**4.**

**Customer Placing the Largest Number of Orders**

Table: Orders

+-----------------+----------+

| Column Name | Type |

+-----------------+----------+

| order\_number | int |

| customer\_number | int |

+-----------------+----------+

order\_number is the primary key for this table.

This table contains information about the order ID and the customer ID.

Write an SQL query to find the customer\_number for the customer who has placed the largest number of orders.

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

The query result format is in the following example:

Orders table:

+--------------+-----------------+

| order\_number | customer\_number |

+--------------+-----------------+

| 1 | 1 |

| 2 | 2 |

| 3 | 3 |

| 4 | 3 |

+--------------+-----------------+

Result table:

+-----------------+

| customer\_number |

+-----------------+

| 3 |

+-----------------+

The customer with number 3 has two orders, which is greater than either customer 1 or 2 because each of them only has one order.

So the result is customer\_number 3.

\*/

Create table If Not Exists orders (order\_number int, customer\_number int, order\_date date, required\_date date, shipped\_date date, status char(15), comment char(200), key(order\_number))

Truncate table orders

insert into orders (order\_number, customer\_number) values ('1', '1')

insert into orders (order\_number, customer\_number) values ('2', '2')

insert into orders (order\_number, customer\_number) values ('3', '3')

insert into orders (order\_number, customer\_number) values ('4', '3')