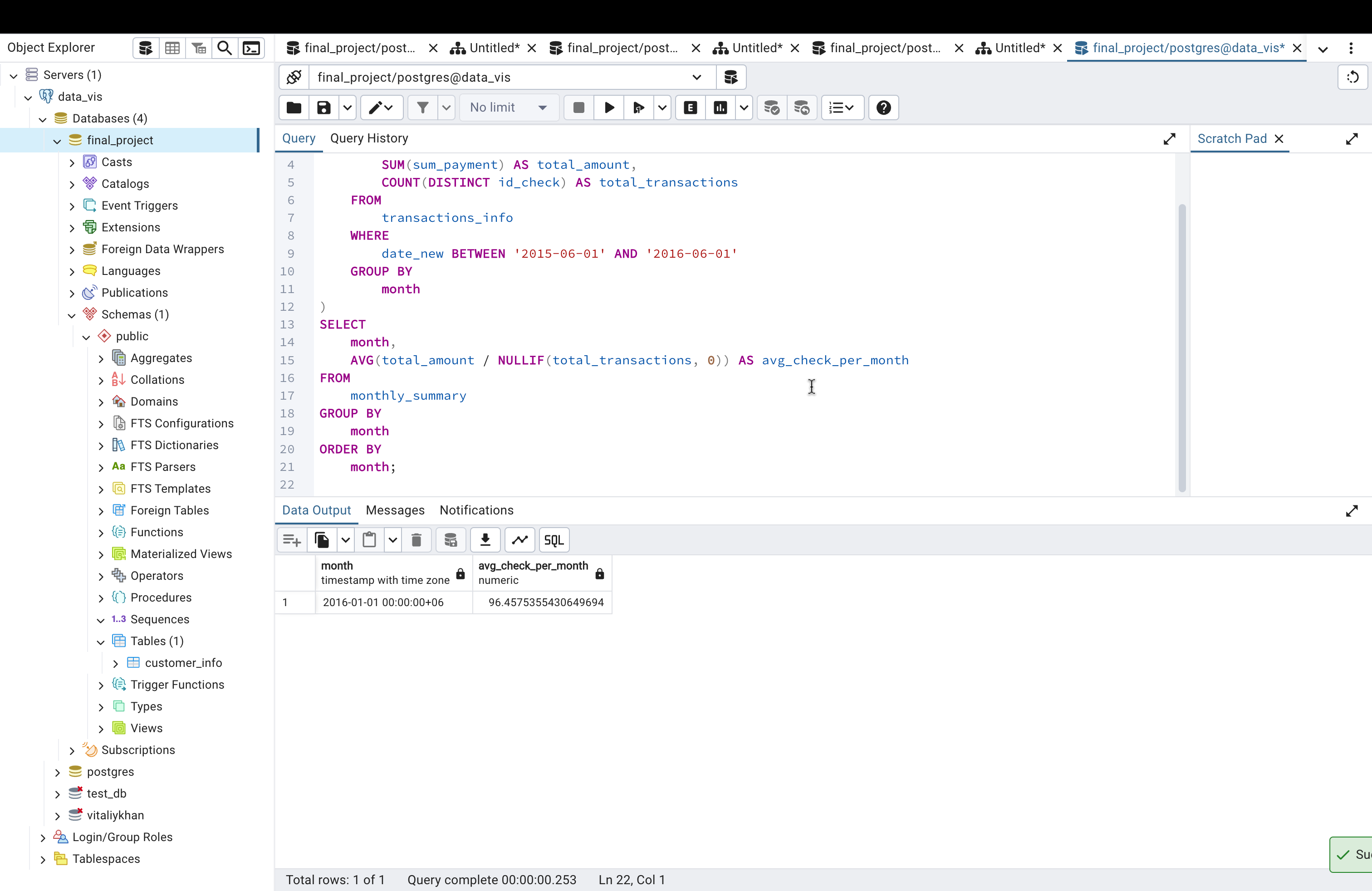
monthly\_summary

GROUP BY

month

ORDER BY

month;



Average Number of Operations Per Month

WITH monthly\_operations AS (

SELECT

DATE\_TRUNC('month', date\_new) AS month,

COUNT(id\_check) AS total\_operations

FROM

transactions\_info

WHERE

date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY

month

)

