

Problem 3601: Find Drivers with Improved Fuel Efficiency

Problem Information

Difficulty: **Medium**

Acceptance Rate: 47.32%

Paid Only: No

Tags: Database

Problem Description

Table: `drivers`

```
+-----+-----+ | Column Name | Type | +-----+-----+ | driver_id | int | |
driver_name | varchar | +-----+-----+ driver_id is the unique identifier for this table.
Each row contains information about a driver.
```

Table: `trips`

```
+-----+-----+ | Column Name | Type | +-----+-----+ | trip_id | int | | driver_id
| int | | trip_date | date | | distance_km | decimal | | fuel_consumed | decimal |
+-----+-----+ trip_id is the unique identifier for this table. Each row represents a trip
made by a driver, including the distance traveled and fuel consumed for that trip.
```

Write a solution to find drivers whose **fuel efficiency has improved** by **comparing** their average fuel efficiency in the **first half** of the year with the **second half** of the year.

* Calculate **fuel efficiency** as `distance_km / fuel_consumed` for **each** trip * **First half** : January to June, **Second half** : July to December * Only include drivers who have trips in **both halves** of the year * Calculate the **efficiency improvement** as `(second_half_avg - first_half_avg)` * **Round** all **results** to **2** **decimal** **places**

Return `_` the result table ordered by efficiency improvement in **descending** order, then by driver name in **ascending** order.

The result format is in the following example.

****Example:****

****Input:****

drivers table:

driver_id	driver_name	1	Alice Johnson
2	Bob Smith	3	Carol Davis
4	David Wilson	5	Emma Brown

trips table:

trip_id	driver_id	trip_date	distance_km	fuel_consumed
1	1	2023-02-15	120.5	10.2
2	1	2023-03-20	200.0	16.5
3	1	2023-08-10	150.0	11.0
4	1	2023-09-25	180.0	12.5
5	2	2023-01-10	100.0	9.0
6	2	2023-04-15	250.0	22.0
7	2	2023-10-05	200.0	15.0
8	3	2023-03-12	80.0	8.5
9	3	2023-05-18	90.0	9.2
10	4	2023-07-22	160.0	12.8
11	4	2023-11-30	140.0	11.0
12	5	2023-02-28	110.0	11.5

****Output:****

driver_id	driver_name	first_half_avg	second_half_avg	efficiency_improvement
2	Bob Smith	11.24	13.33	2.10
1	Alice Johnson	11.97	14.02	2.05

****Explanation:****

* **Alice Johnson (driver_id = 1):** * First half trips (Jan-Jun): Feb 15 (120.5/10.2 = 11.81), Mar 20 (200.0/16.5 = 12.12) * First half average efficiency: (11.81 + 12.12) / 2 = 11.97 * Second half trips (Jul-Dec): Aug 10 (150.0/11.0 = 13.64), Sep 25 (180.0/12.5 = 14.40) * Second half average efficiency: (13.64 + 14.40) / 2 = 14.02 * Efficiency improvement: 14.02 - 11.97 = 2.05 * **Bob Smith (driver_id = 2):** * First half trips: Jan 10 (100.0/9.0 = 11.11), Apr 15 (250.0/22.0 = 11.36) * First half average efficiency: (11.11 + 11.36) / 2 = 11.24 * Second half trips: Oct 5 (200.0/15.0 = 13.33) * Second half average efficiency: 13.33 * Efficiency improvement: 13.33 - 11.24 = 2.10 (rounded to 2 decimal places) * **Drivers not included:** * Carol Davis (driver_id = 3): Only has trips in first half (Mar, May) * David Wilson (driver_id = 4): Only has trips in second half (Jul, Nov) * Emma Brown (driver_id = 5): Only has trips in first

half (Feb)

The output table is ordered by efficiency improvement in descending order then by name 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
```