








3.8 Performing Subqueries

Answers 3.8

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

```
SELECT AVG(Total_Amount_Paid) AS average_by_top5_customers
FROM
```

```
(SELECT B.customer_id, B.first_name,
        B.last_name, D.city, E.country,
SUM(A.amount) AS Total_Amount_Paid
FROM customer B
INNER JOIN payment A ON B.customer_id = A.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 D.city IN ('Aurora', 'Atlixco',
                'Xintai', 'Adoni', 'Dhule (Dhulia)',
                'Kurashiki', 'Pingxiang',
                'Sivas', 'Celaya', 'So Leopoldo')
GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country
ORDER BY Total_Amount_Paid DESC
LIMIT 5) AS total_amount_paid;
```

Query	Query History	Data Output	More			
<pre>1 SELECT AVG(Total_Amount_Paid) AS average_by_top5_customers 2 FROM 3 4 (SELECT B.customer_id, B.first_name, 5 B.last_name, D.city, E.country, 6 SUM(A.amount) AS Total_Amount_Paid 7 FROM customer B 8 INNER JOIN payment A ON B.customer_id = A.customer_id 9 INNER JOIN address C ON B.address_id = C.address_id 10 INNER JOIN city D ON C.city_id = D.city_id 11 INNER JOIN country E ON D.country_id = E.country_id 12 WHERE D.city IN ('Aurora', 'Atlixco', 13 'Xintai', 'Adoni', 'Dhule (Dhulia)', 14 'Kurashiki', 'Pingxiang', 15 'Sivas', 'Celaya', 'So Leopoldo') 16 GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.coun 17 ORDER BY Total_Amount_Paid DESC 18 LIMIT 5) AS total_amount_paid; 19</pre>	<div></div> <table><thead><tr><th></th><th>average_by_top5_customers numeric</th></tr></thead><tbody><tr><td>1</td><td>107.3540000000000000</td></tr></tbody></table>		average_by_top5_customers numeric	1	107.3540000000000000	
	average_by_top5_customers numeric					
1	107.3540000000000000					

- **Step 2: Find out how many of the top 5 customers are based within each country.**

```

SELECT DISTINCT(A.country),
      COUNT (DISTINCT D.customer_id) AS all_customer_count,
      COUNT(DISTINCT A.country) AS top_customer_count
FROM country A
INNER JOIN city b ON A.country_id=B.country_id
INNER JOIN address c ON B.city_id=C.city_id
INNER JOIN customer D ON C.address_id=D.address_id
LEFT JOIN (SELECT A.customer_id, A.first_name,
                  A.last_name,
                  E.country, B.city,
                  SUM(C.amount) AS Total_paid
            FROM customer A
            INNER JOIN address D ON A.address_id=D.address_id
            INNER JOIN city B ON D.city_id=B.city_id
            INNER JOIN country E ON B.country_id=E.country_id
            INNER JOIN payment C ON A.customer_id= C.customer_ID
            WHERE E.country IN ('India', 'China','United States',
                                'Japan','Mexico', 'Brazil',
                                'Russian Federation', 'Turkey',
                                'Philippines', 'Indonesia'))
      GROUP BY A.customer_id, E.country, B.city
      ORDER BY Total_paid DESC
      LIMIT 5) AS top_5_customers ON A.country=top_5_customers.country
GROUP BY A.country, top_5_customers
ORDER BY all_customer_count DESC
LIMIT 5;

```

Rockbuster/postgres@PostgreSQL 15

Query

```

1 SELECT DISTINCT(A.country),
2     COUNT (DISTINCT D.customer_id) AS all_customer_count,
3     COUNT(DISTINCT A.country) AS top_customer_count
4 FROM country A
5 INNER JOIN city b ON A.country_id=B.country_id
6 INNER JOIN address c ON B.city_id=C.city_id
7 INNER JOIN customer D ON C.address_id=D.address_id
8 LEFT JOIN (SELECT A.customer_id, A.first_name,
9                 A.last_name,
10                E.country, B.city,
11                SUM(C.amount) AS Total_paid
12            FROM customer A
13            INNER JOIN address D ON A.address_id=D.address_id
14            INNER JOIN city B ON D.city_id=B.city_id
15            INNER JOIN country E ON B.country_id=E.country_id
16            INNER JOIN payment C ON A.customer_id= C.customer_ID
17            WHERE E.country IN ('India', 'China','United States',
18                                'Japan','Mexico', 'Brazil',
19                                'Russian Federation', 'Turkey',
20                                'Philippines', 'Indonesia'))
21
22 GROUP BY A.customer_id, E.country, B.city
23 ORDER BY Total_paid DESC
24 LIMIT 5) AS top_5_customers ON A.country=top_5_customers.country
25 GROUP BY A.country, top_5_customers
26 ORDER BY all_customer_count DESC
27 LIMIT 5;

```

Query History 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:

1. Write 1 to 2 short paragraphs on the following:

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

I think both could have been done with a JOIN and is more efficient in most cases.

- **When do you think subqueries are useful?**

We should use subqueries when the result we want requires more than one query, and each subquery provides a subset of the table involved in the query.