SELECT

month,

AVG(total\_operations) AS avg\_operations\_per\_month

FROM

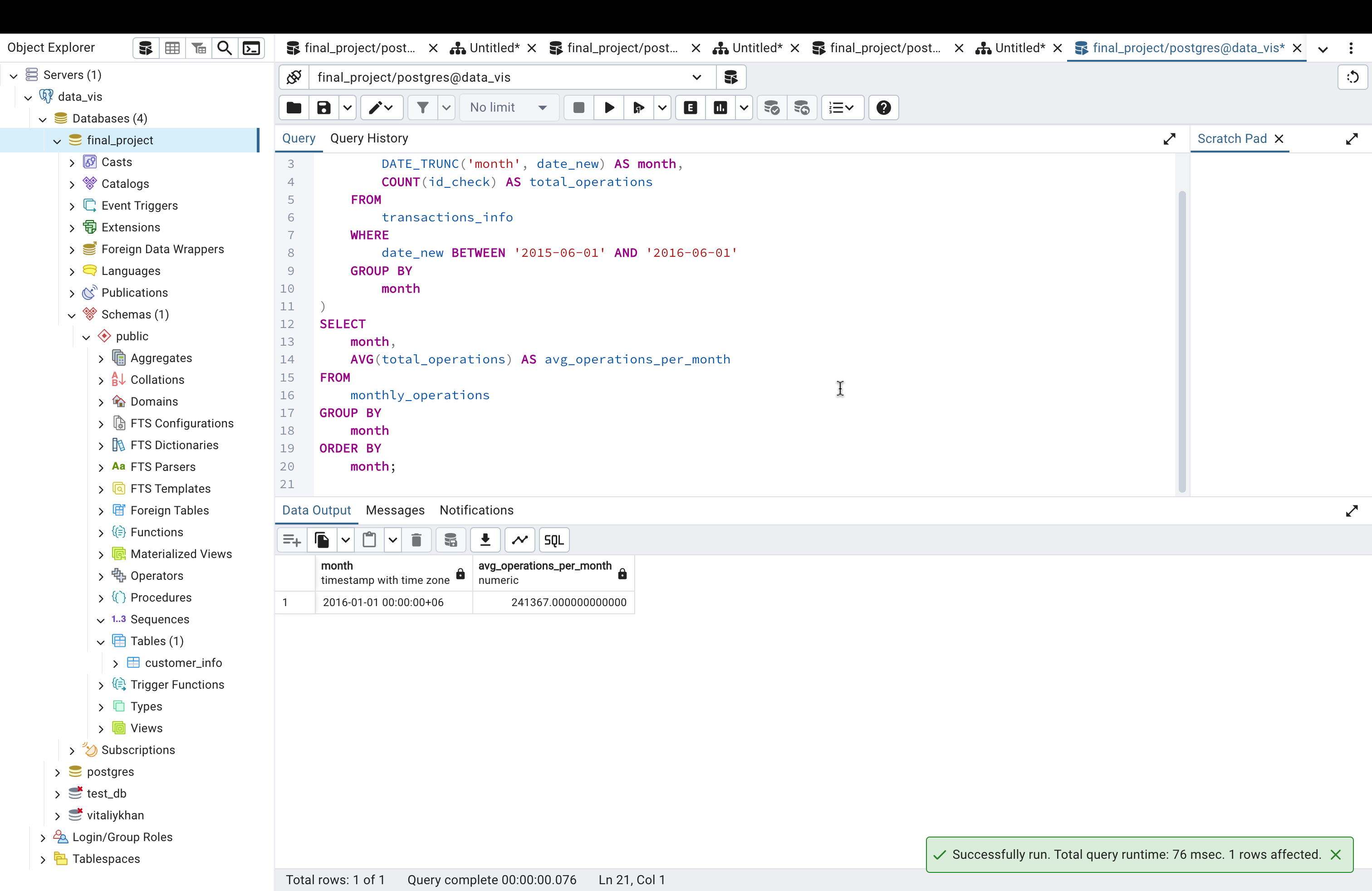
monthly\_operations

GROUP BY

month

ORDER BY

month;



Average Number of Clients Who Performed Transactions Per Month

WITH monthly\_clients AS (

SELECT

DATE\_TRUNC('month', date\_new) AS month,

COUNT(DISTINCT id\_client) AS total\_clients

FROM

transactions\_info

WHERE

date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY

month

)

