3.8 Subqueries

1.

```
Select AVG (payment) AS average
FROM
(SELECT A.customer_id, A.first_name, A.last_name, D.country, C.city,
SUM(E.amount) AS payment
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 city IN ('Aurora', 'Tokat', 'Tarsus', 'Altixco',
               'Emeishan', 'Pontainak', 'Shimoga', 'Aparaceida de Goiania',
               'Zalantun', 'Taguig')
GROUP BY A.customer_id, D.country, C.city
ORDER BY payment DESC
LIMIT 5) AS Total_amount_paid
```

	average numeric
1	104.946

2.

```
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, D.country, C.city,
SUM(E.amount) AS payment
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
GROUP BY A.customer_id, D.country, C.city
ORDER BY payment 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
```

	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

3. The first query can be done without a subquery because we are only looking for one figure. The second one needs the subquery because we are pulling from different databases.

4. Subqueries are useful when pulling from multiple data sources that may have sensitive information so you can get the exact data you need.