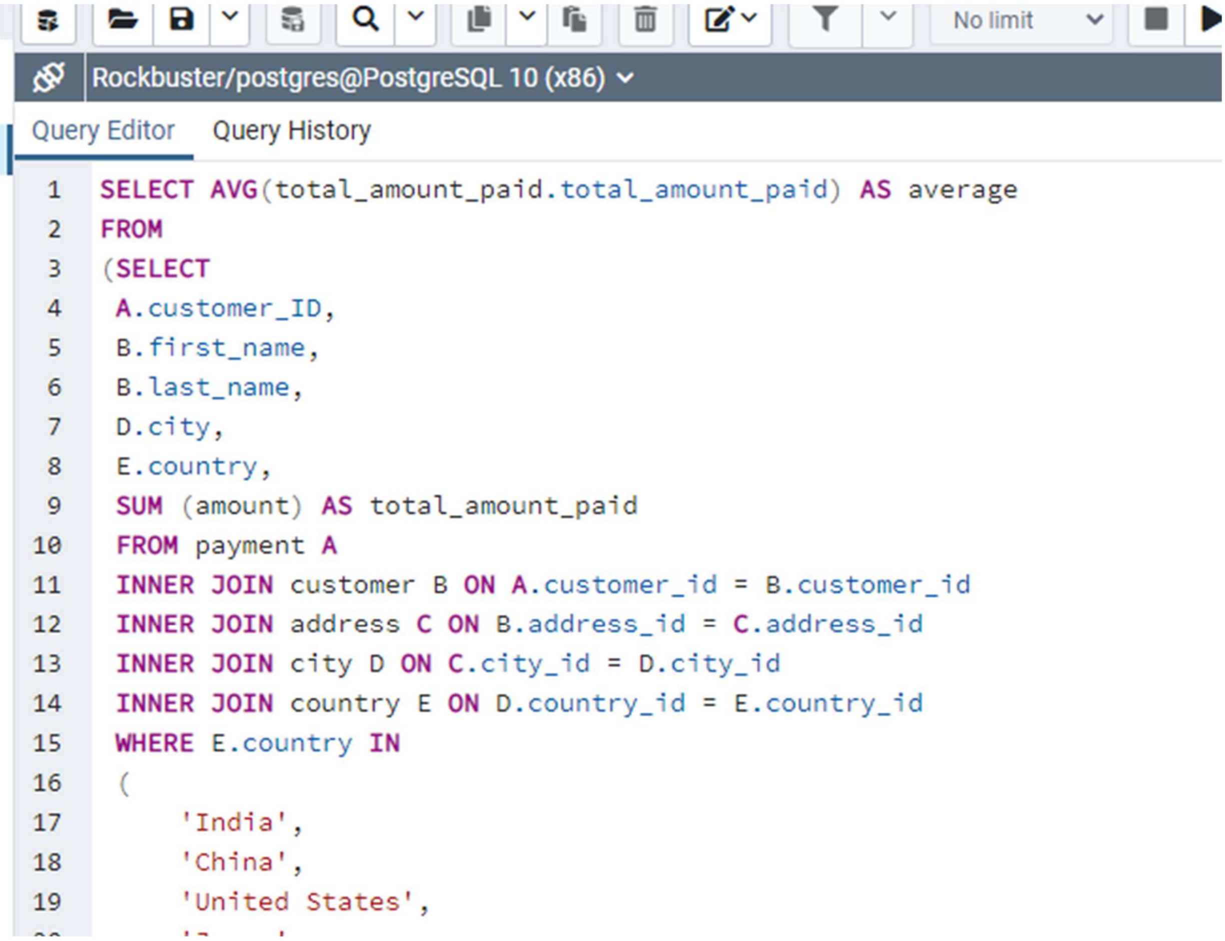
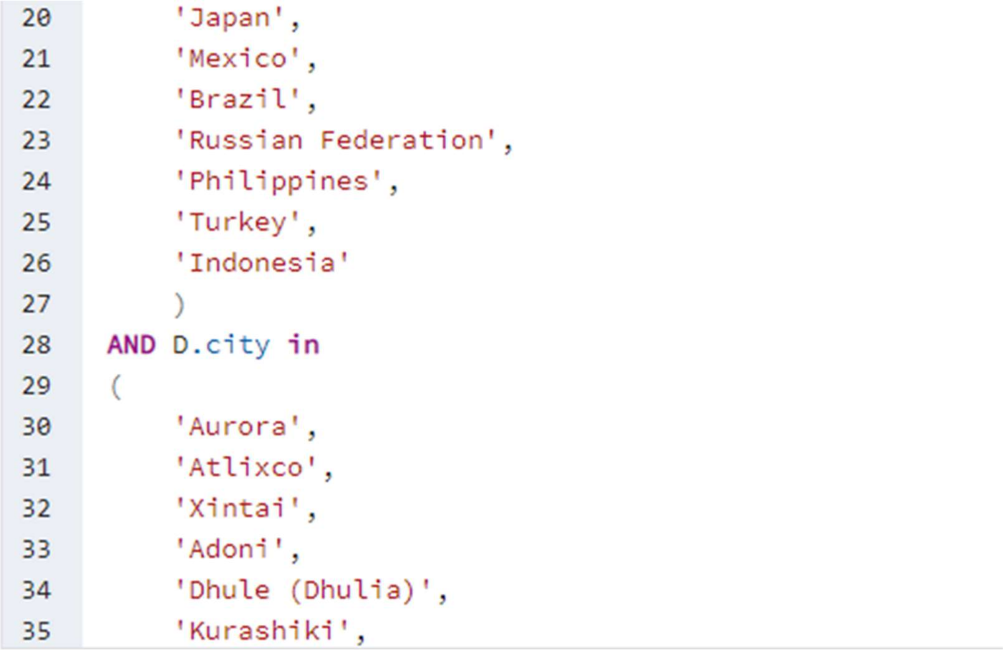
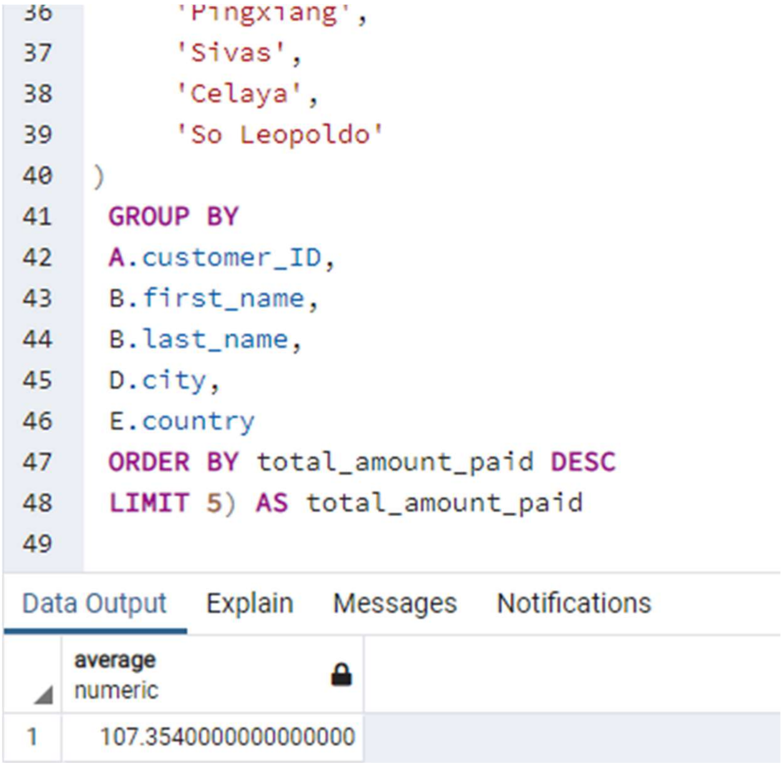
Performing Subqueries



```
SELECT AVG(total_amount_paid.total_amount_paid) AS average
FROM
(SELECT
 A.customer_ID,
 B.first_name,
 B.last_name,
 D.city,
 E.country,
 SUM (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 E.country IN
 (
     'India',
     'China',
     'United States',
```



