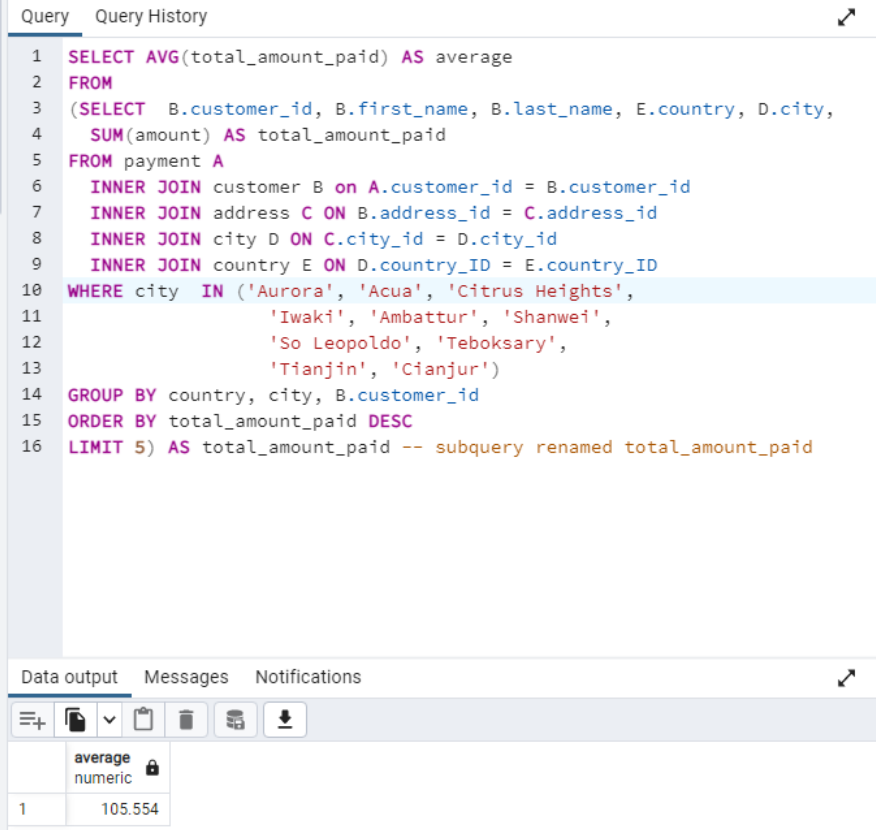
ACHIEVEMENT 3.8

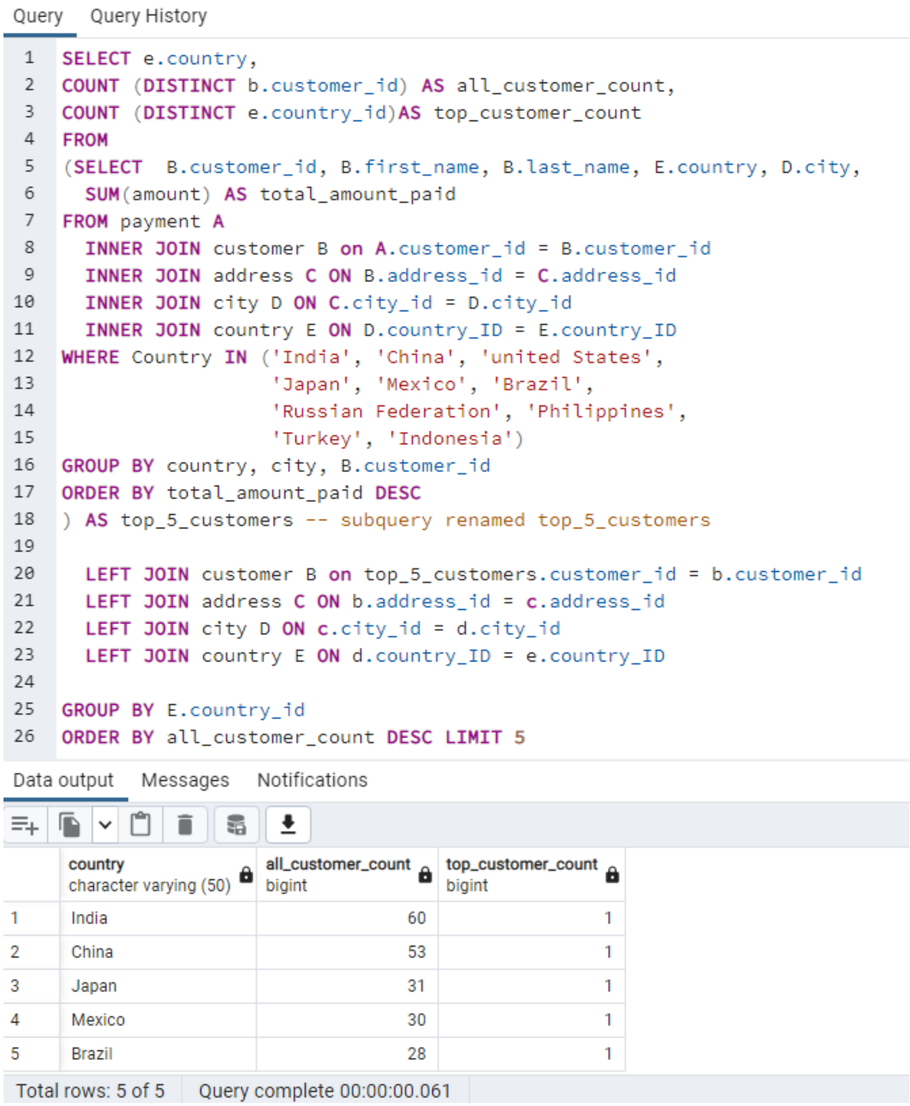
1)



```sql
SELECT AVG(total_amount_paid) AS average
FROM
(SELECT  B.customer_id, B.first_name, B.last_name, E.country, D.city,
  SUM(amount) AS total_amount_paid
FROM payment A
  INNER JOIN customer B on A.customer_id = B.customer_id
  INNER JOIN address C ON B.address_id = C.address_id
  INNER JOIN city D ON C.city_id = D.city_id
  INNER JOIN country E ON D.country_ID = E.country_ID
WHERE city  IN ('Aurora', 'Acua', 'Citrus Heights',
                  'Iwaki', 'Ambattur', 'Shanwei',
                  'So Leopoldo', 'Teboksary',
                  'Tianjin', 'Cianjur')
GROUP BY country, city, B.customer_id
ORDER BY total_amount_paid DESC
LIMIT 5) AS total_amount_paid -- subquery renamed total_amount_paid
```

	average numeric
1	105.554

2)



```sql
SELECT e.country,
COUNT (DISTINCT b.customer_id) AS all_customer_count,
COUNT (DISTINCT e.country_id)AS top_customer_count
FROM
(SELECT  B.customer_id, B.first_name, B.last_name, E.country, D.city,
  SUM(amount) AS total_amount_paid
FROM payment A
  INNER JOIN customer B on A.customer_id = B.customer_id
