MySQL: IN Condition

This MySQL tutorial explains how to use the MySQL **IN condition** with syntax and examples.

Description

The MySQL IN condition is used to help reduce the need to use multiple OR conditions in a SELECT, INSERT, UPDATE, or DELETE statement.

Syntax

The syntax for the IN condition in MySQL is:

```
expression IN (value1, value2, .... value_n);
```

Parameters or Arguments

expression

The value to test.

value1, value2, ... or value_n

These are the values to test against *expression*. If any of these values matches *expression*, then the IN condition will evaluate to true. This is a quick method to test if any one of the values matches *expression*.

Note

- The MySQL IN condition will return the records where *expression* is *value1*, *value2*..., or *value_n*.
- The MySQL IN condition is also called the MySQL IN operator.

Example - With Character

Let's look at a MySQL IN condition example using character values.

The following is a MySQL SELECT statement that uses the IN condition to compare character values:

```
SELECT *
FROM contacts
WHERE last_name IN ('Johnson', 'Anderson', 'Smith');
```

This MySQL IN condition example would return all rows from the *contacts* table where the *last_name* is either Johnson, Anderson or Smith. Because the * is used in the SELECT, all fields from the *contacts* table would appear in the result set.

The above IN example is equivalent to the following SELECT statement:

```
SELECT *
FROM contacts
WHERE last_name = 'Johnson'
OR last_name = 'Anderson'
OR last_name = 'Smith';
```

As you can see, using the MySQL IN condition makes the statement easier to read and more efficient.

Example - With Numeric

Next, let's look at a MySQL IN condition example using numeric values.

For example:

```
SELECT *
FROM suppliers
WHERE supplier_id IN (200, 201, 203, 300);
```

This MySQL IN condition example would return all suppliers where the *supplier_id* is either 200, 201, 203, or 300.

The above IN example is equivalent to the following SELECT statement:

```
SELECT *
FROM suppliers
WHERE supplier_id = 200
OR supplier_id = 201
OR supplier_id = 203
OR supplier_id = 300;
```

Example - Using NOT operator

Finally, let's look at an IN condition example using the NOT operator.

For example:

```
SELECT *
FROM contacts
WHERE last_name NOT IN ('Johnson', 'Anderson', 'Smith');
```

This MySQL IN condition example would return all rows from the *contacts* table where the *last_name* is **not** Johnson, Anderson, or Smith. Sometimes, it is more efficient to list the values that you do **not** want, as opposed to the values that you do want.