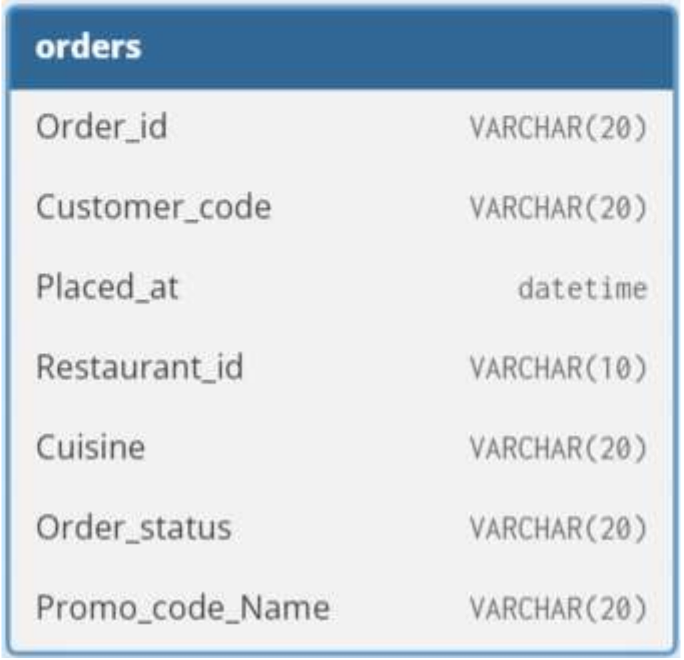
SQL Project

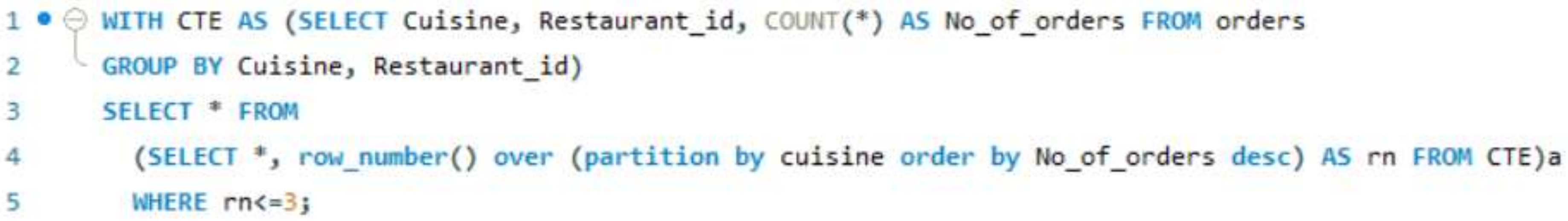
Tool: MySQL Workbench

Table Schema

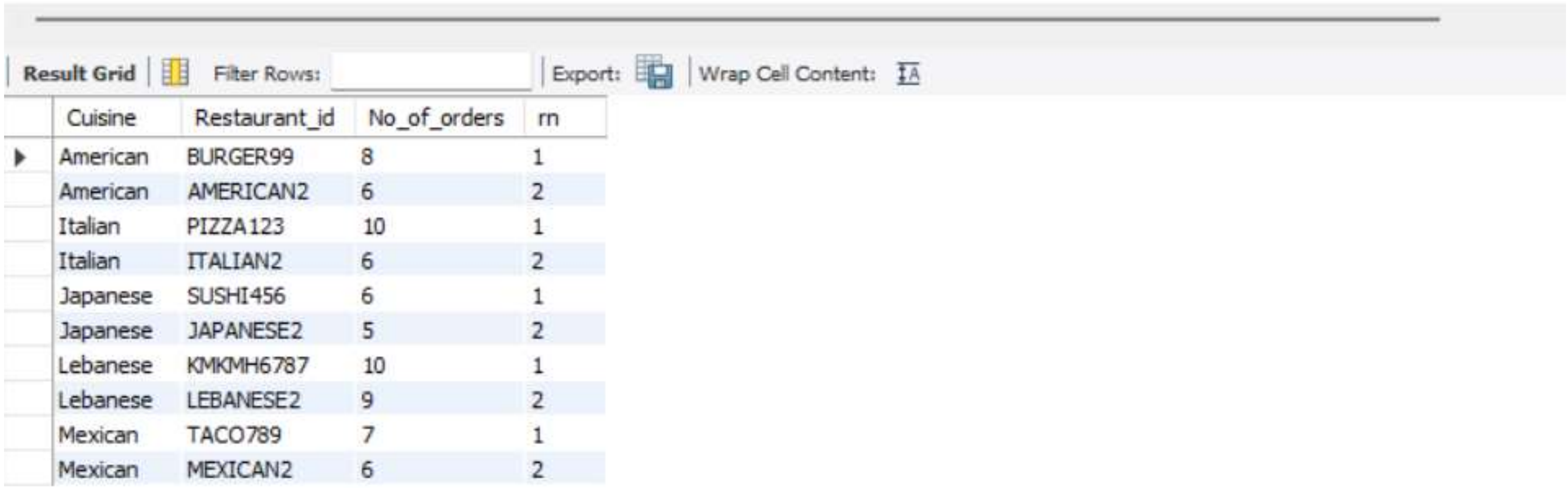


orders	
Order_id	VARCHAR(20)
Customer_code	VARCHAR(20)
Placed_at	datetime
Restaurant_id	VARCHAR(10)
Cuisine	VARCHAR(20)
Order_status	VARCHAR(20)
Promo_code_Name	VARCHAR(20)