  INNER JOIN address C ON B.address_id = C.address_id
  INNER JOIN city D ON C.city_id = D.city_id
  INNER JOIN country E ON D.country_ID = E.country_ID
WHERE Country IN ('India', 'China', 'united States',
                  'Japan', 'Mexico', 'Brazil',
                  'Russian Federation', 'Philippines',
                  'Turkey', 'Indonesia')
GROUP BY country, city, B.customer_id
ORDER BY total_amount_paid DESC
) AS top_5_customers -- subquery renamed top_5_customers

  LEFT JOIN customer B on top_5_customers.customer_id = b.customer_id
  LEFT JOIN address C ON b.address_id = c.address_id
  LEFT JOIN city D ON c.city_id = d.city_id
  LEFT JOIN country E ON d.country_ID = e.country_ID

GROUP BY E.country_id
ORDER BY all_customer_count DESC LIMIT 5
```

	country character varying (50)	all_customer_count bigint	top_customer_count bigint
1	India	60	1
2	China	53	1
3	Japan	31	1
4	Mexico	30	1
5	Brazil	28	1

Total rows: 5 of 5   Query complete 00:00:00.061

3)

3a) Both step1 and step2 most likely can be completed without using subqueries by using a temporary table.

3b) Subqueries are useful because it save time to write different queries if they changes.