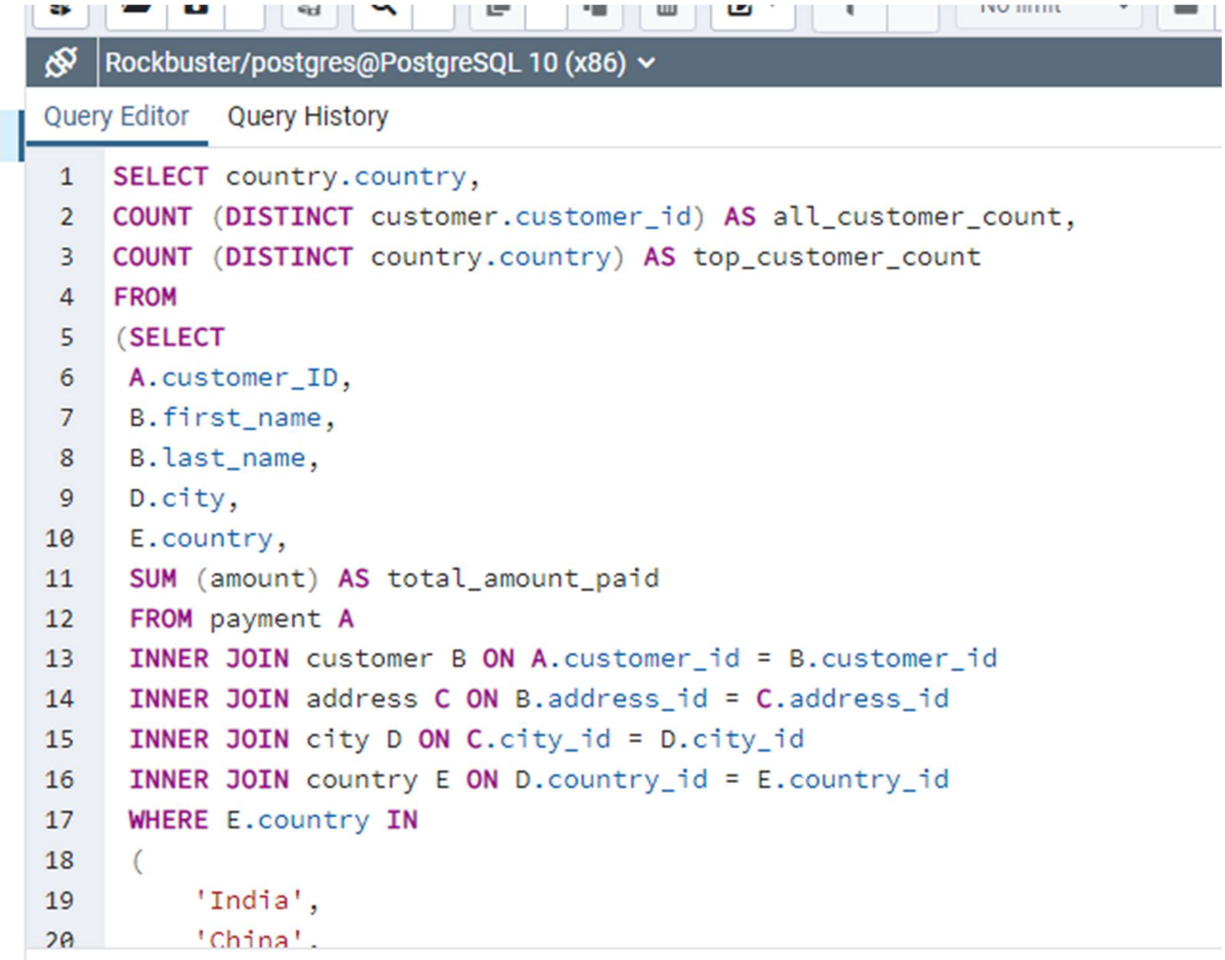
```
     'Japan',
     'Mexico',
     'Brazil',
     'Russian Federation',
     'Philippines',
     'Turkey',
     'Indonesia'
     )
 AND D.city in
 (
     'Aurora',
     'Atlixco',
     'Xintai',
     'Adoni',
     'Dhule (Dhulia)',
     'Kurashiki',
```



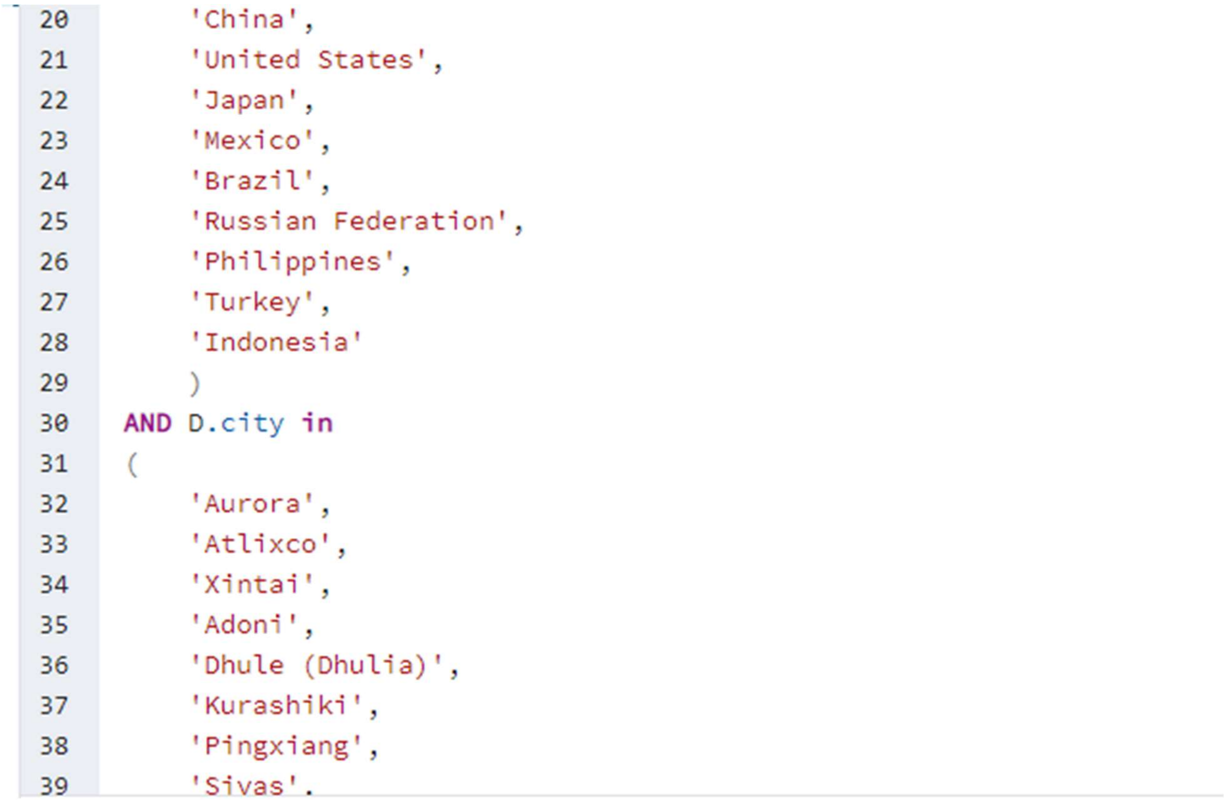
```
     'Pingxiang',
     'Sivas',
     'Celaya',
     'So Leopoldo'
)
 GROUP BY
 A.customer_ID,
 B.first_name,
