Databases & SQL for Analysts

3.7: Joining Tables of Data

1. Top 10 Countries for Rockbuster in Terms of Customer Numbers

- **Query & answer:**

```
1.  -- Write a query to find the top 10 countries in terms of customer numbers
2.  -- customer = A, address = B, city = C, country = D
3.
4.  SELECT D.country,
5.         COUNT(customer_id) AS customer_count
6.  FROM public.customer A
7.  LEFT JOIN public.address B ON A.address_id = B.address_id
8.  LEFT JOIN public.city C ON B.city_id = C.city_id
9.  LEFT JOIN public.country D ON C.country_id = D.country_id
10. GROUP BY country
11. ORDER BY customer_count DESC
12. LIMIT 10
```

country	customer_count
India	60
China	53
United States	36
Japan	31
Mexico	30
Brazil	28
Russian Federation	28
Philippines	20
Turkey	15
Indonesia	14

- **Thought process:**

This task requires the use of the *country* and *customer* tables. According to the ERD diagram, a relationship chain is established among the four tables:

customer -> address -> city -> country

The SELECT statement, therefore, should include a column for the country name and another for the customer count.

To uncover the correlation between **customer numbers** and **countries**:

- LEFT JOIN is used with the *customer* and *address* tables, as all customers are considered.
- LEFT JOIN is used with the address and *city* tables, as only addresses of the customers from the *customer* table are considered.
- LEFT JOIN is used with the *city* and *country* tables, as only cities of the customers from the *customer* table are considered.

After joining the tables, the results should be grouped by country, arranged in a descending order and limited to 10 results. The query yields the top 10 countries in terms of customer count.

2. Top 10 Cities Within the Top 10 Countries

- **Query & answer:**

```
1.  -- Write a query to find the top 10 cities within the top 10 countries
2.  -- customer = A, address = B, city = C, country = D
3.
4.  SELECT C.city,
5.         D.country,
6.         COUNT(A.customer_id) AS customer_count
7.  FROM public.customer A
8.  LEFT JOIN public.address B ON A.address_id = B.address_id
9.  LEFT JOIN public.city C ON B.city_id = C.city_id
10. LEFT JOIN public.country D ON C.country_id = D.country_id
11.            WHERE D.country IN (
12.                                SELECT D.country
13.                                FROM public.customer A
```

```
                                    LEFT JOIN public.address B
                                              ON A.address_id = B.address_id
                                    LEFT JOIN public.city C
                                              ON B.city_id = C.city_id
                                    LEFT JOIN public.country D
                                              ON C.country_id = D.country_id
                                    GROUP BY country
                                    ORDER BY COUNT(customer_id) DESC
                                    LIMIT 10
                                    )
GROUP BY C.city, D.country
ORDER BY customer_count DESC
LIMIT 10
```

city	country	customer_count
Aurora	United States	2
Atlixco	Mexico	1
Xintai	China	1
Adoni	India	1
Dhule (Dhulia)	India	1
Kurashiki	Japan	1
Pingxiang	China	1
Sivas	Turkey	1
Celaya	Mexico	1
So Leopoldo	Brazil	1

- **Thought process:**

This task requires the use of the *city*, *country* and *customer* tables. As in the previous task, a relationship chain is established among the four tables:

customer -> address -> city -> country

As a result, the SELECT statement should include a column for the city name, another for the nation name, and yet another for the customer count.

To uncover the correlation between **customer numbers** and the **cities** within the top 10 countries:

- LEFT JOIN is used with the *customer* and *address* tables, as all customers are considered.
- LEFT JOIN is used with the *address* and *city* tables, as only addresses of the customers from the *customer* table are considered.
- LEFT JOIN is used with the *city* and *country* tables, as only cities of the customers from the *customer* table are considered.
- A WHERE filter is used, which includes the SQL query for the top 10 countries.

After joining the tables, the results should be grouped by city and country, arranged in a descending order and limited to 10 results. The query yields the top 10 cities within the top 10 countries in terms of customer count.

