

ANSWERS 3.8

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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure for 'Rockbuster' on a PostgreSQL 15 server. The main pane shows a SQL query in the 'Query' tab. The query is as follows:

```

1 SELECT AVG(total_amount_paid) AS average
2 FROM
3 (SELECT B.customer_id,
4  B.first_name,
5  B.last_name,
6  D.city,
7  E.country,
8  SUM(A.amount) as Total_Amount_Paid
9 FROM payment A
10 INNER JOIN customer B ON A.customer_id = B.customer_id
11 INNER JOIN address C ON B.address_id = C.address_id
12 INNER JOIN city D ON C.city_id = D.city_id
13 INNER JOIN country E ON D.country_id = E.country_id
14 WHERE city IN ('Aurora', 'Atlisco', 'Xintai', 'Adoni', 'Dhule (Dhulia)', 'Kursahiki', 'Pingxiang', 'Sivas')
15 GROUP BY B.customer_id,
16  B.first_name,
17  B.last_name,
18  D.city,
19  E.country
20 ORDER BY Total_Amount_Paid DESC
21 LIMIT 5) AS total_amount_paid

```

The 'Data Output' tab shows the result of the query:

average
107.35400000000000000000

Total rows: 1 of 1 Query complete 00:00:00.731 Ln 16, Col 16

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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure for 'Rockbuster' on a PostgreSQL 15 server. The main pane shows a SQL query in the 'Query' tab. The query is as follows:

```

1 SELECT E.country,
2 COUNT (DISTINCT B.customer_id) AS all_customer_count,
3 COUNT (DISTINCT top_5_customers.customer_id) AS top_customer_count
4 FROM country E
5 INNER JOIN city D ON E.country_id = D.country_id
6 INNER JOIN address C ON D.city_id = C.city_id
7 INNER JOIN customer B ON C.address_id = B.address_id
8 LEFT JOIN
9 (SELECT B.customer_id, B.first_name, B.last_name, D.city, E.country,
10  SUM(A.amount) AS total_amount_paid
11 FROM payment A
12 INNER JOIN customer B ON A.customer_id = B.customer_id
13 INNER JOIN address C ON B.address_id = C.address_id
14 INNER JOIN city D ON C.city_id = D.city_id
15 INNER JOIN country E ON D.country_id = E.country_id
16 WHERE city IN ('Aurora', 'Atlisco', 'Xintai', 'Adoni', 'Dhule (Dhulia)', 'Kursahiki',
17  'Pingxiang', 'Sivas', 'Celaya', 'So Leopoldo')
18 GROUP BY B.customer_id, B.first_name, B.last_name, D.city, E.country
19 ORDER BY total_amount_paid DESC
20 LIMIT 5) AS top_5_customers
21 ON E.country = top_5_customers.country

```

The 'Data Output' tab shows the result of the query:

country	all_customer_count	top_customer_count
Mexico	30	2
United States	36	1
India	60	1
Turkey	15	1
American Samoa	1	0

Total rows: 5 of 5 Query complete 00:00:00.260 Ln 25, Col 1

ANSWERS 3.8

```

SELECT E.country,
COUNT (DISTINCT B.customer_id) AS all_customer_count,
COUNT (DISTINCT top_5_customers.customer_id) AS top_customer_count
FROM country E
INNER JOIN city D ON E.country_id = D.country_id
INNER JOIN address C ON D.city_id = C.city_id
INNER JOIN customer B ON C.address_id = B.address_id
LEFT JOIN
(SELECT B.customer_id, B.first_name, B.last_name, D.city, E.country,
SUM(A.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', 'Atlixco', 'Xintai', 'Adoni', 'Dhule (Duhlia)', 'Kursahiki',
'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 top_5_customers
ON E.country = top_5_customers.country
GROUP BY E.country
ORDER BY top_customer_count DESC
LIMIT 5

```

Step 3:

1. Do you think steps 1 and 2 could be done without using subqueries?
 - It can be done without subqueries, but then we would have to write a lot of different queries to get the same result. This would be extremely time consuming and not cost effective.
2. When do you think subqueries are useful?
 - Subqueries are quite useful when we are working on getting output from lot of columns from different tables. We can join multiple tables and get the desired output from writing subqueries and save time and cost.