Top 3 outlets by Cuisine type

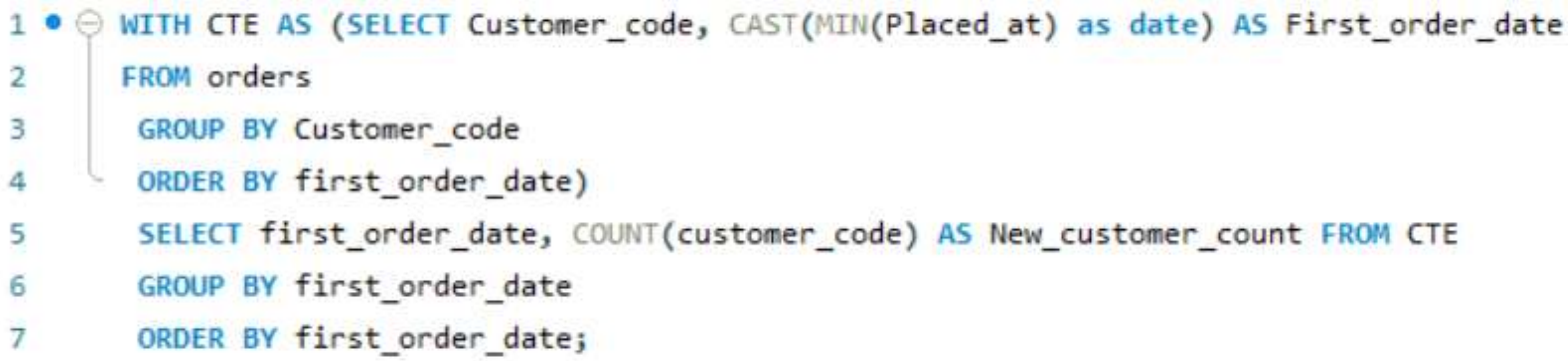


```sql
WITH CTE AS (SELECT Cuisine, Restaurant_id, COUNT(*) AS No_of_orders FROM orders
GROUP BY Cuisine, Restaurant_id)
SELECT * FROM
  (SELECT *, row_number() over (partition by cuisine order by No_of_orders desc) AS rn FROM CTE)a
  WHERE rn<=3;
```