Note:

When the LIMIT 10 is removed, the SQL query produces a list of 314 records, with all cities having a customer count of 1, apart from Aurora, United States, which has 2 customers. This implies that there are only two possibilities regarding customer counts: cities with one customer and cities with two.

Consequently, the list above is not the only accurate depiction of the top ten cities.

3. Top 5 Customers in the Top 10 Cities Who Have Paid the Highest Total Amounts

- **Query & answer:**

```
1. -- Write a query to find the top 5 customers in the top 10 cities
2. -- who have paid the highest total amount
3. -- customer = A, address = B, city = C, country = D, payment = E
4.
5. SELECT A.customer_id, A.first_name, A.last_name,
6.        D.country,
7.        C.city,
8.        SUM(E.amount) AS total_amount_paid
9. FROM public.payment E
10. RIGHT JOIN public.customer A ON E.customer_id = A.customer_id
11. LEFT JOIN public.address B ON A.address_id = B.address_id
```

```
LEFT JOIN public.city C ON B.city_id = C.city_id
LEFT JOIN public.country D ON C.country_id = D.country_id
WHERE C.city IN (
                SELECT C.city
                FROM public.city C
                RIGHT JOIN public.address B
                           ON C.city_id = B.city_id
                LEFT JOIN public.country D
                           ON C.country_id = D.country_id
                RIGHT JOIN public.customer A
                           ON B.address_id = A.address_id
                WHERE D.country IN (
                                    SELECT D.country
                                    FROM public.customer A
                                    LEFT JOIN public.address B
                                              ON A.address_id = B.address_id
                                    LEFT JOIN public.city C
                                              ON B.city_id = C.city_id
                                    LEFT JOIN public.country D
                                              ON C.country_id = D.country_id
                                    GROUP BY country
                                    ORDER BY COUNT(customer_id) DESC
                                    LIMIT 10
                                    )
                GROUP BY C.city, D.country
                ORDER BY COUNT(customer_id) DESC
                LIMIT 10
                )
GROUP BY A.customer_id, A.first_name, A.last_name,
         D.country,
         C.city
ORDER BY total_amount_paid DESC
LIMIT 5
```

customer_id	first_name	last_name	country	city	total_amount_paid
84	Sara	Perry	Mexico	Atlixco	128.70
518	Gabriel	Harder	Turkey	Sivas	108.75
587	Sergio	Stanfield	Mexico	Celaya	102.76
537	Clinton	Buford	United States	Aurora	98.76
367	Adam	Gooch	India	Adoni	97.80

- **Thought process:**

This task requires the use of the *payment*, *city*, and *customer* tables. In addition to the relationship chain from the previous task, another node is added:

payment -> customer -> address -> city -> country

According to the task requirement, the SELECT statement includes 6 columns for: customer ID, customer first name and last name, country, city, total amount paid.

To uncover the correlation among **customer**, their **payment amount** and their location, including **city** and **country**:

- RIGHT JOIN is used with the *payment* and *customer* tables, as all customers are considered.
- LEFT JOIN is used with the *customer* and *address* tables, for similar reason as above.
- LEFT JOIN is used with the *address* and *city* tables, as only addresses of the customers from the *customer* table are considered.
- LEFT JOIN is used with the *city* and *country* tables, as only cities of the customers from the *customer* table are considered.
- A WHERE filter is used, which includes the SQL query for the top 10 cities:
  - RIGHT JOIN is used with the *city* and *address* tables.
  - LEFT JOIN is used with the *city* and *country* tables.
  - RIGHT JOIN is used with the *address* and *customer* tables.
  - A WHERE filter is used, which includes the SQL query for the top 10 countries.

After joining the tables, the results should be grouped by customer ID, first name, last name, country and city, arranged in a descending order and limited to 5 results. The query yields the top 5 paying customers from the top 10 cities.

Note:

This list of top 5 customers is not the only accurate representation of the highest paying customers, just as the list in the previous task was not the only accurate depiction of the top 10 cities.