SELECT

month,

AVG(total\_clients) AS avg\_clients\_per\_month

FROM

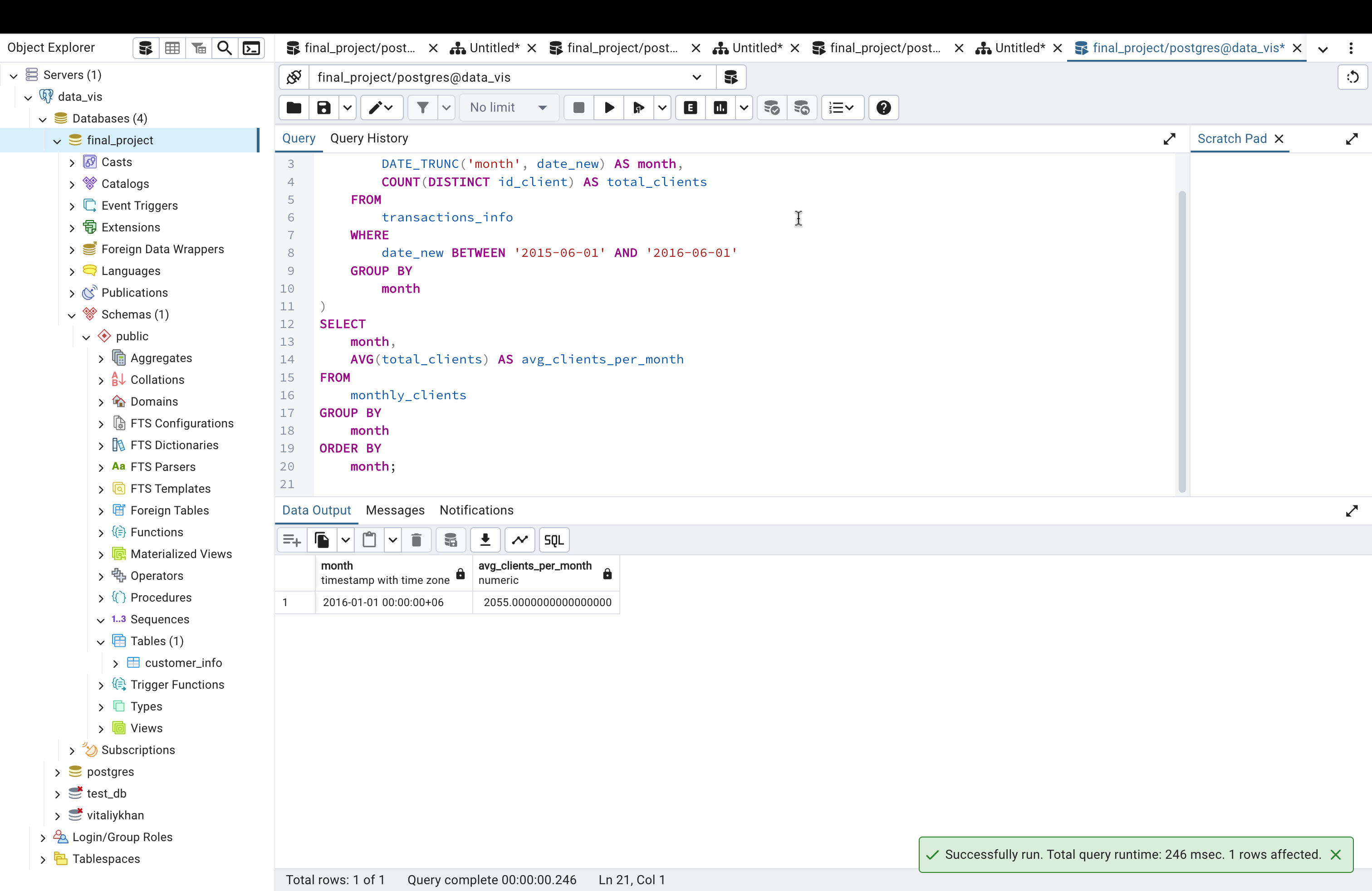
monthly\_clients

GROUP BY

month

ORDER BY

month;