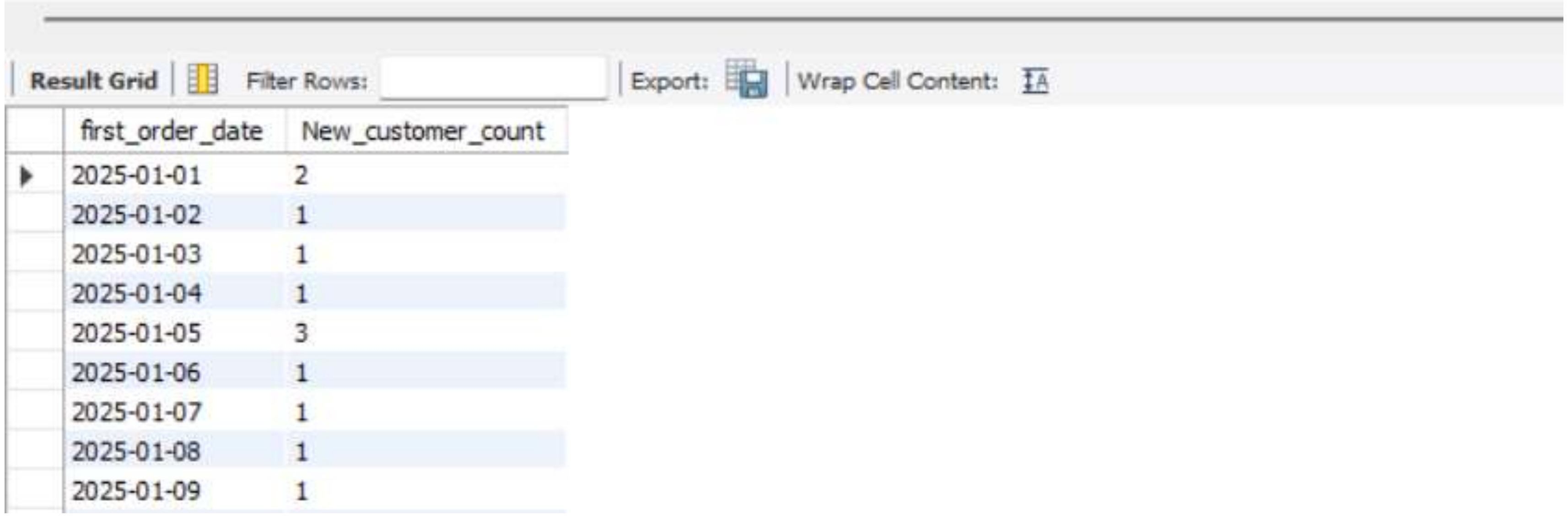


Cuisine	Restaurant_id	No_of_orders	rn
American	BURGER99	8	1
American	AMERICAN2	6	2
Italian	PIZZA123	10	1
Italian	ITALIAN2	6	2
Japanese	SUSHI456	6	1
Japanese	JAPANESE2	5	2
Lebanese	KMKMH6787	10	1
Lebanese	LEBANESE2	9	2
Mexican	TACO789	7	1
Mexican	MEXICAN2	6	2

Count of new customers acquired every day from the launch date

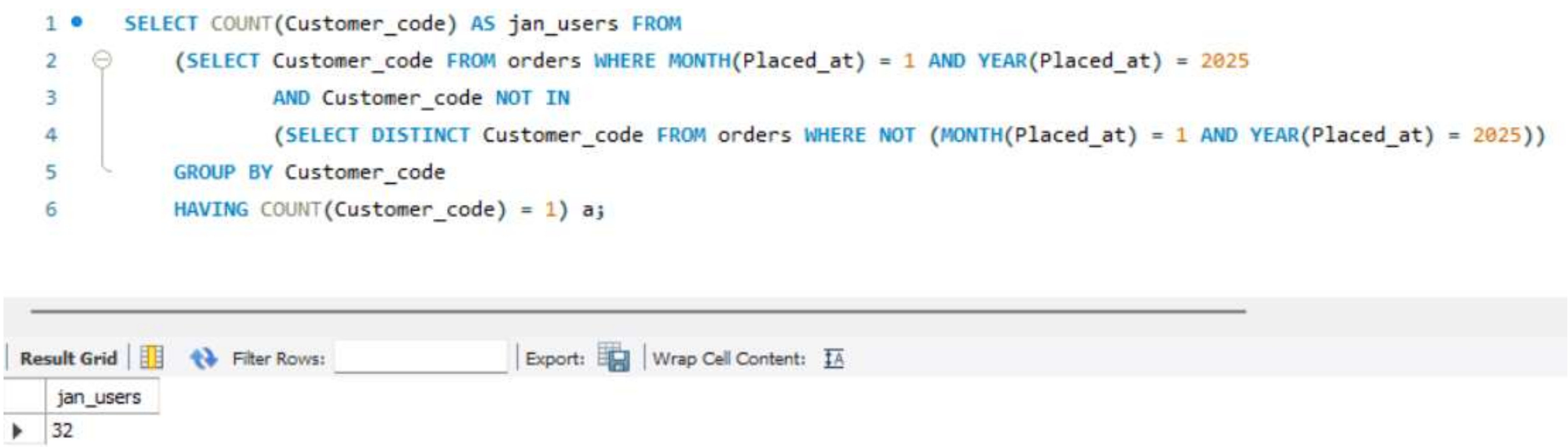


```sql
WITH CTE AS (SELECT Customer_code, CAST(MIN(Placed_at) as date) AS First_order_date
FROM orders
 GROUP BY Customer_code
 ORDER BY first_order_date)
 SELECT first_order_date, COUNT(customer_code) AS New_customer_count FROM CTE
 GROUP BY first_order_date
 ORDER BY first_order_date;
```



