Problem 2292: Products With Three or More Orders in Two Consecutive Years

Problem Information

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

Problem Description

Table:

Orders

+---------------+------+ | Column Name | Type | +---------------+------+ | order_id | int | | product_id | int | | quantity | int | | purchase_date | date | +---------------+------+ order_id contains unique values. Each row in this table contains the ID of an order, the id of the product purchased, the quantity, and the purchase date.

Write a solution to report the IDs of all the products that were ordered three or more times in two consecutive years.

Return the result table in

any order

.

The result format is shown in the following example.

Example 1:

Input:

Orders table: +----------+------------+----------+---------------+ | order_id | product_id | quantity | purchase_date | +----------+------------+----------+---------------+ | 1 | 1 | 7 | 2020-03-16 | | 2 | 1 | 4 | 2020-12-02 | | 3 | 1 | 7 | 2020-05-10 | | 4 | 1 | 6 | 2021-12-23 | | 5 | 1 | 5 | 2021-05-21 | | 6 | 1 | 6 | 2021-10-11 | | 7 | 2 | 6 | 2022-10-11 | +----------+------------+----------+---------------+

Output:

+------------+ | product_id | +------------+ | 1 | +------------+

Explanation:

Product 1 was ordered in 2020 three times and in 2021 three times. Since it was ordered three times in two consecutive years, we include it in the answer. Product 2 was ordered one time in 2022. We do not include it in the answer.

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