

Problem 2854: Rolling Average Steps

Problem Information

Difficulty: Medium

Acceptance Rate: 0.00%

Paid Only: No

Problem Description

Table:

Steps

+-----+-----+ | Column Name | Type | +-----+-----+ | user_id | int | | steps_count | int | | steps_date | date | +-----+-----+ (user_id, steps_date) is the primary key for this table. Each row of this table contains user_id, steps_count, and steps_date.

Write a solution to calculate

3-day

rolling averages

of steps for each user.

We calculate the

n-day

rolling average

this way:

For each day, we calculate the average of

n

consecutive days of step counts ending on that day if available, otherwise,

n-day

rolling average is not defined for it.

Output the

user_id

,

steps_date

, and rolling average. Round the rolling average to

two decimal places

.

Return

the result table ordered by

user_id

,

steps_date

in

ascending

order.

The result format is in the following example.

Example 1:

Input:

Steps table: +-----+-----+-----+ | user_id | steps_count | steps_date |
+-----+-----+-----+ | 1 | 687 | 2021-09-02 || 1 | 395 | 2021-09-04 || 1 | 499 |
2021-09-05 || 1 | 712 | 2021-09-06 || 1 | 576 | 2021-09-07 || 2 | 153 | 2021-09-06 || 2 | 171 |
2021-09-07 || 2 | 530 | 2021-09-08 || 3 | 945 | 2021-09-04 || 3 | 120 | 2021-09-07 || 3 | 557 |
2021-09-08 || 3 | 840 | 2021-09-09 || 3 | 627 | 2021-09-10 || 5 | 382 | 2021-09-05 || 6 | 480 |
2021-09-01 || 6 | 191 | 2021-09-02 || 6 | 303 | 2021-09-05 | +-----+-----+

Output:

+-----+-----+-----+ | user_id | steps_date | rolling_average |
+-----+-----+-----+ | 1 | 2021-09-06 | 535.33 | | 1 | 2021-09-07 | 595.67 | | 2 |
2021-09-08 | 284.67 | | 3 | 2021-09-09 | 505.67 | | 3 | 2021-09-10 | 674.67 |
+-----+-----+

Explanation:

- For user id 1, the step counts for the three consecutive days up to 2021-09-06 are available. Consequently, the rolling average for this particular date is computed as $(395 + 499 + 712) / 3 = 535.33$. - For user id 1, the step counts for the three consecutive days up to 2021-09-07 are available. Consequently, the rolling average for this particular date is computed as $(499 + 712 + 576) / 3 = 595.67$. - For user id 2, the step counts for the three consecutive days up to 2021-09-08 are available. Consequently, the rolling average for this particular date is computed as $(153 + 171 + 530) / 3 = 284.67$. - For user id 3, the step counts for the three consecutive days up to 2021-09-09 are available. Consequently, the rolling average for this particular date is computed as $(120 + 557 + 840) / 3 = 505.67$. - For user id 3, the step counts for the three consecutive days up to 2021-09-10 are available. Consequently, the rolling average for this particular date is computed as $(557 + 840 + 627) / 3 = 674.67$. - For user id 4 and 5, the calculation of the rolling average is not viable as there is insufficient data for the consecutive three days. Output table ordered by user_id and steps_date 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 rolling_average(steps: 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 rolling_average(steps: pd.DataFrame) -> pd.DataFrame:
```