first_order_date	New_customer_count
2025-01-01	2
2025-01-02	1
2025-01-03	1
2025-01-04	1
2025-01-05	3
2025-01-06	1
2025-01-07	1
2025-01-08	1
2025-01-09	1

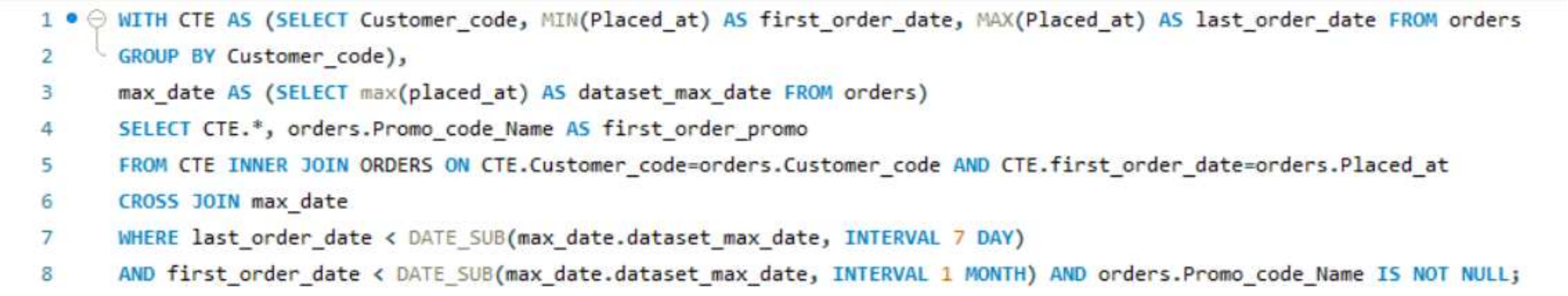
Count of all the users acquired with only a single order in January



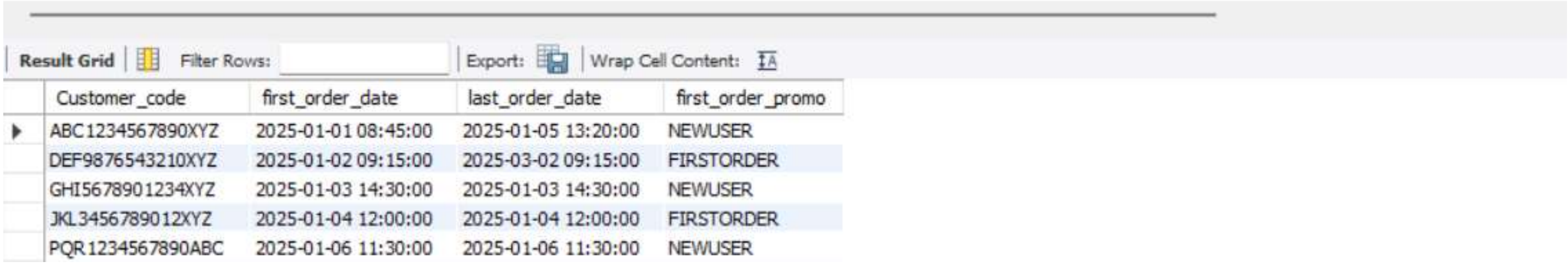
```
SELECT COUNT(Customer_code) AS jan_users FROM
    (SELECT Customer_code FROM orders WHERE MONTH(Placed_at) = 1 AND YEAR(Placed_at) = 2025
            AND Customer_code NOT IN
            (SELECT DISTINCT Customer_code FROM orders WHERE NOT (MONTH(Placed_at) = 1 AND YEAR(Placed_at) = 2025))
    GROUP BY Customer_code
    HAVING COUNT(Customer_code) = 1) a;
```

```
jan_users
32
```

List of all the customers with no order in the last 7 days, but were acquired one month ago with their order on promo.