Share of Total Number of Transactions for the Year

WITH total\_transactions AS (

SELECT

COUNT(id\_check) AS total\_year\_transactions

FROM

transactions\_info

WHERE

date\_new BETWEEN '2015-06-01' AND '2016-06-01'

),

monthly\_transactions AS (

SELECT

DATE\_TRUNC('month', date\_new) AS month,

COUNT(id\_check) AS month\_transactions

FROM

transactions\_info

WHERE

date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY

month

)

SELECT

mt.month,

mt.month\_transactions,

(mt.month\_transactions::decimal / NULLIF(tt.total\_year\_transactions, 0)) \* 100 AS share\_per\_month

FROM

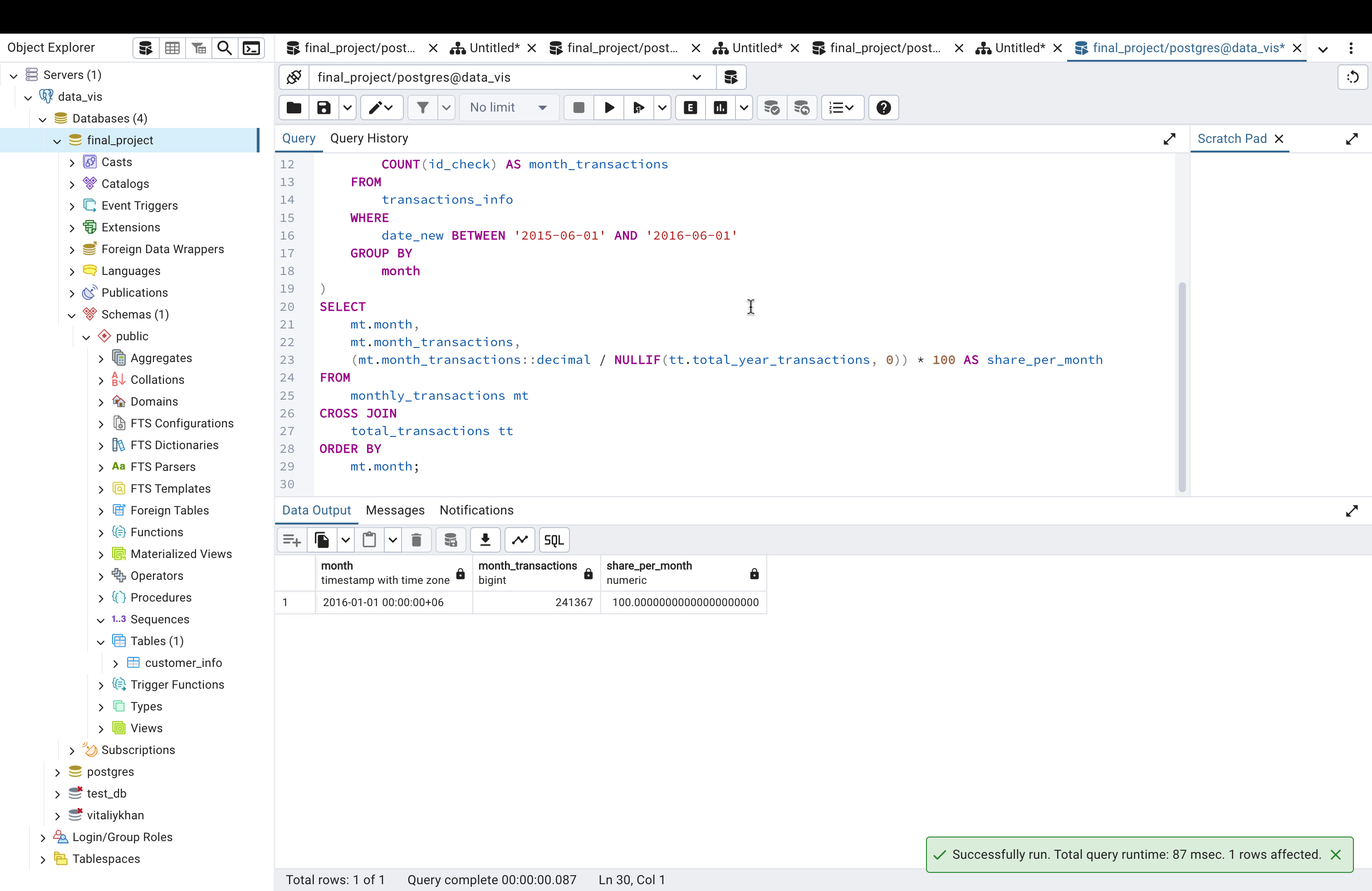
monthly\_transactions mt

CROSS JOIN

total\_transactions tt

ORDER BY

mt.month;



