UPDATE Table

The MySQL UPDATE statement is used to update columns of existing rows in a table with new values.

**Syntax :**

**Single table:**

UPDATE tableName

      SET col\_name1={expr1|DEFAULT}

[, col\_name2={expr2|DEFAULT}]...

      [WHERE where\_condition]

* For a single table, the UPDATE statement updates columns of existing rows in the named table with new values. Specific columns can be modified using the SET clause by supplying new values for that column.
* The WHERE clause can be used to specify the conditions those identify which rows to update. Without using WHERE clause, all rows are updated.

**MySQL UPDATE column**

MySQL UPDATE column can be used to update some specific columns. The following MySQL statement will update the 'receive\_qty' column of *newpurchase* table with a new value 20.

UPDATE newpurchase SET receive\_qty=20;

**MySQL UPDATE with WHERE**

MySQL UPDATE command can be used with WHERE clause to filter (against certain conditions) which rows will be updated. The following MySQL statement will update the 'receive\_qty' column of *newpurchase* table with a new value 25 if the value of purch\_price is more than 50.

UPDATE newpurchase

SET receive\_qty=25

WHERE purch\_price>50;

**MySQL UPDATE using NULL**

MySQL UPDATE command can be used to update a column value to NULL by setting column\_name = NULL, where column\_name is the name of the column to be updated. The following MySQL statement will update pub\_lang column with NULL if purch\_price is more than 50. In this statement, other columns are also updated with respective new values.

UPDATE newpurchase

SET receive\_qty=20,pub\_lang='Hindi',pub\_lang=NULL

WHERE purch\_price>50;

**MySQL UPDATE multiple columns**

MySQL UPDATE command can be used to update multiple columns by specifying a comma separated list of column\_name = new\_value. Where column\_name is the name of the column to be updated and new\_value is the new value with which the column will be updated. The following MySQL statement will update receive\_qty, pub\_lang, and receive\_dt columns with new values 20, Hindi and 2008-07-10 if purch\_price is more than 50.

UPDATE newpurchase

SET receive\_qty=20,pub\_lang='Hindi',receive\_dt='2008-07-10'

WHERE purch\_price>50;

**MySQL UPDATE with subqueries**

Here in the following, we have discussed how to use MySQL UPDATE command with subqueries.

The following MySQL statement will update purch\_price with purch\_price multiplied by 5 if it satisfies the condition defined in the subquery started with SELECT wrapped within a pair of parenthesis.

The subquery retrieves only those cate\_ids from purchase table if their corresponding receive\_qty is more than 10.

UPDATE newpurchase

SET purch\_price=purch\_price\*.05

WHERE cate\_id IN(SELECT cate\_id

FROM purchase

WHERE receive\_qty>10);