**(1)** (2) (3)

# 

586. Customer Placing the Largest Number of Orders 🗗

Average Rating: 4.61 (18 votes)

June 9, 2017 | 23.8K views

Query the **customer\_number** from the **orders** table for the customer who has placed the largest number of orders.

The orders table is defined as follows:

It is guaranteed that exactly one customer will have placed more orders than any other customer.

Column	Type
order_number (PK)	int
customer_number	int
order_date	date
required_date	date
shipped_date	date
status	char(15)
comment	char(200)

	order_number	customer_number	order_date	required_date	shipped_date	status
	1	1	2017-04-09	2017-04-13	2017-04-12	Closed
	2	2	2017-04-15	2017-04-20	2017-04-18	Closed
	3	3	2017-04-16	2017-04-25	2017-04-20	Closed
1	4	3	2017-04-18	2017-04-28	2017-04-25	Closed
4						<b>+</b>
	UNANGOTOTO :					
Sar	nple Output					

#### customer\_number

```
Explanation
```

#### The customer with number '3' has two orders, which is greater than either customer '1'

```
Follow up: What if more than one customer have the largest number of orders, can you find all the
customer_number in this case?
```

## Approach: Using LIMIT [Accepted]

Solution

First, we can select the **customer\_number** and the according count of orders using **GROUP BY**.

So the result is customer\_number '3'.

### SELECT

2

3

customer\_number

SELECT

LIMIT 1

FROM

Algorithm

customer\_number, COUNT(\*) FROM

```
orders
GROUP BY customer_number
                                                      COUNT(*)
customer_number
                                                      1
```

1

2

COUNT(\*)

3

Then, the **customer\_number** of first record is the result after sorting them by order count descending.

```
In MySQL, the LIMIT clause can be used to constrain the number of rows returned by the SELECT statement.
It takes one or two nonnegative numeric arguments, the first of which specifies the offset of the first row to
return, and the second specifies the maximum number of rows to return. The offset of the initial row is 0 (not
1).
```

MySQL

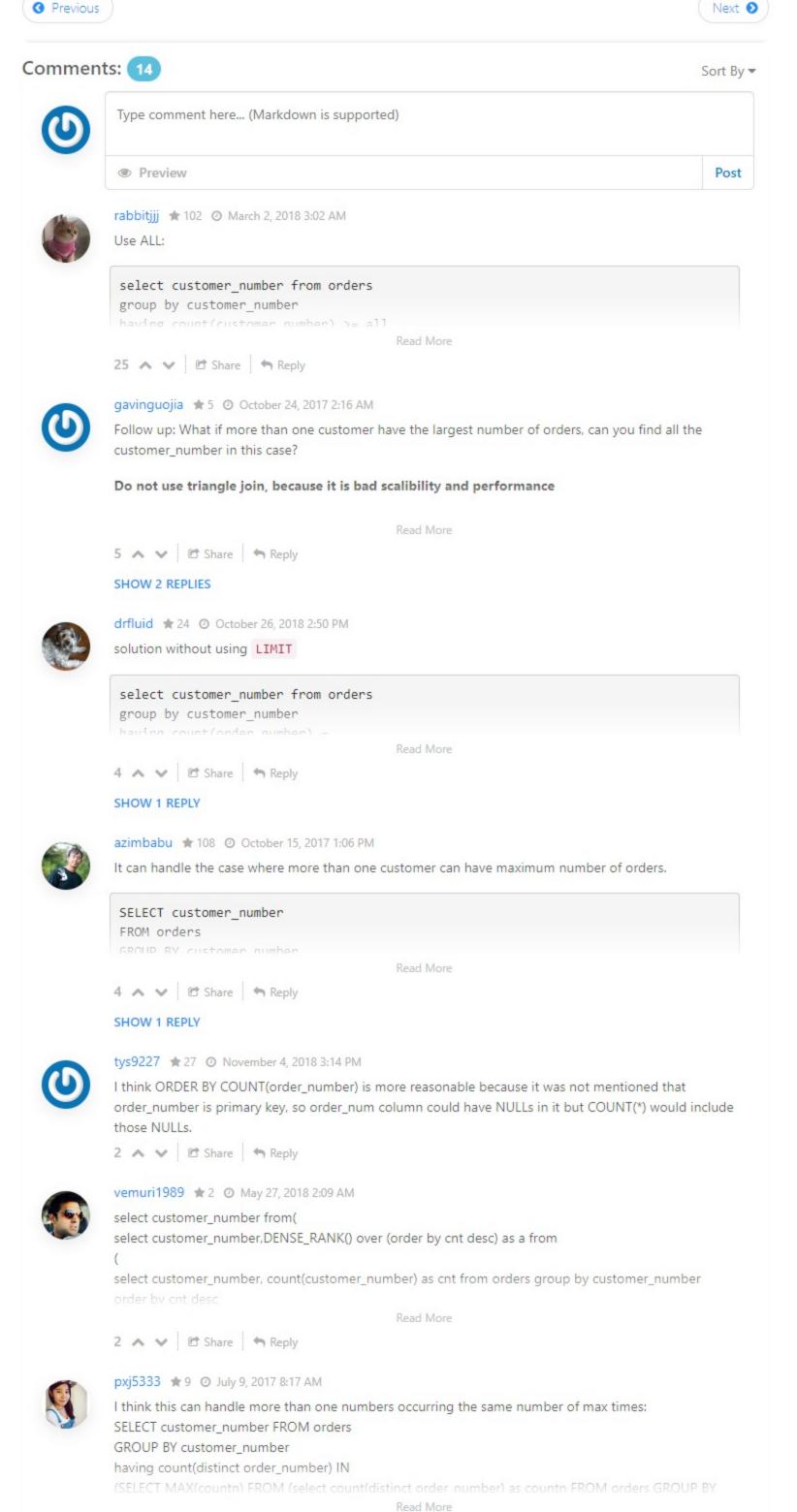
It can be used with only one argument, which specifies the number of rows to return from the beginning of

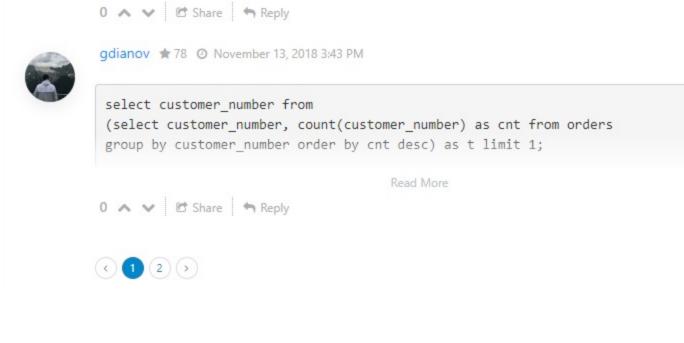
```
Rate this article: * * * * *
```

customer\_number

GROUP BY customer\_number ORDER BY COUNT(\*) DESC

the result set. So LIMIT 1 will return the first record.





(SELECT count( distinct order\_number) as num, customer\_number FROM orders group by

Read More

Read More

2 A V C Share Share

select customer\_number from

0 ∧ ∨ Ø Share ♠ Reply

select customer\_number

group by customer\_number

order by count(order\_number) desc

customer\_number)

SHOW 1 REPLY

from orders

sweta31jan 🛊 0 🗿 January 20, 2019 2:27 PM

( select max(num),customer\_number from

Irongoddess \* 0 @ January 15, 2019 8:36 AM