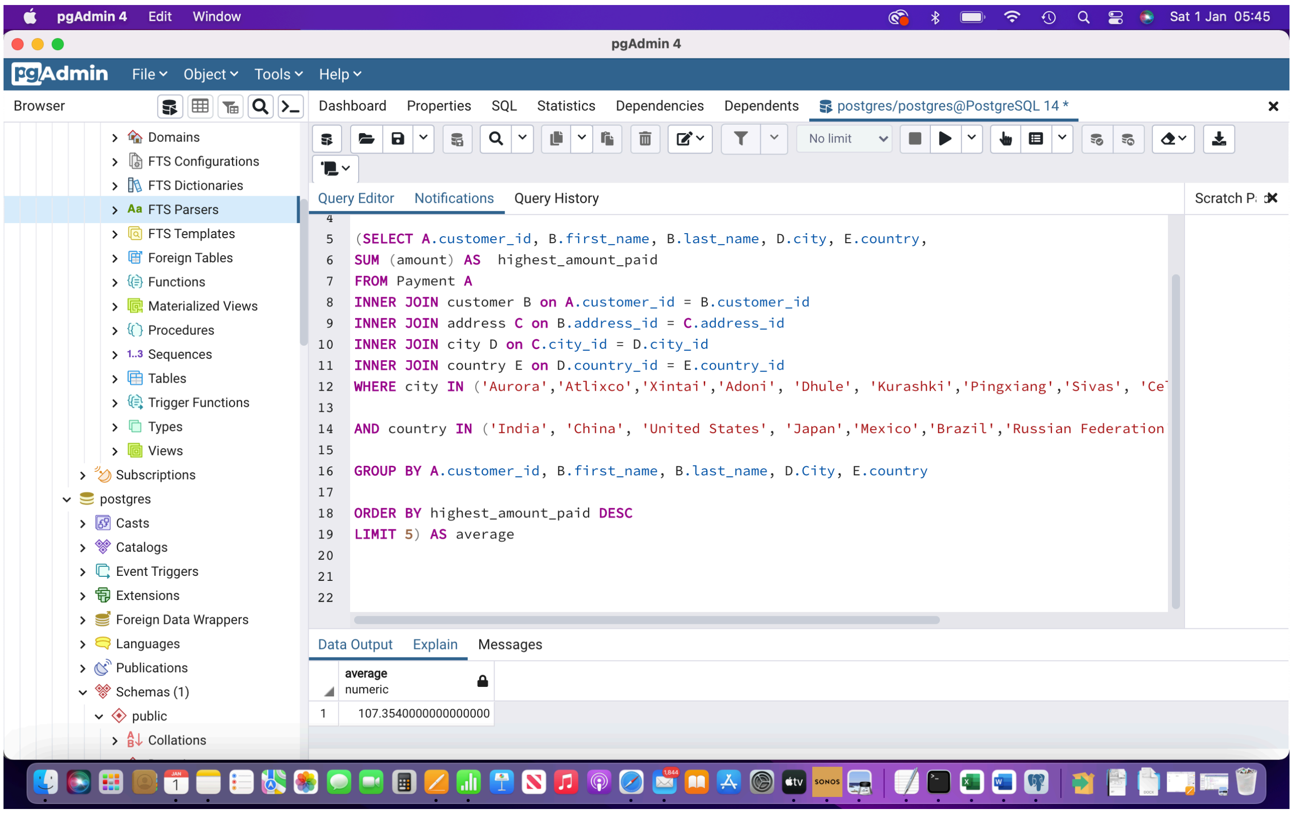
EXERCISE 3.8

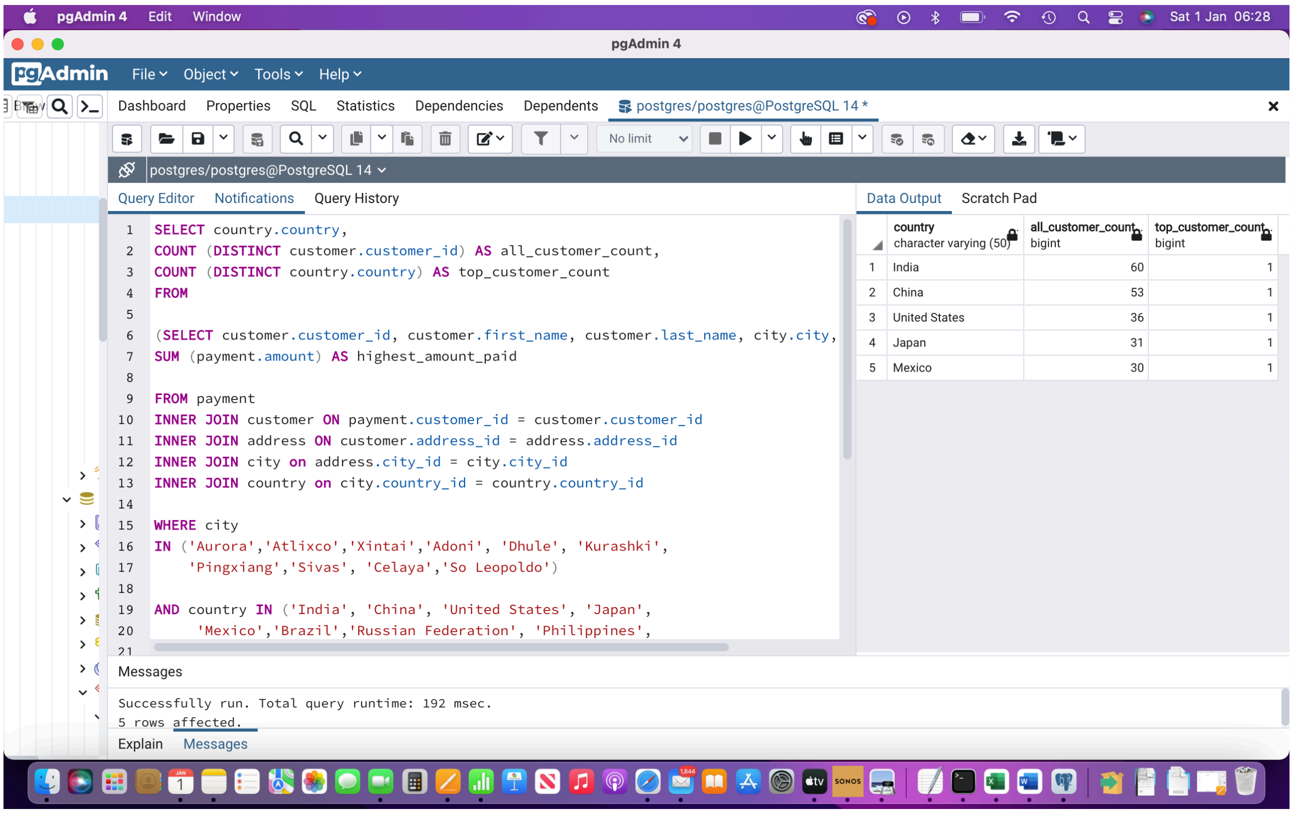
Part 1.



```
(SELECT A.customer_id, B.first_name, B.last_name, D.city, E.country,
SUM (amount) AS  highest_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', 'Kurashki','Pingxiang','Sivas', 'Ce

AND country IN ('India', 'China', 'United States', 'Japan','Mexico','Brazil','Russian Federation

GROUP BY A.customer_id, B.first_name, B.last_name, D.City, E.country

ORDER BY highest_amount_paid DESC
LIMIT 5) AS average
```

	average numeric
1	107.3540000000000000

Part 2. (a) - ***OUTPUT is on the right hand side and not bottom due to size of SYNTAX.

