Problem 2701: Consecutive Transactions with Increasing Amounts

Problem Information

Difficulty: Hard
Acceptance Rate: 0.00%
Paid Only: No

Problem Description

Table:

Transactions

+------------------+------+ | Column Name | Type | +------------------+------+ | transaction_id | int | | customer_id | int | | transaction_date | date | | amount | int | +------------------+------+ transaction_id is the primary key of this table. Each row contains information about transactions that includes unique (customer_id, transaction_date) along with the corresponding customer_id and amount.

Write an SQL query to find the customers who have made consecutive transactions with increasing

amount

for at least three consecutive days. Include the

customer_id

, start date of the consecutive transactions period and the end date of the consecutive transactions period. There can be multiple consecutive transactions by a customer.

Return

the result table ordered by

customer_id, consecutive_start, consecutive_end

in

ascending

order.

The query result format is in the following example.

Example 1:

Input:

Transactions table: +----------------+-------------+------------------+--------+ | transaction_id | customer_id | transaction_date | amount | +----------------+-------------+------------------+--------+ | 1 | 101 | 2023-05-01 | 100 | | 2 | 101 | 2023-05-02 | 150 | | 3 | 101 | 2023-05-03 | 200 | | 4 | 102 | 2023-05-01 | 50 | | 5 | 102 | 2023-05-03 | 100 | | 6 | 102 | 2023-05-04 | 200 | | 7 | 105 | 2023-05-01 | 100 | | 8 | 105 | 2023-05-02 | 150 | | 9 | 105 | 2023-05-03 | 200 | | 10 | 105 | 2023-05-04 | 300 | | 11 | 105 | 2023-05-12 | 250 | | 12 | 105 | 2023-05-13 | 260 | | 13 | 105 | 2023-05-14 | 270 | +----------------+-------------+------------------+--------+

Output:

+-------------+------------------+-----------------+ | customer_id | consecutive_start | consecutive_end | +-------------+------------------+-----------------+ | 101 | 2023-05-01 | 2023-05-03 | | 105 | 2023-05-01 | 2023-05-04 | | 105 | 2023-05-12 | 2023-05-14 | +-------------+------------------+-----------------+

Explanation:

- customer_id 101 has made consecutive transactions with increasing amounts from May 1st, 2023, to May 3rd, 2023 - customer_id 102 does not have any consecutive transactions for at least 3 days. - customer_id 105 has two sets of consecutive transactions: from May 1st, 2023, to May 4th, 2023, and from May 12th, 2023, to May 14th, 2023. customer_id is sorted in ascending order.

Code Snippets

MySQL:

```
# Write your MySQL query statement below
```

MS SQL Server:

```
/* Write your T-SQL query statement below */
```

PostgreSQL:

```
-- Write your PostgreSQL query statement below
```

Oracle:

```
/* Write your PL/SQL query statement below */
```

Pandas:

```
import pandas as pd

def consecutive_increasing_transactions(transactions: pd.DataFrame) ->
pd.DataFrame:
```

Solutions

MySQL Solution:

```
# Write your MySQL query statement below
```

MS SQL Server Solution:

```
/* Write your T-SQL query statement below */
```

PostgreSQL Solution:

```
-- Write your PostgreSQL query statement below
```

Oracle Solution:

```
/* Write your PL/SQL query statement below */
```

Pandas Solution:

```
import pandas as pd

def consecutive_increasing_transactions(transactions: pd.DataFrame) ->
pd.DataFrame:
```