Problem 3564: Seasonal Sales Analysis

Problem Information

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

Problem Description

Table:

sales

+---------------+---------+ | Column Name | Type | +---------------+---------+ | sale_id | int | | product_id | int | | sale_date | date | | quantity | int | | price | decimal | +---------------+---------+ sale_id is the unique identifier for this table. Each row contains information about a product sale including the product_id, date of sale, quantity sold, and price per unit.

Table:

products

+---------------+---------+ | Column Name | Type | +---------------+---------+ | product_id | int | | product_name | varchar | | category | varchar | +---------------+---------+ product_id is the unique identifier for this table. Each row contains information about a product including its name and category.

Write a solution to find the most popular product category for each season. The seasons are defined as:

Winter

: December, January, February

Spring

: March, April, May

Summer

: June, July, August

Fall

: September, October, November

The

popularity

of a

category

is determined by the

total quantity sold

in that

season

. If there is a

tie

, select the category with the highest

total revenue

(

quantity × price

).

Return

the result table ordered by season in

ascending

order

.

The result format is in the following example.

Example:

Input:

sales table:

sale_id	product_id	sale_date	quantity	price
1	1	2023-01-15	5	10.00
2	2	2023-01-20	4	15.00
3	3	2023-03-10	3	18.00
4	4	2023-04-05	1	20.00
5	1	2023-05-20	2	10.00
6	2	2023-06-12	4	15.00
7	5	2023-06-15	5	12.00
8	3	2023-07-24	2	18.00
9	4	2023-08-01	5	20.00
10	5	2023-09-03	3	12.00
11	1	2023-09-25	6	10.00
12	2	2023-11-10	4	15.00
13	3	2023-12-05	6	18.00
14	4	2023-12-22	3	20.00
15	5	2024-02-14	2	12.00

products table:

product_id	product_name	category
1	Warm Jacket	Apparel
2	Designer Jeans	Apparel
3	Cutting Board	Kitchen
4	Smart Speaker	Tech
5	Yoga Mat	Fitness

Output:

season	category	total_quantity	total_revenue
Fall	Apparel	10	120.00
Spring	Kitchen	3	54.00
Summer	Tech	5	100.00
Winter	Apparel	9	110.00

Explanation:

Fall (Sep, Oct, Nov):

Apparel: 10 items sold (6 Jackets in Sep, 4 Jeans in Nov), revenue $120.00 (6×$10.00 + 4×$15.00)

Fitness: 3 Yoga Mats sold in Sep, revenue $36.00

Most popular: Apparel with highest total quantity (10)

Spring (Mar, Apr, May):

Kitchen: 3 Cutting Boards sold in Mar, revenue $54.00

Tech: 1 Smart Speaker sold in Apr, revenue $20.00

Apparel: 2 Warm Jackets sold in May, revenue $20.00

Most popular: Kitchen with highest total quantity (3) and highest revenue ($54.00)

Summer (Jun, Jul, Aug):

Apparel: 4 Designer Jeans sold in Jun, revenue $60.00

Fitness: 5 Yoga Mats sold in Jun, revenue $60.00

Kitchen: 2 Cutting Boards sold in Jul, revenue $36.00

Tech: 5 Smart Speakers sold in Aug, revenue $100.00

Most popular: Tech and Fitness both have 5 items, but Tech has higher revenue ($100.00 vs $60.00)

Winter (Dec, Jan, Feb):

Apparel: 9 items sold (5 Jackets in Jan, 4 Jeans in Jan), revenue $110.00

Kitchen: 6 Cutting Boards sold in Dec, revenue $108.00

Tech: 3 Smart Speakers sold in Dec, revenue $60.00

Fitness: 2 Yoga Mats sold in Feb, revenue $24.00

Most popular: Apparel with highest total quantity (9) and highest revenue ($110.00)

The result table is ordered by season 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
```

Oracle:

```
/* Write your PL/SQL query statement below */
```

Pandas:

```
import pandas as pd

def seasonal_sales_analysis(products: pd.DataFrame, sales: 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:

```python
import pandas as pd

def seasonal_sales_analysis(products: pd.DataFrame, sales: pd.DataFrame) ->
pd.DataFrame:
```