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

```sql
SELECT AVG (total_amount_paid)
FROM
(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 payment E ON A.customer_id = E.customer_id
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
WHERE C. city IN ('Aurora', 'Acua', 'Citrus Heights', 'Iwaki', 'Amb
GROUP BY A. customer_id,
         A. first_name,
         A. last_name,
         C. city,
         D. country
ORDER BY total_amount_paid DESC
LIMIT 5) AS average
```

Data Output | Messages | Notifications

	avg 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 D.country,
COUNT(DISTINCT A.customer_id) AS all_customers_count,
COUNT(top_5_customers) 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
(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 payment E ON A.customer_id = E.customer_id
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
WHERE C. city IN ('Aurora', 'Acua', 'Citrus Heights', 'Iwaki', 'Ambattur', 'Shanwei', 'So Leopoldo', 'Teboksary',
GROUP BY A. customer_id,
         A. first_name,
         A. last_name,
         C. city,
         D. country
ORDER BY total_amount_paid DESC
LIMIT 5) top_5_customers ON A.customer_id = top_5_customers.customer_id
GROUP BY D.country
ORDER BY all_customers_count DESC
LIMIT 5;
```

	country character varying (50)	all_customers_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: In my opinion writing an outer query in step 2 was sort of arduous; whenever is possible try to eliminate nested queries by using JOIN operations by joining tables together, so it is possible to retrieve data from multiple tables in a single query, avoiding the need for subqueries.