HANDLING MYSQL NULL VALUES

http://www.tutorialspoint.com/mysql/mysql-null-values.htm Copyright © tutorialspoint.com

We have seen SQL **SELECT** command along with **WHERE** clause to fetch data from MySQL table, but when we try to give a condition, which compare field or column value to **NULL**, it does not work properly.

To handle such situation MySQL provides three operators

- **IS NULL:** operator returns true if column value is NULL.
- **IS NOT NULL:** operator returns true if column value is not NULL.
- **<=>:** operator compares values, which (unlike the = operator) is true even for two NULL values.

Conditions involving NULL are special. You cannot use = NULL or != NULL to look for NULL values in columns. Such comparisons always fail because it's impossible to tell whether or not they are true. Even NULL = NULL fails.

To look for columns that are or are not NULL, use IS NULL or IS NOT NULL.

Using NULL values at Command Prompt:

Suppose a table **tcount_tbl** in TUTORIALS database and it contains two columns **tutorial_author** and **tutorial_count**, where a NULL tutorial_count indicates that the value is unknown:

Example:

Try out the following examples:

```
root@host# mysql -u root -p password;
Enter password:*******
mysql> use TUTORIALS;
Database changed
mysql> create table tcount_tbl
    -> (
    -> tutorial_author varchar(40) NOT NULL,
    -> tutorial_count  INT
    -> );
Query OK, 0 rows affected (0.05 sec)
mysql> INSERT INTO tcount_tbl
    -> (tutorial_author, tutorial_count) values ('mahran', 20);
mysql> INSERT INTO tcount_tbl
    -> (tutorial_author, tutorial_count) values ('mahnaz', NULL);
mysql> INSERT INTO tcount_tbl
    -> (tutorial_author, tutorial_count) values ('Jen', NULL);
mysql> INSERT INTO tcount_tbl
    -> (tutorial_author, tutorial_count) values ('Gill', 20);

mysql> SELECT * from tcount_tbl;
+-----------------+----------------+
| tutorial_author | tutorial_count |
+-----------------+----------------+
| mahran          |             20 |
| mahnaz          |           NULL |
| Jen             |           NULL |
| Gill            |             20 |
+-----------------+----------------+
4 rows in set (0.00 sec)

mysql>
```

You can see that = and != do not work with NULL values as follows:

```
mysql> SELECT * FROM tcount_tbl WHERE tutorial_count = NULL;
Empty set (0.00 sec)
mysql> SELECT * FROM tcount_tbl WHERE tutorial_count != NULL;
```

```
Empty set (0.01 sec)
```

To find records where the tutorial_count column is or is not NULL, the queries should be written like this:

```
mysql> SELECT * FROM tcount_tbl
    -> WHERE tutorial_count IS NULL;
+-----------------+----------------+
| tutorial_author | tutorial_count |
+-----------------+----------------+
| mahnaz          |           NULL |
| Jen             |           NULL |
+-----------------+----------------+
2 rows in set (0.00 sec)
mysql> SELECT * from tcount_tbl
    -> WHERE tutorial_count IS NOT NULL;
+-----------------+----------------+
| tutorial_author | tutorial_count |
+-----------------+----------------+
| mahran          |             20 |
| Gill            |             20 |
+-----------------+----------------+
2 rows in set (0.00 sec)
```

Handling NULL Values in PHP Script:

You can use *if...else* condition to prepare a query based on NULL value.

Example:

Following example takes tutorial_count from outside and then compares it with the value available in the table.

```
<?php
$dbhost = 'localhost:3036';
$dbuser = 'root';
$dbpass = 'rootpassword';
$conn = mysql_connect($dbhost, $dbuser, $dbpass);
if(! $conn )
{
  die('Could not connect: ' . mysql_error());
}
if( isset($tutorial_count ))
{
   $sql = 'SELECT tutorial_author, tutorial_count
           FROM  tcount_tbl
           WHERE tutorial_count = $tutorial_count';
}
else
{
   $sql = 'SELECT tutorial_author, tutorial_count
           FROM  tcount_tbl
           WHERE tutorial_count IS $tutorial_count';
}

mysql_select_db('TUTORIALS');
$retval = mysql_query( $sql, $conn );
if(! $retval )
{
  die('Could not get data: ' . mysql_error());
}
while($row = mysql_fetch_array($retval, MYSQL_ASSOC))
{
    echo "Author:{$row['tutorial_author']}  <br> ".
         "Count: {$row['tutorial_count']} <br> ".
         "--------------------------------<br>";
}
echo "Fetched data successfully\n";
mysql_close($conn);
?>
```