July 11, 2022

Data Immersion
Databases & SQL for Analysts
3.8: Performing Subqueries
David Guillen Aroche

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

1. **Copy query wrote on step 3 from task 3.7 (Joining Tables of Data) into the query tool; this will be the subquery, with *"total_amount_paid"* as its alias.**

```
Inner Query (Inner Statement):
SELECT A.customer_id,
      A.first_name,
      A.last_name,
      C.city,
      D.country,
      SUM(E.amount) AS payments_total_amount
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',
                  'Atlixco',
                  'Emeishan',
                  'Pontianak',
                  'Shimoga',
                  'Aparecida de Goinia',
                  'Zalantun',
                  'Taguig')
AND country IN('India',
                  'China',
                  'United States',
                  'Japan',
                  'Mexico',
                  'Brazil',
                  'Russian Federation',
                  'Philippines',
                  'Turkey',
                  'Indonesia')
GROUP BY (A.customer_id, C.city, D.country)
```

```
ORDER BY payments_total_amount DESC
LIMIT 5
```

Inner S. within Outer S.:

```
SELECT AVG(total_amount_paid.total_amount_paid) AS average
FROM (SELECT A.customer_id,
             A.first_name,
             A.last_name,
             C.city,
             D.country,
             SUM(E.amount) AS payments_total_amount
             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',
                            'Atlixco',
                            'Emeishan',
                            'Pontianak',
                            'Shimoga',
                            'Aparecida de Goinia',
                            'Zalantun',
                            'Taguig')
              AND country IN('India',
                             'China',
                             'United States',
                              'Japan',
                              'Mexico',
                              'Brazil',
                              'Russian Federation',
                              'Philippines',
                              'Turkey',
                              'Indonesia')
             GROUP BY (A.customer_id, C.city, D.country)
             ORDER BY payments_total_amount DESC
        LIMIT 5) AS total_amount_paid
```

```sql
SELECT AVG(total_amount_paid.total_amount_paid) AS average
FROM (SELECT A.customer_id,
             A.first_name,
             A.last_name,
             C.city,
             D.country,
             SUM(E.amount) AS total_amount_paid
      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',
                     'Atlixco',
                     'Emeishan',
                     'Pontianak',
                     'Shimoga',
                     'Aparecida de Goinia',
                     'Zalantun',
                     'Taguig')
      AND country IN('India',
                     'China',
                     'United States',
                     'Japan',
                     'Mexico',
                     'Brazil',
                     'Russian Federation',
```

Data output | Messages | Notifications

	average numeric
1	120.322

Total rows: 1 of 1 | Query complete 00:00:00.111

2. **Find out how many of the Top 5 Customers are vases within each country. fInal output should include 3 columns: *"country"*, *"all_customer_count"*, and *"top_customer_count"***

Outer Query (counting the number of customers living in each country):

```
SELECT DISTINCT  D.country,
    COUNT(DISTINCT customer_id) AS all_customer_count
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
GROUP BY D.country
ORDER BY all_customer_count DESC
```

```
645  --2.-Write outer statement containing the number od custo
646  SELECT DISTINCT  D.country,
647         COUNT(DISTINCT customer_id) AS all_customer_count
648  FROM customer A
649  INNER JOIN address B ON A.address_id = B.address_id
650  INNER JOIN city C ON B.city_id = C.city_id
651  INNER JOIN country D ON C.country_id = D.country_id
652  GROUP BY D.country
653  ORDER BY all_customer_count DESC
654
```

Data output | Messages | Notifications

	country character varying (50)	all_customer_count bigint
1	India	60
2	China	53
3	United States	36
4	Japan	31
5	Mexico	30
6	Brazil	28
7	Russian Federation	28
8	Philippines	20
9	Turkey	15
10	Indonesia	14
11	Argentina	13
12	Nigeria	13
13	South Africa	11
14	Taiwan	10
15	United Kingdom	9
16	Iran	8

Total rows: 108 of 108 Query complete 00:00:00.082

```
SELECT DISTINCT  D.country,
    COUNT(DISTINCT A.customer_id) AS all_customer_count,
    COUNT(DISTINCT D.country) AS top_customer_count
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
LEFT JOIN (SELECT A.customer_id,
```

```
                A.first_name,
                A.last_name,
                C.city,
                D.country,
                SUM(E.amount) AS payments_total_amount
                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',
                               'Atlixco',
                               'Emeishan',
                               'Pontianak',
                               'Shimoga',
                               'Aparecida de Goinia',
                               'Zalantun',
                               'Taguig')
                AND country IN('India',
                                'China',
                                'United States',
                                'Japan',
                                'Mexico',
                                'Brazil',
                                'Russian Federation',
                                'Philippines',
                                'Turkey',
                                'Indonesia')
                GROUP BY (A.customer_id, C.city, D.country)
                ORDER BY payments_total_amount DESC
                LIMIT 5) AS total_amount_paid ON D.country = total_amount_paid.country
GROUP BY D.country
ORDER BY all_customer_count DESC
LIMIT 10
```

```
747                 'Zalantun',
748                 'Taguig')
749 AND country IN('India',
750                 'China',
751                 'United States',
752                 'Japan',
753                 'Mexico',
754                 'Brazil',
755                 'Russian Federation',
756                 'Philippines',
757                 'Turkey',
758                 'Indonesia')
759 GROUP BY (A.customer_id, C.city, D.country)
760 ORDER BY payments_total_amount DESC
761 LIMIT 5) AS total_amount_paid ON D.country =total_amount_paid.country
762 GROUP BY D.country
763 ORDER BY all_customer_count DESC
764 LIMIT 10
```

Data output | Messages | Notifications

	country character varying (50)	all_customer_count bigint	top_customer_count bigint
2	China	53	1
3	United States	36	1
4	Japan	31	1
5	Mexico	30	1
6	Brazil	28	1
7	Russian Federation	28	1
8	Philippines	20	1
9	Turkey	15	1
10	Indonesia	14	1

Total rows: 10 of 10 | Query complete 00:00:00.117

3. **Comment on the following:**
    - **Could we have achieved results without using subqueries in both steps?**

In **STEP 1** we could have had results without using a query, either specifying the IDs from customers with WHERE or using HAVING and specifying the values.

In **STEP 2** it was necessary to have/use the generated table total_amount_paid, not possible to retrieve any results if that subquery element is not present.

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

When we need to determine several different sets of data points (or single data points) that are within the database but is not necessarily explicit, or when we need to perform operations or other functions that need information that has to be processes or calculated first. In general, when we have to answer several questions which content or information is not necessarily related and/or needs to be factorized in order to use it with other set of columns/tables.