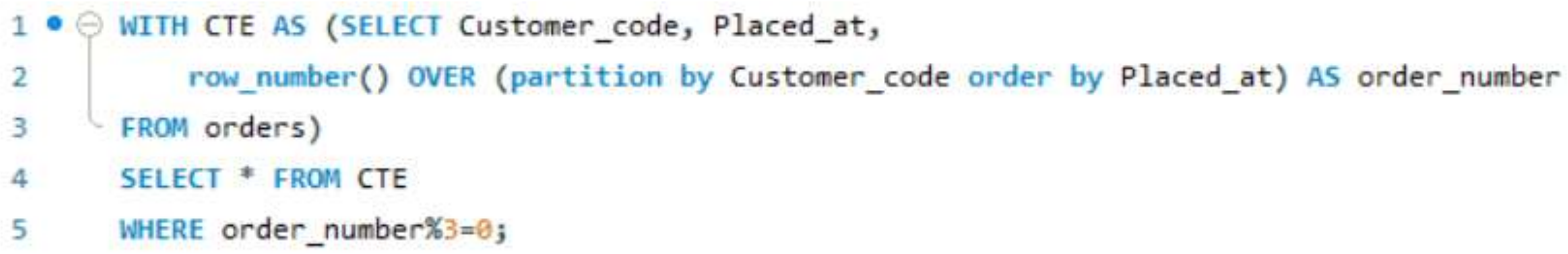


```sql
WITH CTE AS (SELECT Customer_code, MIN(Placed_at) AS first_order_date, MAX(Placed_at) AS last_order_date FROM orders
GROUP BY Customer_code),
max_date AS (SELECT max(placed_at) AS dataset_max_date FROM orders)
SELECT CTE.*, orders.Promo_code_Name AS first_order_promo
FROM CTE INNER JOIN ORDERS ON CTE.Customer_code=orders.Customer_code AND CTE.first_order_date=orders.Placed_at
CROSS JOIN max_date
WHERE last_order_date < DATE_SUB(max_date.dataset_max_date, INTERVAL 7 DAY)
AND first_order_date < DATE_SUB(max_date.dataset_max_date, INTERVAL 1 MONTH) AND orders.Promo_code_Name IS NOT NULL;
```

