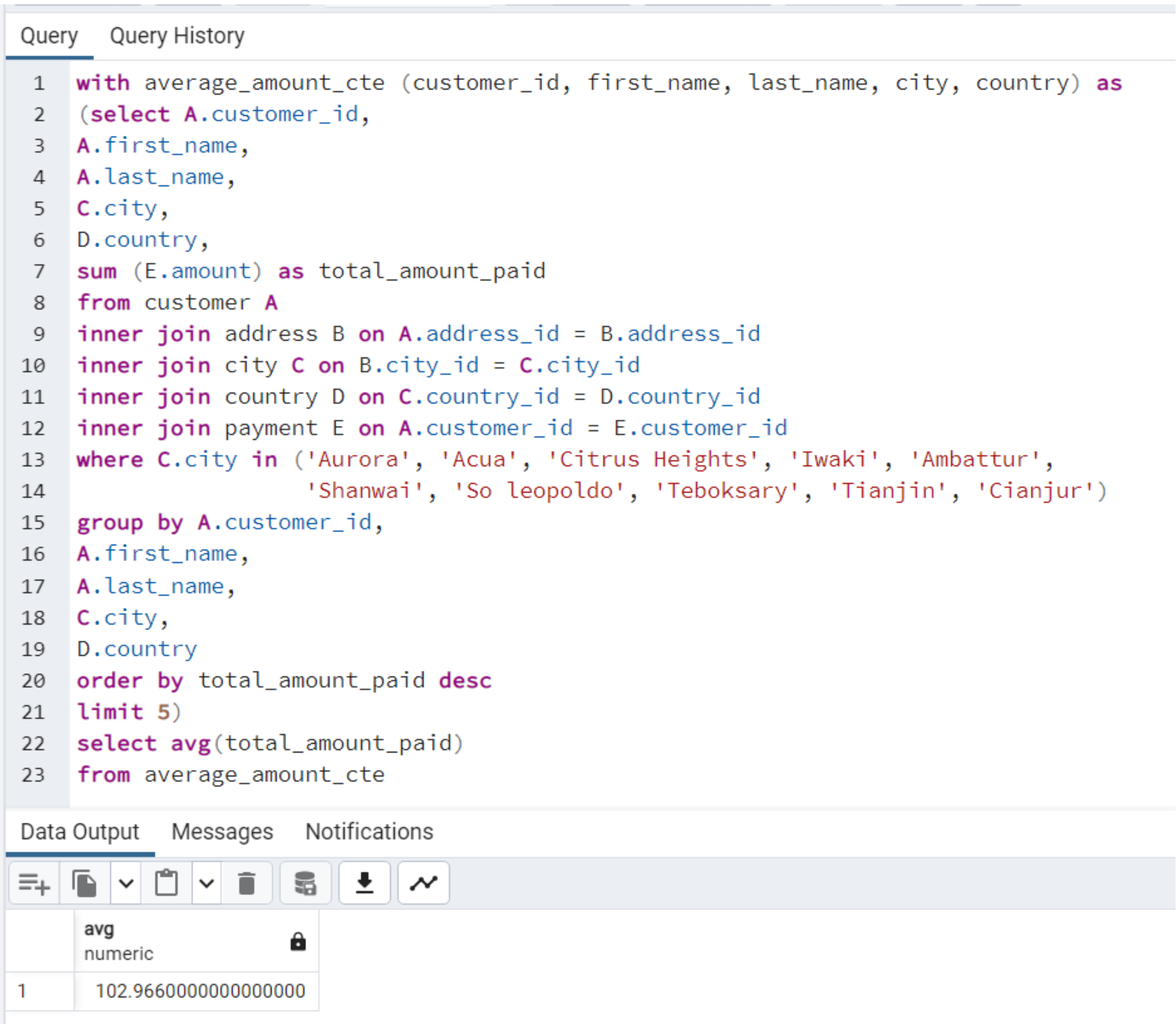
Surekha

Ach.3 Task 3.9

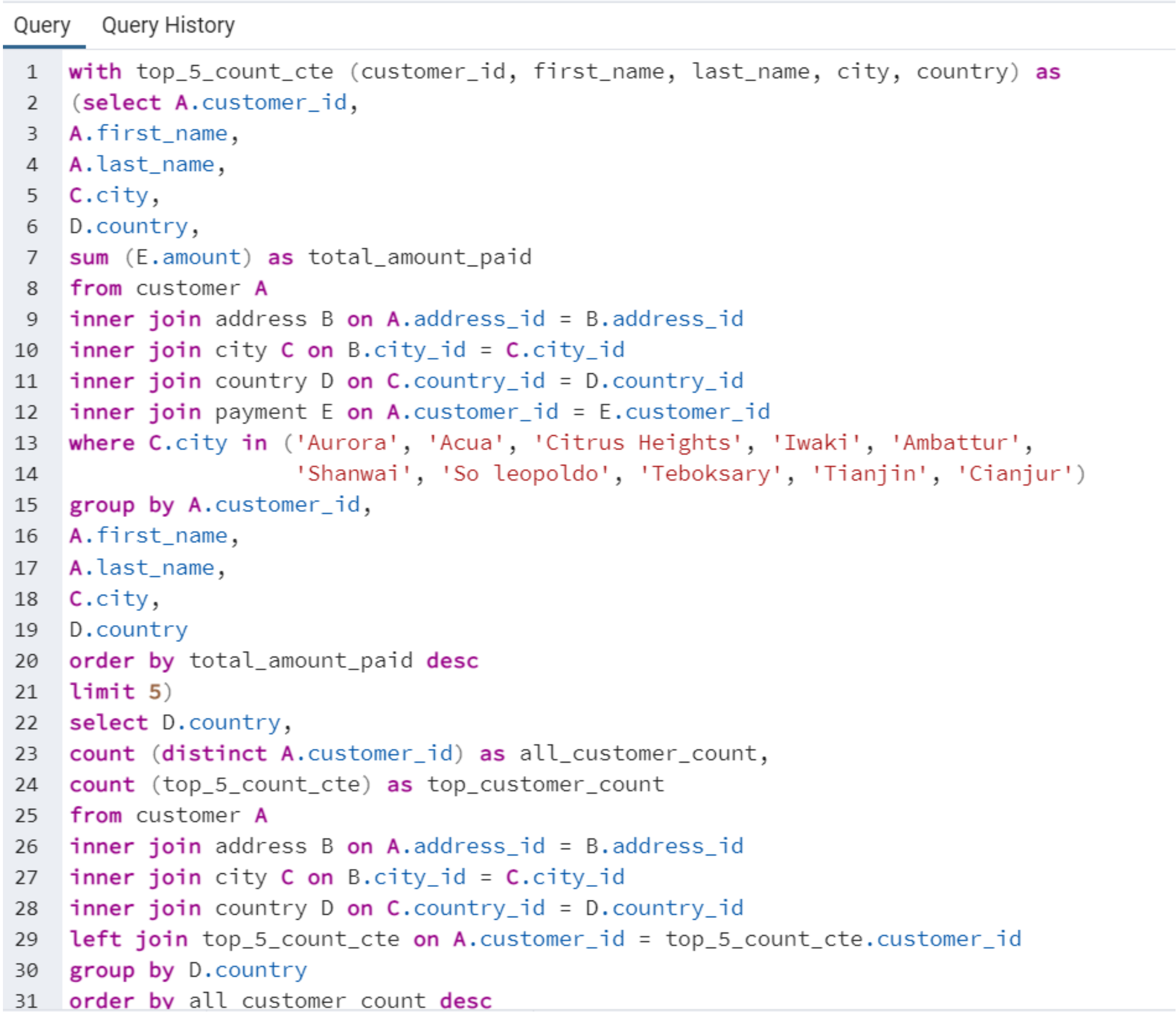
Exercise: 3.9

Step: 1

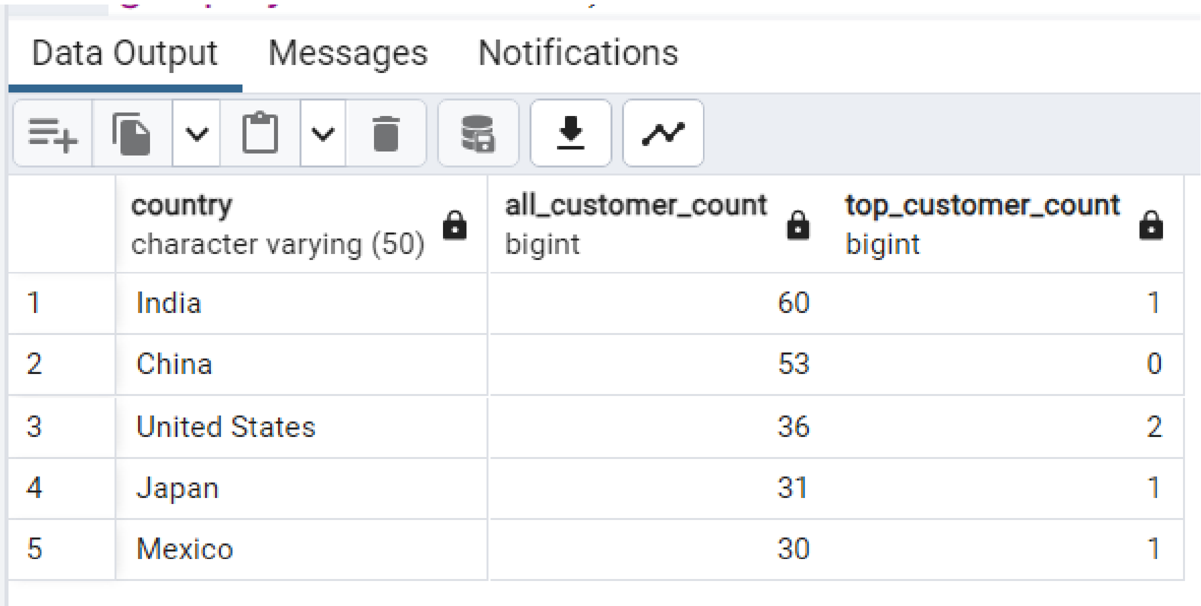


```
with average_amount_cte (customer_id, first_name, last_name, city, country) as
(select A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country,
sum (E.amount) as total_amount_paid
from customer A
inner join address B on A.address_id = B.address_id
inner join city C on B.city_id = C.city_id
inner join country D on C.country_id = D.country_id
inner join payment E on A.customer_id = E.customer_id
where C.city in ('Aurora', 'Acua', 'Citrus Heights', 'Iwaki', 'Ambattur',
                 'Shanwai', 'So leopoldo', 'Teboksary', 'Tianjin', 'Cianjur')
group by A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country
order by total_amount_paid desc
limit 5)
select avg(total_amount_paid)
from average_amount_cte
```

	avg numeric
1	102.9660000000000000



```
with top_5_count_cte (customer_id, first_name, last_name, city, country) as