 B.last_name,
 D.city,
 E.country
 ORDER BY total_amount_paid DESC
 LIMIT 5) AS total_amount_paid

```

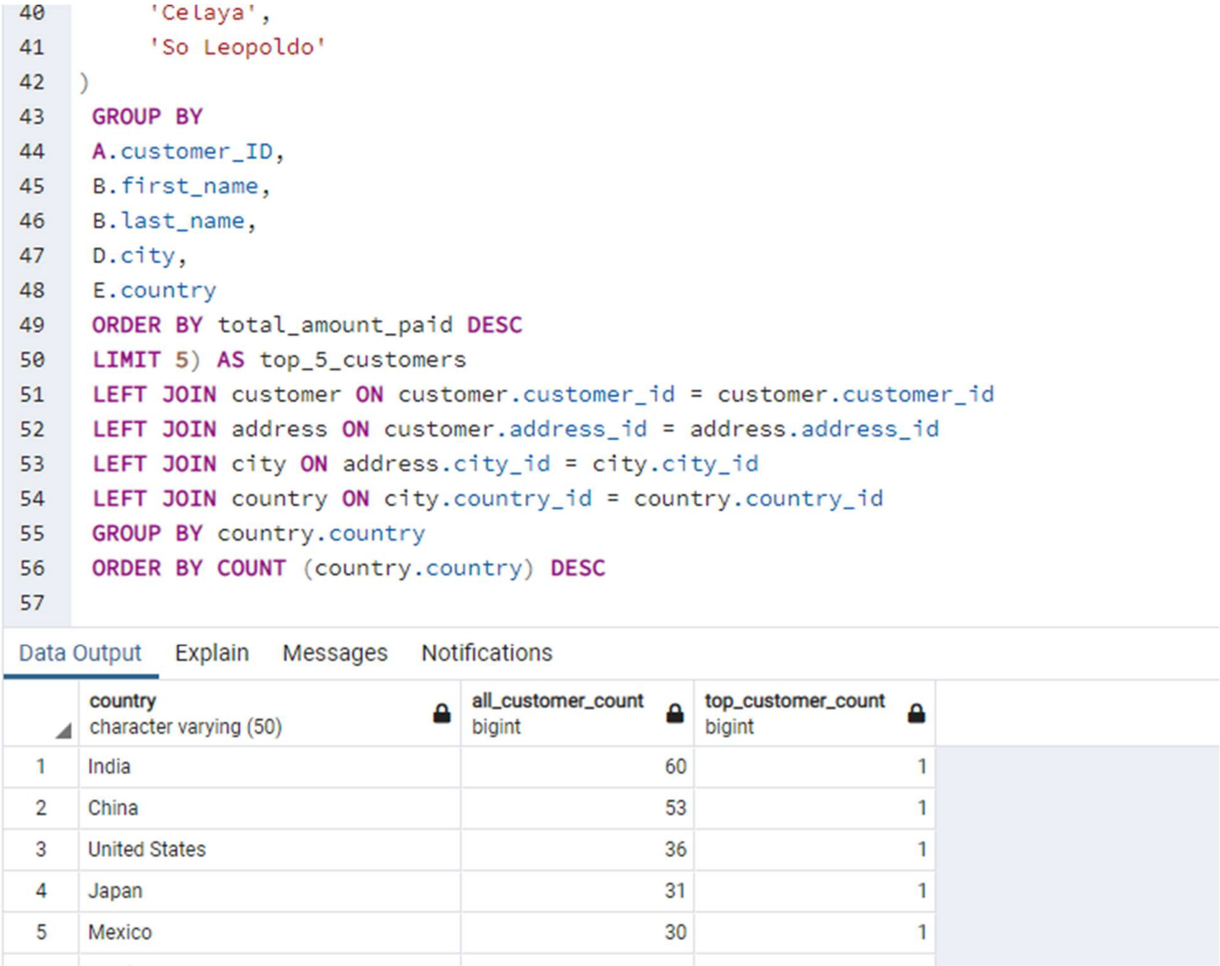
```
    average
    numeric
1   107.3540000000000000
```



```
SELECT country.country,
COUNT (DISTINCT customer.customer_id) AS all_customer_count,
COUNT (DISTINCT country.country) AS top_customer_count
FROM
(SELECT
 A.customer_ID,
 B.first_name,
 B.last_name,
 D.city,
 E.country,
 SUM (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 E.country IN
 (
     'India',
     'China',
```



```
    'China',
    'United States',
    'Japan',
    'Mexico',
    'Brazil',
    'Russian Federation',
    'Philippines',
    'Turkey',
    'Indonesia'
    )
AND D.city in
(
    'Aurora',
    'Atlixco',
    'Xintai',
    'Adoni',
    'Dhule (Dhulia)',
    'Kurashiki',
    'Pingxiang',
    'Sivas',
```



```
         'Celaya',
         'So Leopoldo'
)
 GROUP BY
 A.customer_ID,
 B.first_name,
 B.last_name,
 D.city,
 E.country
 ORDER BY total_amount_paid DESC
 LIMIT 5) AS top_5_customers
 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
```

	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 can be done without a subquery function whilst with Step 2 we will need a subquery because we have to compare other tables.
- Subqueries are useful for comparing different tables.