Print the % Ratio of M/F/NA in Each Month with Their Share of Costs

WITH monthly\_gender\_summary AS (

SELECT

DATE\_TRUNC('month', ti.date\_new) AS month,

SUM(CASE WHEN ci.gender = 'M' THEN ti.sum\_payment ELSE 0 END) AS male\_amount,

SUM(CASE WHEN ci.gender = 'F' THEN ti.sum\_payment ELSE 0 END) AS female\_amount,

SUM(CASE WHEN ci.gender IS NULL THEN ti.sum\_payment ELSE 0 END) AS na\_amount,

SUM(ti.sum\_payment) AS total\_amount

FROM

transactions\_info ti

LEFT JOIN

customer\_info ci ON ti.id\_client = ci.id\_client

WHERE

ti.date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY

month

)

SELECT

month,

(male\_amount / NULLIF(total\_amount, 0)) \* 100 AS male\_share,

(female\_amount / NULLIF(total\_amount, 0)) \* 100 AS female\_share,

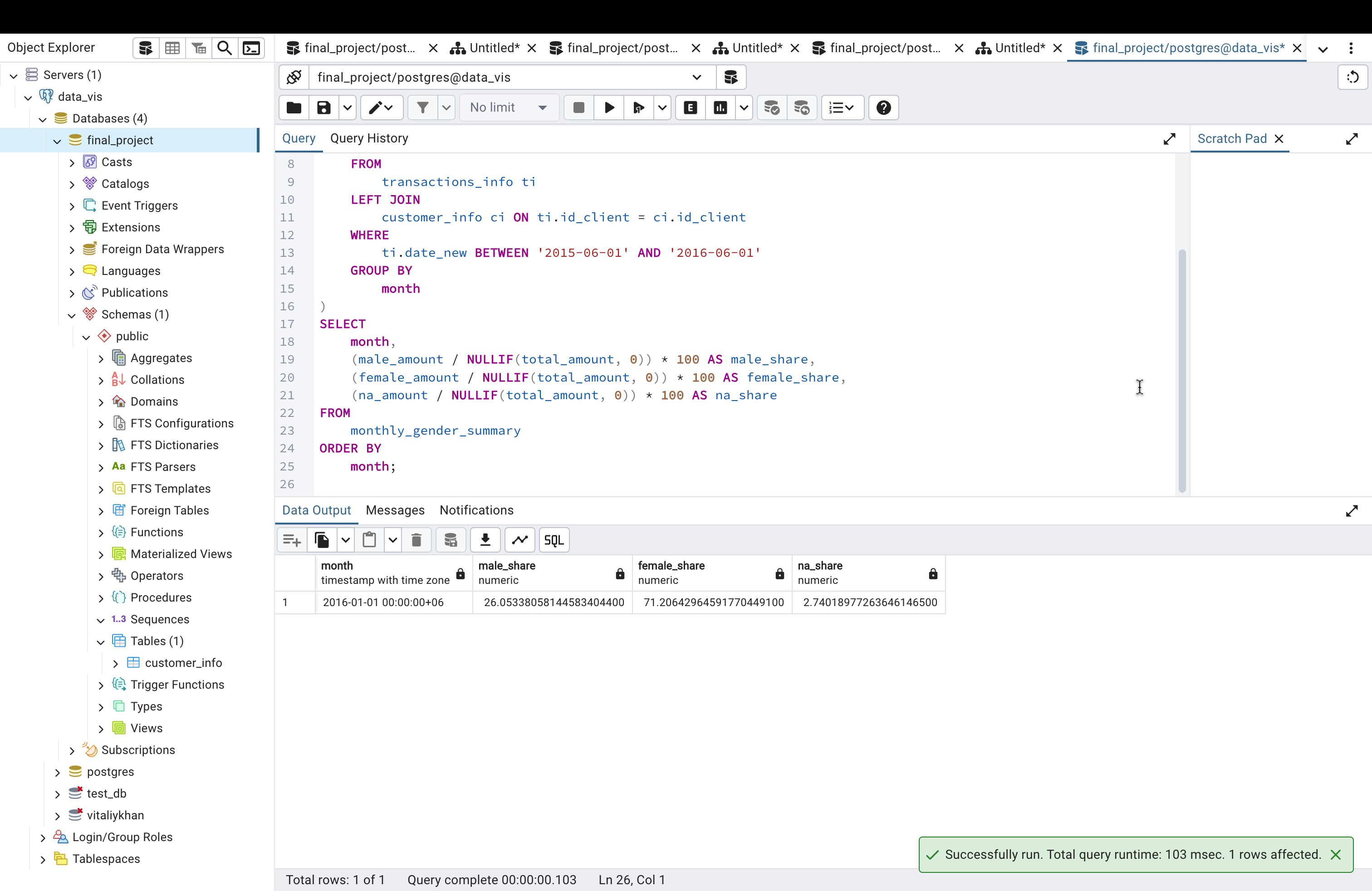
(na\_amount / NULLIF(total\_amount, 0)) \* 100 AS na\_share

