Answers 3.8

Step 1: Find the average amount paid by the top 5 customers.

Query | Query History

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

Data Output | Messages | Notifications

	average numeric
1	105.5540000000000000

Step 2: Find out how many of the top 5 customers you identified in step 1 are based within each country.

Query | Query History

```sql
SELECT country.country,
COUNT(DISTINCT customer.customer_id) AS all_customer_count,
COUNT(DISTINCT top_5_customers) AS top_customer_count
FROM customer
INNER JOIN address ON customer.address_id = address.address_id
INNER JOIN city ON address.city_id = city.city_id
INNER JOIN country ON city.country_id = country.country_id
LEFT JOIN (SELECT customer.customer_id, first_name, last_name, country, city,
           SUM(amount) AS total_amount_paid
FROM customer
INNER JOIN address ON customer.address_id = address.address_id
INNER JOIN city ON address.city_id = city.city_id
INNER JOIN country ON city.country_id = country.country_id
INNER JOIN payment ON customer.customer_id = payment.customer_id
WHERE city IN('Aurora', 'Acua', 'Citrus Heights', 'Iwaki', 'Ambattur',
                'Shanwei', 'So Leopoldo', 'Teboksary', 'Tianjin', 'Cianjur')
GROUP BY customer.customer_id, country, city
ORDER BY SUM(amount) DESC
LIMIT 5) AS top_5_customers
ON customer.customer_id = top_5_customers.customer_id
GROUP BY country.country
ORDER BY all_customer_count DESC
LIMIT 5
```

Data Output | Messages | Notifications

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

Step 3:

- Do you think steps 1 and 2 could be done without using subqueries?

I had a couple of attempts to find a way to answer the questions without using a subquery. In the first example, I tried to add the average column in the existing query but nesting ang and sum was not possible so that was not successful. About the second question, even specifying the top customer's names or IDs didn't exclude using the subquery to have two separate clauses, to count all customers and separately only the top. So, I allow the possibility of another way that can be used here, but I haven't uncovered it yet.

- When do you think subqueries are useful?

It is very useful when we need to have several different clauses, without impacting each other in one table. Or when nesting is not an option. I think the most common use of subqueries is when we have very complex requests.