



# MySQL OR Operator

**Summary:** in this tutorial, you will learn how to use the MySQL `OR` operator to combine Boolean expressions.

## Introduction to the MySQL OR operator

The MySQL `OR` operator is a logical operator that combines two [Boolean](https://www.mysqltutorial.org/mysql-boolean/) expressions.

```
A OR B
```

If both A and B are not NULL, the OR operator returns 1 (true) if either A or B is non-zero. For example:

```
SELECT 1 OR 1, 1 OR 0, 0 OR 1;
```

```
+-----+-----+-----+
| 1 OR 1 | 1 OR 0 | 0 OR 1 |
+-----+-----+-----+
|      1 |      1 |      1 |
+-----+-----+-----+
1 row in set (0.00 sec)
```

If both A and B are zero (false), the OR operator returns zero. For example:

```
SELECT 0 OR 0;
```

```
+-----+
| 0 OR 0 |
+-----+
|      0 |
+-----+
1 row in set (0.00 sec)
```

When A and / or B is NULL, the OR operator returns 1 (true) if either A or B is non-zero. Otherwise, it returns NULL. For example:

```
SELECT 1 OR NULL, 0 OR NULL, NULL or NULL;
```

```
+-----+-----+-----+
| 1 OR NULL | 0 OR NULL | NULL or NULL |
+-----+-----+-----+
|          1 |        NULL |          NULL |
+-----+-----+-----+
1 row in set (0.00 sec)
```

The following table shows the result of the **OR** operator when combining true (non-zero), false (zero), and NULL:

|       | TRUE | FALSE | NULL |
|-------|------|-------|------|
| TRUE  | TRUE | TRUE  | TRUE |
| FALSE | TRUE | FALSE | NULL |
| NULL  | TRUE | NULL  | NULL |

## MySQL OR operator and short-circuit evaluation

Like the **AND** (<https://www.mysqltutorial.org/mysql-and/>) operator, the **OR** operator is also short-circuited. In other words, MySQL stops evaluating the remaining parts of the expression as soon as it can determine the result. For example:

```
SELECT 1 = 1 OR 1 / 0;
```

Try It Out >

```
+-----+
| 1 = 1 OR 1 / 0 |
+-----+
|          1 |
+-----+
1 row in set (0.00 sec)
```

Because the expression `1 = 1` always returns 1, MySQL won't evaluate the `1 / 0` expression. And MySQL would issue an error if it did.

## Operator precedence

When an expression contains both `AND` and `OR` operators, MySQL uses the operator precedence to determine the order of evaluation of the operators. MySQL evaluates the operator with higher precedence first.

Since the `AND` operator has higher precedence than the `OR` operator, MySQL evaluates the `AND` operator before the `OR` operator. For example:

```
SELECT 1 OR 0 AND 0;
```

Try It Out >

```
+-----+
| 1 OR 0 AND 0 |
+-----+
|           1 |
+-----+
1 row in set (0.00 sec)
```

How it works.

```
1 OR 0 AND 0 = 1 OR 0 = 1
```

To change the order of evaluation, you use the parentheses. For example:

```
SELECT (1 OR 0) AND 0;
```

Try It Out >

```
+-----+
| (1 OR 0) AND 0 |
+-----+
|           0 |
+-----+
1 row in set (0.00 sec)
```

How it works.

```
(1 OR 0) AND 0 = 1 AND 0 = 0
```

## MySQL OR operator examples

We'll use the `customers` table from the [sample database](https://www.mysqltutorial.org/mysql-sample-database.aspx) (<https://www.mysqltutorial.org/mysql-sample-database.aspx>) for the demonstration.

The following query uses the `OR` operator in the `WHERE` (<https://www.mysqltutorial.org/mysql-where/>) clause to select all the customers located in the USA or France:

```
SELECT
    customername,
    country
FROM
    customers
WHERE country = 'USA' OR
       country = 'France';
```

Try It Out



```
+-----+-----+
| customername | country |
+-----+-----+
```

|                              |        |
|------------------------------|--------|
| Atelier graphique            | France |
| Signal Gift Stores           | USA    |
| La Rochelle Gifts            | France |
| Mini Gifts Distributors Ltd. | USA    |
| Mini Wheels Co.              | USA    |
| Land of Toys Inc.            | USA    |
| Saveley & Henriot, Co.       | France |
| Muscle Machine Inc           | USA    |
| Diecast Classics Inc.        | USA    |
| ...                          |        |

The following example uses the `OR` operator to select the customers who locate in the USA or France and have a credit limit greater than 100,000.

```
SELECT
    customername,
    country,
    creditLimit
FROM
    customers
WHERE (country = 'USA'
       OR country = 'France')
      AND creditlimit > 100000;
```

[Try It Out >](#)

| customername                 | country | creditLimit |
|------------------------------|---------|-------------|
| La Rochelle Gifts            | France  | 118200.00   |
| Mini Gifts Distributors Ltd. | USA     | 210500.00   |
| Land of Toys Inc.            | USA     | 114900.00   |
| Saveley & Henriot, Co.       | France  | 123900.00   |
| Muscle Machine Inc           | USA     | 138500.00   |
| Diecast Classics Inc.        | USA     | 100600.00   |
| Collectable Mini Designs Co. | USA     | 105000.00   |
| Marta's Replicas Co.         | USA     | 123700.00   |
| Mini Classics                | USA     | 102700.00   |
| Corporate Gift Ideas Co.     | USA     | 105000.00   |
| Online Diecast Creations Co. | USA     | 114200.00   |

11 rows in set (0.00 sec)

Notice that if you do not use the parentheses, the query will return the customers who locate in the USA or the customers located in France with a credit limit greater than 100,000.

```
SELECT
    customername,
    country,
    creditLimit
FROM
    customers
WHERE
    country = 'USA'
    OR country = 'France'
    AND creditlimit > 100000;
```

[Try It Out](#)

| customername                 | country | creditLimit |
|------------------------------|---------|-------------|
| Signal Gift Stores           | USA     | 71800.00    |
| La Rochelle Gifts            | France  | 118200.00   |
| Mini Gifts Distributors Ltd. | USA     | 210500.00   |
| Mini Wheels Co.              | USA     | 64600.00    |
| Land of Toys Inc.            | USA     | 114900.00   |
| Saveley & Henriot, Co.       | France  | 123900.00   |
| Muscle Machine Inc           | USA     | 138500.00   |
| Diecast Classics Inc.        | USA     | 100600.00   |
| Technics Stores Inc.         | USA     | 84600.00    |
| American Souvenirs Inc       | USA     | 0.00        |

...

## Summary

- The **OR** operator combines two Boolean expressions and returns true when either expression is true. Otherwise, it returns false.
- MySQL evaluates the **OR** operator after the **AND** operator if an expression contains both **AND** and **OR** operators.
- Use parentheses to change the order of evaluation.