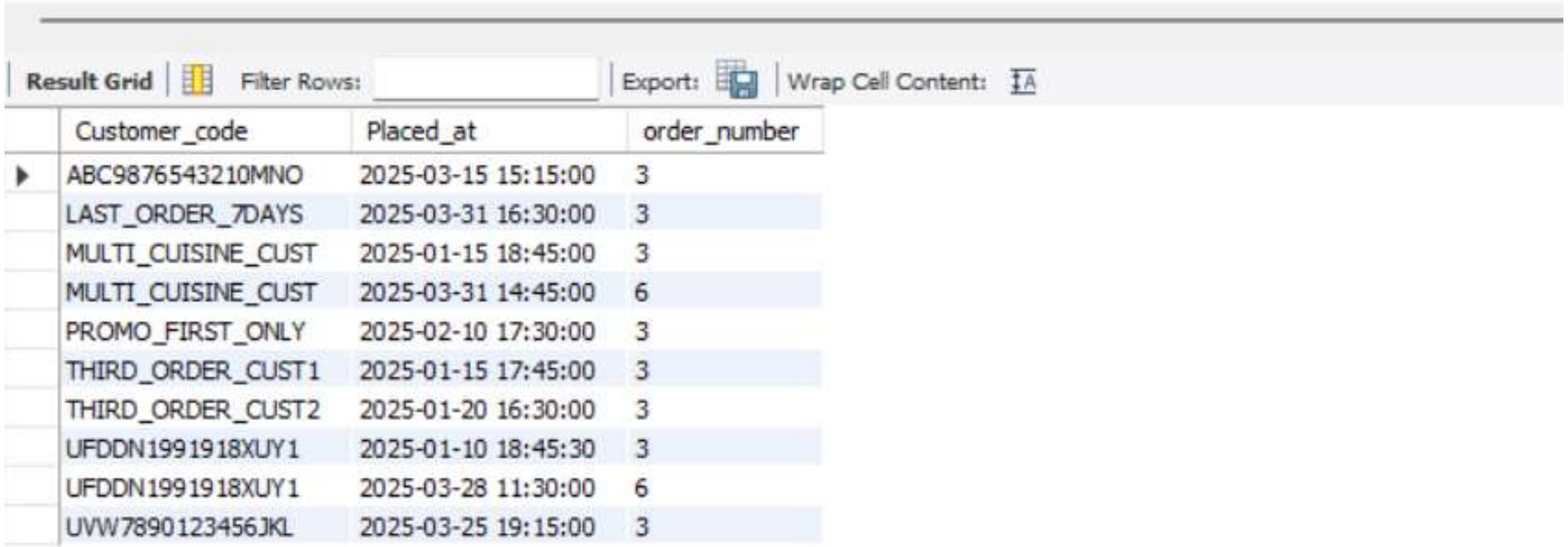


Customer_code	first_order_date	last_order_date	first_order_promo
ABC1234567890XYZ	2025-01-01 08:45:00	2025-01-05 13:20:00	NEWUSER
DEF9876543210XYZ	2025-01-02 09:15:00	2025-03-02 09:15:00	FIRSTORDER
GHI5678901234XYZ	2025-01-03 14:30:00	2025-01-03 14:30:00	NEWUSER
JKL3456789012XYZ	2025-01-04 12:00:00	2025-01-04 12:00:00	FIRSTORDER
PQR1234567890ABC	2025-01-06 11:30:00	2025-01-06 11:30:00	NEWUSER

Create a trigger that will target customers after every third order with a personalized communication

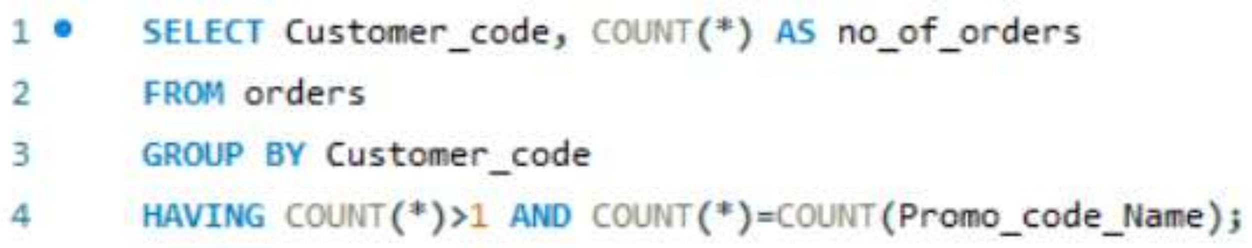


```sql
WITH CTE AS (SELECT Customer_code, Placed_at,
    row_number() OVER (partition by Customer_code order by Placed_at) AS order_number
FROM orders)
SELECT * FROM CTE
WHERE order_number%3=0;
```

