

RESTAURANT GROWTH

JUNE 2024

Window, Aggregate, Grouping & Clause Functions

Leetcode

Advanced

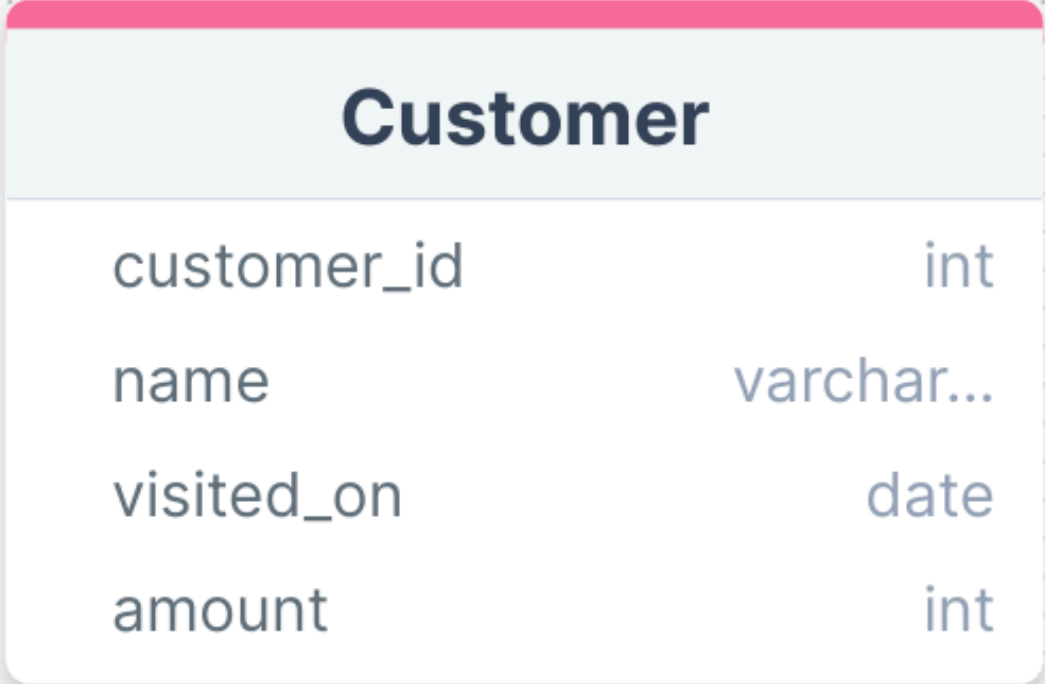
A restaurant owner has requested an analysis for a possible expansion (there will be at least one customer every day).

Compute the moving average of how much the customer paid in a seven days window (i.e., current day + 6 days before). `average_amount` should be **rounded to two decimal places**.

Return the result table ordered by `visited_on` in **ascending order**.

INPUT FORMAT

The main source tables is **CUSTOMER**.



Customer	
customer_id	int
name	varchar...
visited_on	date
amount	int

Customer table:

customer_id	name	visited_on	amount
1	Jhon	2019-01-01	100
2	Daniel	2019-01-02	110
3	Jade	2019-01-03	120
4	Khaled	2019-01-04	130
5	Winston	2019-01-05	110
6	Elvis	2019-01-06	140
7	Anna	2019-01-07	150
8	Maria	2019-01-08	80

9	Jaze	2019-01-09	110
1	Jhon	2019-01-10	130

CODE SOLUTION

```
SELECT * FROM
(
    SELECT
        visited_on,
        SUM(cust.amount) OVER (ORDER BY visited_on RANGE BETWEEN INTERVAL '6' DAY
PRECEDING AND CURRENT ROW) AS amount,
        ROUND(AVG(cust.amount) OVER (ORDER BY visited_on RANGE BETWEEN INTERVAL '6'
DAY PRECEDING AND CURRENT ROW),2) AS average_amount
    FROM (SELECT visited_on, SUM(amount) AS amount FROM Customer GROUP BY 1
ORDER BY 1) AS cust
) AS c
WHERE visited_on >= (SELECT MIN(visited_on) FROM Customer) + INTERVAL '6' DAY
```

SOLUTION PROCESS

- Innermost subqueries: Select statement under the alias **cust** groups sum of revenue based on unique identifier column to ensure a single record of revenue per customer. Otherwise, moving averages in outer select will be calculated using varying denominators. i.e. dividing by 8 instead of 7 days
 - Window function: Aggregates of revenue in OVER function calculated within interval range as opposed to row number ranges to correspond with date type column
 - Outermost select statement: WHERE clause filters for dates at the end of a 7-day interval
-

OUTPUT

visited_on	amount	average_amount
2019-01-07	860	122.86
2019-01-08	840	120
2019-01-09	840	120
2019-01-10	1000	142.86
