Problem 2993: Friday Purchases I

Problem Information

Difficulty: Medium
Acceptance Rate: 80.33%
Paid Only: Yes
Tags: Database

Problem Description

Table: `Purchases`

+---------------+------+ | Column Name | Type | +---------------+------+ | user_id | int | | purchase_date | date | | amount_spend | int | +---------------+------+ (user_id, purchase_date, amount_spend) is the primary key (combination of columns with unique values) for this table. purchase_date will range from November 1, 2023, to November 30, 2023, inclusive of both dates. Each row contains user id, purchase date, and amount spend.

Write a solution to calculate the **total spending** by users on **each Friday** of **every week** in **November 2023**. Output only weeks that include **at least one** purchase on a **Friday**.

Return _the result table ordered by week of month_ _in**ascending**_ _**** order._

The result format is in the following example.

Example 1:

Input: Purchases table: +---------+---------------+--------------+ | user_id | purchase_date | amount_spend | +---------+---------------+--------------+ | 11 | 2023-11-07 | 1126 | | 15 | 2023-11-30 | 7473 | | 17 | 2023-11-14 | 2414 | | 12 | 2023-11-24 | 9692 | | 8 | 2023-11-03 | 5117 | | 1 | 2023-11-16 | 5241 | | 10 | 2023-11-12 | 8266 | | 13 | 2023-11-24 | 12000 | +---------+---------------+--------------+ **Output:** +---------------+---------------+--------------+ | week_of_month | purchase_date | total_amount | +---------------+---------------+--------------+ | 1 | 2023-11-03 | 5117 | | 4 | 2023-11-24 | 21692 | +---------------+---------------+--------------+ **Explanation:** - During the first week of November 2023, transactions amounting to $5,117

occurred on Friday, 2023-11-03. - For the second week of November 2023, there were no transactions on Friday, 2023-11-10. - Similarly, during the third week of November 2023, there were no transactions on Friday, 2023-11-17. - In the fourth week of November 2023, two transactions took place on Friday, 2023-11-24, amounting to $12,000 and $9,692 respectively, summing up to a total of $21,692. Output table is ordered by week_of_month 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
```