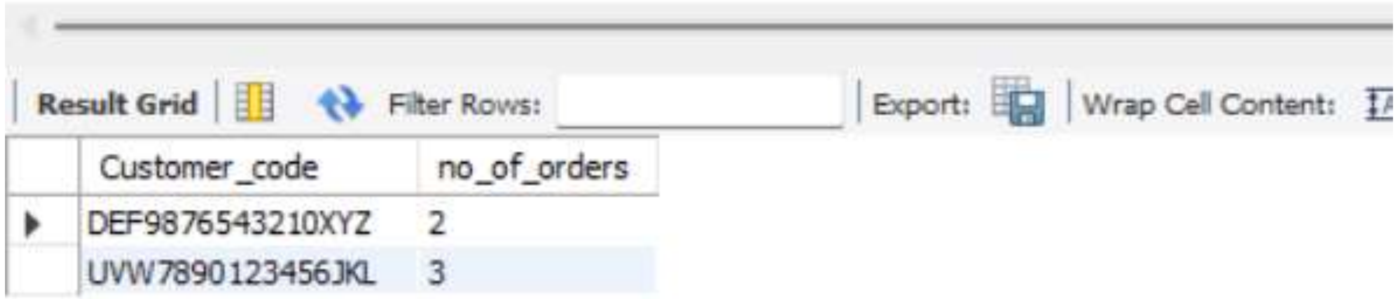


Customer_code	Placed_at	order_number
ABC9876543210MNO	2025-03-15 15:15:00	3
LAST_ORDER_7DAYS	2025-03-31 16:30:00	3
MULTI_CUISINE_CUST	2025-01-15 18:45:00	3
MULTI_CUISINE_CUST	2025-03-31 14:45:00	6
PROMO_FIRST_ONLY	2025-02-10 17:30:00	3
THIRD_ORDER_CUST1	2025-01-15 17:45:00	3
THIRD_ORDER_CUST2	2025-01-20 16:30:00	3
UFDDN1991918XUY1	2025-01-10 18:45:30	3
UFDDN1991918XUY1	2025-03-28 11:30:00	6
UVW7890123456JKL	2025-03-25 19:15:00	3

List of the customers who placed more than 1 order, and all their orders were on a promo.

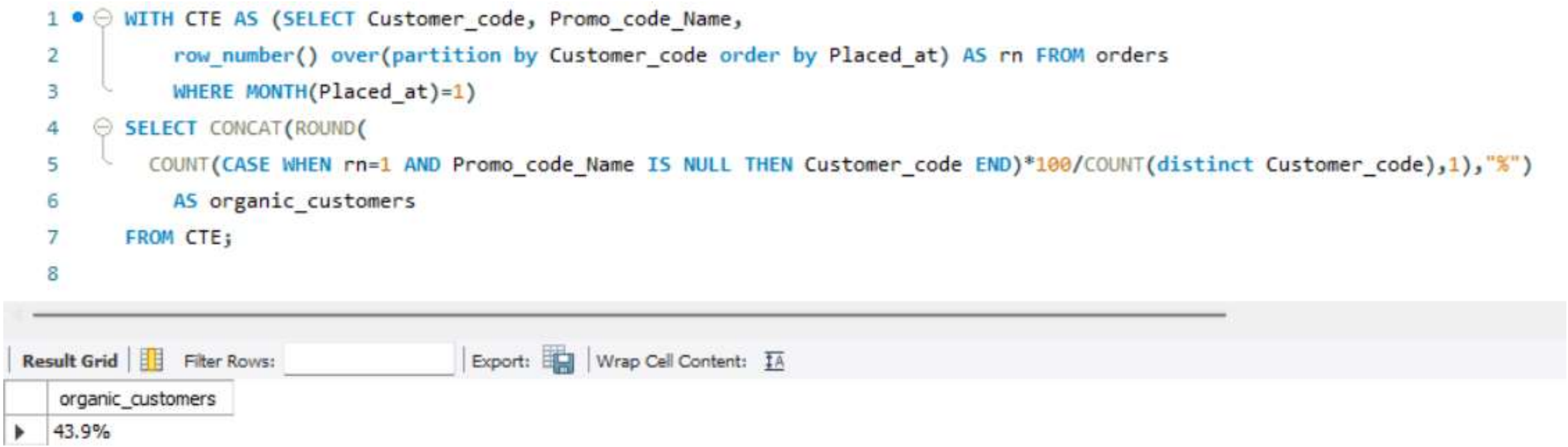


```sql
SELECT Customer_code, COUNT(*) AS no_of_orders
FROM orders
GROUP BY Customer_code
HAVING COUNT(*)>1 AND COUNT(*)=COUNT(Promo_code_Name);
```



Customer_code	no_of_orders
DEF9876543210XYZ	2
UVW7890123456JKL	3

Percentage of customers organically acquired in Jan 2025 (placed their first order without a promo code)



```sql
WITH CTE AS (SELECT Customer_code, Promo_code_Name,
    row_number() over(partition by Customer_code order by Placed_at) AS rn FROM orders
    WHERE MONTH(Placed_at)=1)
SELECT CONCAT(ROUND(
  COUNT(CASE WHEN rn=1 AND Promo_code_Name IS NULL THEN Customer_code END)*100/COUNT(distinct Customer_code),1),"%")
    AS organic_customers
FROM CTE;
```

```
organic_customers
43.9%
```