

Insert on Duplicate Key Update

The INSERT INTO statement in MySQL is used to insert new records into a specific table.

MySQL Insert on Duplicate Key Update Statement

When we are trying to insert a new row into a MySQL table column with a UNIQUE INDEX or PRIMARY KEY, MySQL will issue an error, if the value being inserted already exists in the column. This will happen because these constraints require unique values, and duplicate values are not allowed.

However, if we use the MySQL ON DUPLICATE KEY UPDATE clause with the INSERT INTO statement, MySQL will update the existing rows with the new values instead of showing an error.

Syntax

Following is the basic syntax of ON DUPLICATE KEY UPDATE clause in MySQL –

```
INSERT INTO my_table (col1, col2, ...)
VALUES (val1, val2), (val3, val4), ...
ON DUPLICATE KEY UPDATE <col1>=<val1>, <col2>=<val2>,...;
```

Example

First of all, let us create a table named CUSTOMERS using the following query –

```
CREATE TABLE CUSTOMERS (
  ID int NOT NULL,
  NAME varchar(20) NOT NULL,
  AGE int NOT NULL,
  ADDRESS char (25),
  SALARY decimal (18, 2),
  PRIMARY KEY (ID)
);
```

Here, we are inserting some records into the above-created table using the INSERT INTO statement as shown below –

```
INSERT INTO CUSTOMERS (ID,NAME,AGE,ADDRESS,SALARY) VALUES
(1, 'Ramesh', 32, 'Ahmedabad', 2000.00 ),
(2, 'Khilan', 25, 'Delhi', 1500.00 ),
(3, 'Kaushik', 23, 'Kota', 2000.00 );
```

Execute the following query to display the records present in the above created CUSTOMERS table –

```
INSERT INTO CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)
VALUES (3, 'Chaitali', 25, 'Mumbai', 6500.00);
```

We can avoid the above error and update the existing row with the new information using the **ON DUPLICATE KEY UPDATE** clause along with INSERT INTO statement as shown below –

```
INSERT INTO CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)
VALUES (3, 'Chaitali', 25, 'Mumbai', 6500.00)
ON DUPLICATE KEY UPDATE NAME = "Chaitali",
AGE = 25,
ADDRESS = "Mumbai",
SALARY = 6500.00;
```

```
SELECT * FROM CUSTOMERS;
```

Example

In the following query, we are trying to insert a new row into the **CUSTOMERS** table using the INSERT INTO statement along with the ON DUPLICATE KEY UPDATE clause –

```
INSERT INTO CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)
VALUES (4, 'Hardik', 27, 'Bhopal', 8500.00)
ON DUPLICATE KEY UPDATE NAME = "Hardik",
AGE = 27,
ADDRESS = "Bhopal",
SALARY = 8500.00;
```

```
SELECT * FROM CUSTOMERS;
```

INSERT or UPDATE multiple records at once

While inserting or updating multiple records at the same time in MySQL, the value to set for each column may vary depending on which record or records have a conflict.

For example, if we are trying to insert four new rows, but the third has an **ID** column that conflicts with an existing record, we most likely want to update the existing row based on the data you had in mind for the third row.

Example

Before we perform the next operation, let's look into the records of updated CUSTOMERS table –

```
SELECT * FROM CUSTOMERS;
```

The following query adds two new rows into the CUSTOMERS table –

```
INSERT INTO CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)
```

```
VALUES (5, "Komal", 22, "Hyderabad", 4500.00),
```

```
(4, "Kaushik", 23, "Kota", 2000.00)
```

```
ON DUPLICATE KEY UPDATE
```

```
NAME = VALUES(NAME),
```

```
AGE = VALUES(AGE),
```

```
ADDRESS = VALUES(ADDRESS),
```

```
SALARY = VALUES(SALARY);
```

```
SELECT * FROM CUSTOMERS;
```

Client Program

```
insert_on_duplicate_key_update_query = "INSERT INTO table_name (column1, column2, ...)
VALUES (value1, value2, ...)
ON DUPLICATE KEY UPDATE
column1 = VALUES(column1), column2 = VALUES(column2), ..."
cursorObj.execute(insert_on_duplicate_key_update_query)
```

```
import mysql.connector
#establishing the connection
connection = mysql.connector.connect(
    host='localhost',
    user='root',
    password='password',
    database='tut'
)
cursorObj = connection.cursor()
insert_on_duplicate_key_update = "INSERT INTO tutorials_tbl (tutorial_id, tutorial_title,
tutorial_author, submission_date) VALUES (7, 'New Tutorial', 'John Doe', '2023-07-25') ON DUPLICATE
KEY UPDATE tutorial_title='Updated Tutorial', tutorial_author='Jane Smith', submission_date='2023-
07-28'"
cursorObj.execute(insert_on_duplicate_key_update)
connection.commit()
print("INSERT on duplicate key update query executed successfully.")
cursorObj.close()
connection.close()
```