DESCRIPTION

Table: Transactions

+---------------+---------+

| Column Name | Type |

+---------------+---------+

| id | int |

| country | varchar |

| state | enum |

| amount | int |

| trans_date | date |

+---------------+---------+

id is the primary key of this table.

The table has information about incoming transactions.

The state column is an enum of type ["approved", "declined"].

Write an SQL query to find for each month and country, the number of transactions and their total amount, the number of approved transactions and their total amount.

Return the result table in **any order**.

The query result format is in the following example.

Example 1:

Input:

Transactions table:

+------+---------+----------+--------+------------+

| id | country | state | amount | trans_date |

+------+---------+----------+--------+------------+

| 121 | US | approved | 1000 | 2018-12-18 |

| 122 | US | declined | 2000 | 2018-12-19 |

| 123 | US | approved | 2000 | 2019-01-01 |

| 124 | DE | approved | 2000 | 2019-01-07 |

+------+---------+----------+--------+------------+

Output:

+----------+---------+-------------+---------------+-------------------+----------------------+

| month | country | trans_count | approved_count | trans_total_amount | approved_total_amount |

+----------+---------+-------------+---------------+-------------------+----------------------+

| 2018-12 | US | 2 | 1 | 3000 | 1000 |

| 2019-01 | US | 1 | 1 | 2000 | 2000 |

| 2019-01 | DE | 1 | 1 | 2000 | 2000 |

+----------+---------+-------------+---------------+-------------------+----------------------+

SOLUTION

MySQL:

- Select query_name, calculate quality using AVG(), and round the result to 2 decimals using ROUND()
- Calculate poor_query_percentage using IF() (if rating is less than 3, then 1, else 0), add up using SUM(), and round the result to 2 decimals using ROUND()
- GROUP BY query_name

PostgreSQL:

- Select query_name, calculate quality using SUM() and COUNT(), and round the result to 2 decimals using ROUND()
- Calculate poor_query_percentage using CASE (when rating is less than 3, then 1, else 0), add up using SUM(), and round the result to 2 decimals using ROUND()
- GROUP BY query_name

```
SELECT TO_CHAR(trans_date, 'yyyy-mm') AS month, country, COUNT(*) trans_count, COUNT(CASE WHEN state
= 'approved' THEN 1 END) AS approved_count, SUM(amount) AS trans_total_amount, SUM(CASE WHEN state =
'approved' THEN amount ELSE 0 END) AS approved_total_amount
FROM Transactions
GROUP BY 1, 2;
```