Problem 1596: The Most Frequently Ordered Products for Each Customer

Problem Information

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

Problem Description

Table:

Customers

+---------------+---------+ | Column Name | Type | +---------------+---------+ | customer_id | int | | name | varchar | +---------------+---------+ customer_id is the column with unique values for this table. This table contains information about the customers.

Table:

Orders

+---------------+---------+ | Column Name | Type | +---------------+---------+ | order_id | int | | order_date | date | | customer_id | int | | product_id | int | +---------------+---------+ order_id is the column with unique values for this table. This table contains information about the orders made by customer_id. No customer will order the same product

more than once

in a single day.

Table:

Products

+---------------+---------+ | Column Name | Type | +---------------+---------+ | product_id | int | | product_name | varchar | | price | int | +---------------+---------+ product_id is the column with unique values for this table. This table contains information about the products.

Write a solution to find the most frequently ordered product(s) for each customer.

The result table should have the

product_id

and

product_name

for each

customer_id

who ordered at least one order.

Return the result table in

any order

.

The result format is in the following example.

Example 1:

Input:

Customers table: +-------------+-------+ | customer_id | name | +-------------+-------+ | 1 | Alice | | 2 | Bob | | 3 | Tom | | 4 | Jerry | | 5 | John | +-------------+-------+ Orders table: +----------+------------+-------------+------------+ | order_id | order_date | customer_id | product_id | +----------+------------+-------------+------------+ | 1 | 2020-07-31 | 1 | 1 | | 2 | 2020-07-30 | 2 | 2 | | 3 | 2020-08-29 | 3 | 3 | | 4 | 2020-07-29 | 4 | 1 | | 5 | 2020-06-10 | 1 | 2 | | 6 | 2020-08-01 | 2 | 1 | | 7 | 2020-08-01 | 3 | 3 | | 8 | 2020-08-03 | 1 | 2 | | 9 | 2020-08-07 | 2 | 3 | | 10 | 2020-07-15 | 1 | 2 | +----------+------------+-------------+------------+ Products table: +------------+--------------+-------+ | product_id | product_name | price | +------------+--------------+-------+ | 1 | keyboard | 120 | | 2 |

mouse | 80 | | 3 | screen | 600 | | 4 | hard disk | 450 | +------------+--------------+-------+

Output:

+-------------+------------+--------------+ | customer_id | product_id | product_name | +-------------+------------+--------------+ | 1 | 2 | mouse | | 2 | 1 | keyboard | | 2 | 2 | mouse | | 2 | 3 | screen | | 3 | 3 | screen | | 4 | 1 | keyboard | +-------------+------------+--------------+

Explanation:

Alice (customer 1) ordered the mouse three times and the keyboard one time, so the mouse is the most frequently ordered product for them. Bob (customer 2) ordered the keyboard, the mouse, and the screen one time, so those are the most frequently ordered products for them. Tom (customer 3) only ordered the screen (two times), so that is the most frequently ordered product for them. Jerry (customer 4) only ordered the keyboard (one time), so that is the most frequently ordered product for them. John (customer 5) did not order anything, so we do not include them in the result table.

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 most_frequently_products(customers: pd.DataFrame, orders: pd.DataFrame,
```

```
products: 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 most_frequently_products(customers: pd.DataFrame, orders: pd.DataFrame,
products: pd.DataFrame) -> pd.DataFrame:
```