FROM

monthly\_gender\_summary

ORDER BY

month;



TASK3

Age Groups and Transactions for the Entire Period

WITH age\_group\_summary AS (

SELECT

CASE

WHEN ci.age IS NULL THEN 'Unknown'

WHEN ci.age BETWEEN 0 AND 9 THEN '0-9'

WHEN ci.age BETWEEN 10 AND 19 THEN '10-19'

WHEN ci.age BETWEEN 20 AND 29 THEN '20-29'

WHEN ci.age BETWEEN 30 AND 39 THEN '30-39'

WHEN ci.age BETWEEN 40 AND 49 THEN '40-49'

WHEN ci.age BETWEEN 50 AND 59 THEN '50-59'

WHEN ci.age BETWEEN 60 AND 69 THEN '60-69'

WHEN ci.age BETWEEN 70 AND 79 THEN '70-79'

WHEN ci.age >= 80 THEN '80+'

END AS age\_group,

SUM(ti.sum\_payment) AS total\_amount,

COUNT(ti.id\_check) AS total\_transactions

FROM

transactions\_info ti

LEFT JOIN

customer\_info ci ON ti.id\_client = ci.id\_client

WHERE

ti.date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY

age\_group

)

SELECT

age\_group,

total\_amount,

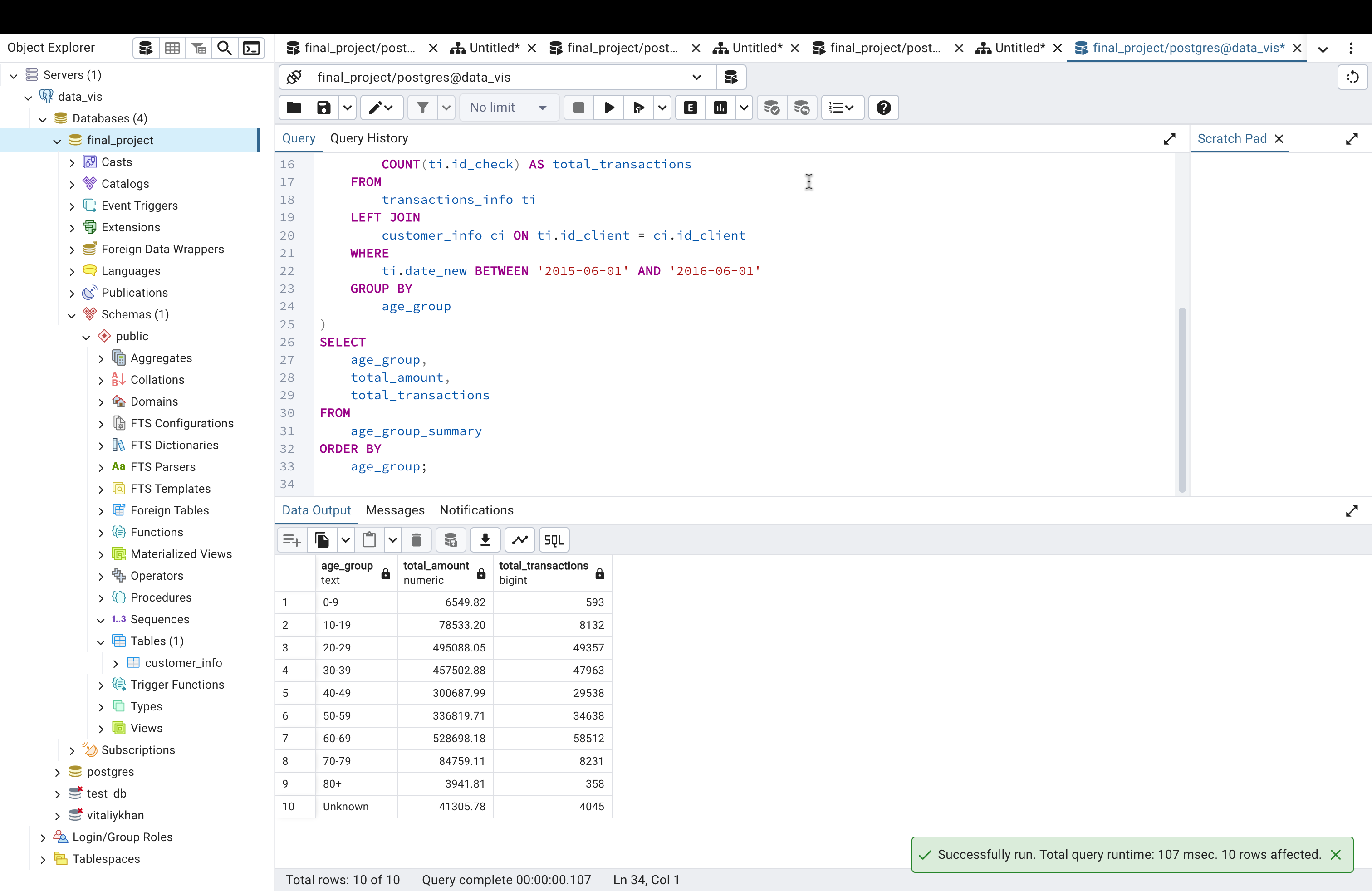
total\_transactions

FROM

age\_group\_summary

ORDER BY

age\_group;



Quarterly Averages and Percentages

WITH quarterly\_age\_group\_summary AS (

SELECT

CASE

WHEN ci.age IS NULL THEN 'Unknown'

WHEN ci.age BETWEEN 0 AND 9 THEN '0-9'

WHEN ci.age BETWEEN 10 AND 19 THEN '10-19'

