Problem 1479: Sales by Day of the Week

Problem Information

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

Problem Description

Table:

Orders

+---------------+---------+ | Column Name | Type | +---------------+---------+ | order_id | int | | customer_id | int | | order_date | date | | item_id | varchar | | quantity | int | +---------------+---------+ (ordered_id, item_id) is the primary key (combination of columns with unique values) for this table. This table contains information on the orders placed. order_date is the date item_id was ordered by the customer with id customer_id.

Table:

Items

+---------------------+---------+ | Column Name | Type | +---------------------+---------+ | item_id | varchar | | item_name | varchar | | item_category | varchar | +---------------------+---------+ item_id is the primary key (column with unique values) for this table. item_name is the name of the item. item_category is the category of the item.

You are the business owner and would like to obtain a sales report for category items and the day of the week.

Write a solution to report how many units in each category have been ordered on each

day of the week

.

Return the result table

ordered

by

category

.

The result format is in the following example.

Example 1:

Input:

Orders table: +------------+--------------+-------------+--------------+-------------+ | order_id | customer_id | order_date | item_id | quantity | +------------+--------------+-------------+--------------+-------------+ | 1 | 1 | 2020-06-01 | 1 | 10 | | 2 | 1 | 2020-06-08 | 2 | 10 | | 3 | 2 | 2020-06-02 | 1 | 5 | | 4 | 3 | 2020-06-03 | 3 | 5 | | 5 | 4 | 2020-06-04 | 4 | 1 | | 6 | 4 | 2020-06-05 | 5 | 5 | | 7 | 5 | 2020-06-05 | 1 | 10 | | 8 | 5 | 2020-06-14 | 4 | 5 | | 9 | 5 | 2020-06-21 | 3 | 5 | +------------+--------------+-------------+--------------+-------------+ Items table: +------------+----------------+---------------+ | item_id | item_name | item_category | +------------+----------------+---------------+ | 1 | LC Alg. Book | Book | | 2 | LC DB. Book | Book | | 3 | LC SmarthPhone | Phone | | 4 | LC Phone 2020 | Phone | | 5 | LC SmartGlass | Glasses | | 6 | LC T-Shirt XL | T-Shirt | +------------+----------------+---------------+

Output:

+------------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+ | Category | Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday | +------------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+ | Book | 20 | 5 | 0 | 0 | 10 | 0 | 0 | | Glasses | 0 | 0 | 0 | 0 | 5 | 0 | 0 | | Phone | 0 | 0 | 5 | 1 | 0 | 0 | 10 | | T-Shirt | 0 | 0 | 0 | 0 | 0 | 0 | 0 | +------------+-----------+-----------+-----------+-----------+-----------+-----------+-----------+

Explanation:

On Monday (2020-06-01, 2020-06-08) were sold a total of 20 units (10 + 10) in the category Book (ids: 1, 2). On Tuesday (2020-06-02) were sold a total of 5 units in the category Book

(ids: 1, 2). On Wednesday (2020-06-03) were sold a total of 5 units in the category Phone (ids: 3, 4). On Thursday (2020-06-04) were sold a total of 1 unit in the category Phone (ids: 3, 4). On Friday (2020-06-05) were sold 10 units in the category Book (ids: 1, 2) and 5 units in Glasses (ids: 5). On Saturday there are no items sold. On Sunday (2020-06-14, 2020-06-21) were sold a total of 10 units (5 +5) in the category Phone (ids: 3, 4). There are no sales of T-shirts.

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 sales_by_day(orders: pd.DataFrame, items: 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 sales_by_day(orders: pd.DataFrame, items: pd.DataFrame) -> pd.DataFrame:
```