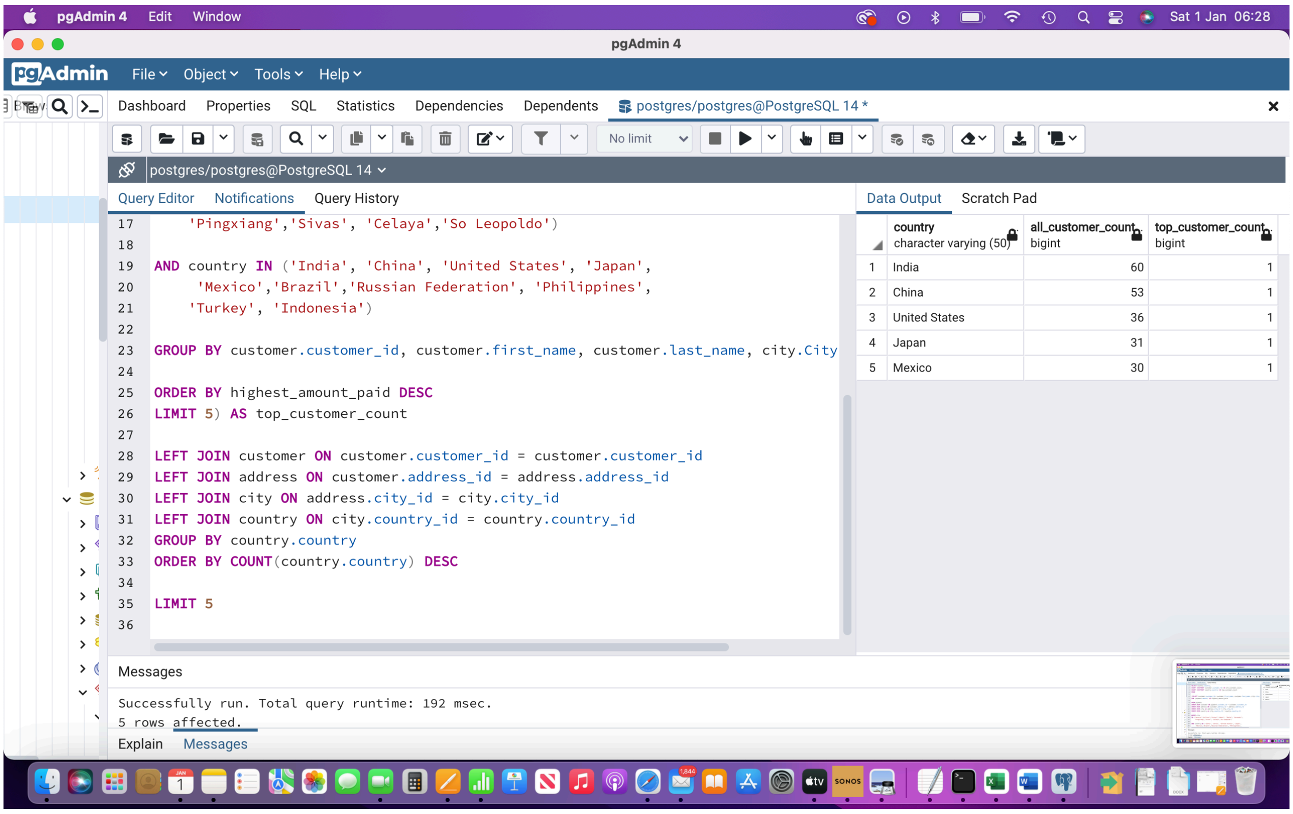


```
SELECT country.country,
COUNT (DISTINCT customer.customer_id) AS all_customer_count,
COUNT (DISTINCT country.country) AS top_customer_count
FROM

(SELECT customer.customer_id, customer.first_name, customer.last_name, city.city,
SUM (payment.amount) AS highest_amount_paid

FROM payment
INNER JOIN customer ON payment.customer_id = customer.customer_id
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

WHERE city
IN ('Aurora','Atlixco','Xintai','Adoni', 'Dhule', 'Kurashki',
    'Pingxiang','Sivas', 'Celaya','So Leopoldo')

AND country IN ('India', 'China', 'United States', 'Japan',
     'Mexico','Brazil','Russian Federation', 'Philippines',
```

	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

```
Successfully run. Total query runtime: 192 msec.
5 rows affected.
```

Part 2 (b)



```
    'Pingxiang','Sivas', 'Celaya','So Leopoldo')

AND country IN ('India', 'China', 'United States', 'Japan',
     'Mexico','Brazil','Russian Federation', 'Philippines',
    'Turkey', 'Indonesia')

GROUP BY customer.customer_id, customer.first_name, customer.last_name, city.City

ORDER BY highest_amount_paid DESC
LIMIT 5) AS top_customer_count

LEFT JOIN customer ON customer.customer_id = customer.customer_id
LEFT JOIN address ON customer.address_id = address.address_id
LEFT JOIN city ON address.city_id = city.city_id
LEFT JOIN country ON city.country_id = country.country_id
GROUP BY country.country
ORDER BY COUNT(country.country) 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

Messages

```
Successfully run. Total query runtime: 192 msec.
5 rows affected.
```

Part 3a)

Step 1 could have been achieved without a subquery.

Step 2 required me to combine related information from different tables and therefore would be efficient.

Part 3 b)

Subqueries are beneficial in dividing complex queries into logical steps and have the added benefit of being efficient .