(select A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country,
sum (E.amount) as total_amount_paid
from customer A
inner join address B on A.address_id = B.address_id
inner join city C on B.city_id = C.city_id
inner join country D on C.country_id = D.country_id
inner join payment E on A.customer_id = E.customer_id
where C.city in ('Aurora', 'Acua', 'Citrus Heights', 'Iwaki', 'Ambattur',
                  'Shanwai', 'So leopoldo', 'Teboksary', 'Tianjin', 'Cianjur')
group by A.customer_id,
A.first_name,
A.last_name,
C.city,
D.country
order by total_amount_paid desc
limit 5)
select D.country,
count (distinct A.customer_id) as all_customer_count,
count (top_5_count_cte) as top_customer_count
from customer A
inner join address B on A.address_id = B.address_id
inner join city C on B.city_id = C.city_id
inner join country D on C.country_id = D.country_id
left join top_5_count_cte on A.customer_id = top_5_count_cte.customer_id
group by D.country
order by all_customer_count desc
```



	country character varying (50)	all_customer_count bigint	top_customer_count bigint
1	India	60	1
2	China	53	0
3	United States	36	2
4	Japan	31	1
5	Mexico	30	1

Step 2:

- I believe that the CTE will perform better than the subquery because it is more optimal and only requires for the CTE to be defined once, as opposed to the subquery having to be defined every time we want to refer to it.

- **Costs:**
  a. Subquery 1 (Exercise 3.8): 256msec
  b. Subquery 2 (Exercise 3.8): 236 msec
  c. CTE 1: 164 msec
  d. CTE 2: 186 msec
- These results did not surprise me, as I had guessed that the CTE's would take less time to run because they were more optimal. CTEs are easier to read and only have to be defined once.
- Surprisingly, changing my subqueries to CTEs were not challenging. Given that I knew the format for the CTE, I just had to move the statements around. All I had to do was make sure the CTE statement came first, places the subquery right after, and then placed all portions of the outer query directly after the subquery .