Problem 2084: Drop Type 1 Orders for Customers With Type 0 Orders

Problem Information

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

Problem Description

Table:

Orders

+-------------+------+ | Column Name | Type | +-------------+------+ | order_id | int | | customer_id | int | | order_type | int | +-------------+------+ order_id is the column with unique values for this table. Each row of this table indicates the ID of an order, the ID of the customer who ordered it, and the order type. The orders could be of type 0 or type 1.

Write a solution to report all the orders based on the following criteria:

If a customer has

at least one

order of type

0

, do

not

report any order of type

1

from that customer.

Otherwise, report all the orders of the customer.

Return the result table in

any order

.

The result format is in the following example.

Example 1:

Input:

Orders table: +----------+-------------+------------+ | order_id | customer_id | order_type | +----------+-------------+------------+ | 1 | 1 | 0 | | 2 | 1 | 0 | | 11 | 2 | 0 | | 12 | 2 | 1 | | 21 | 3 | 1 | | 22 | 3 | 0 | | 31 | 4 | 1 | | 32 | 4 | 1 | +----------+-------------+------------+

Output:

+----------+-------------+------------+ | order_id | customer_id | order_type | +----------+-------------+------------+ | 31 | 4 | 1 | | 32 | 4 | 1 | | 1 | 1 | 0 | | 2 | 1 | 0 | | 11 | 2 | 0 | | 22 | 3 | 0 | +----------+-------------+------------+

Explanation:

Customer 1 has two orders of type 0. We return both of them. Customer 2 has one order of type 0 and one order of type 1. We only return the order of type 0. Customer 3 has one order of type 0 and one order of type 1. We only return the order of type 0. Customer 4 has two orders of type 1. We return both of them.

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 drop_specific_orders(orders: 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 drop_specific_orders(orders: pd.DataFrame) -> pd.DataFrame:
```