WHEN ci.age BETWEEN 20 AND 29 THEN '20-29'

WHEN ci.age BETWEEN 30 AND 39 THEN '30-39'

WHEN ci.age BETWEEN 40 AND 49 THEN '40-49'

WHEN ci.age BETWEEN 50 AND 59 THEN '50-59'

WHEN ci.age BETWEEN 60 AND 69 THEN '60-69'

WHEN ci.age BETWEEN 70 AND 79 THEN '70-79'

WHEN ci.age >= 80 THEN '80+'

END AS age\_group,

DATE\_TRUNC('quarter', ti.date\_new) AS quarter,

SUM(ti.sum\_payment) AS total\_amount,

COUNT(ti.id\_check) AS total\_transactions

FROM

transactions\_info ti

LEFT JOIN

customer\_info ci ON ti.id\_client = ci.id\_client

WHERE

ti.date\_new BETWEEN '2015-06-01' AND '2016-06-01'

GROUP BY

age\_group, quarter

)

SELECT

age\_group,

quarter,

AVG(total\_amount) AS avg\_amount\_per\_quarter,

AVG(total\_transactions) AS avg\_transactions\_per\_quarter,

ROUND(100.0 \* SUM(total\_amount) / (SELECT SUM(ti.sum\_payment) FROM transactions\_info ti WHERE ti.date\_new BETWEEN '2015-06-01' AND '2016-06-01'), 2) AS percent\_amount,

ROUND(100.0 \* SUM(total\_transactions) / (SELECT COUNT(ti.id\_check) FROM transactions\_info ti WHERE ti.date\_new BETWEEN '2015-06-01' AND '2016-06-01'), 2) AS percent\_transactions

FROM

quarterly\_age\_group\_summary

GROUP BY

age\_group, quarter

ORDER BY

age\_group, quarter;

