MySQLTUTORIAL

MySQL INSERT INTO SELECT

Summary: in this tutorial, you will learn how to use the MySQL `INSERT INTO SELECT` statement to insert data into a table, where data comes from the result of a `SELECT` statement.

MySQL INSERT INTO SELECT Overview

In the previous tutorial, you learned how to insert one or more rows into a table using the `INSERT` (https://www.mysqltutorial.org/mysql-insert-statement.aspx) statement with a list of column values specified in the `VALUES` clause.

```
INSERT INTO table_name(c1,c2,...)
VALUES(v1,v2,..);
```

Besides using row values in the `VALUES` clause, you can use the result of a `SELECT` (https://www.mysqltutorial.org/mysql-select-statement-query-data.aspx) statement as the data source for the `INSERT` statement.

The following illustrates the syntax of the `INSERT INTO SELECT` statement:

```
INSERT INTO table_name(column_list)
SELECT
   select_list
FROM
   another_table
WHERE
   condition;
```

In this syntax, instead of using the `VALUES` clause, you can use a `SELECT` statement. The `SELECT` statement can retrieve data from one or more tables.

The `INSERT INTO SELECT` statement is very useful when you want to copy data from other tables to a table or to summary data from multiple tables into a table.

MySQL INSERT INTO SELECT example

First, create a new table (https://www.mysqltutorial.org/mysql-create-table/) called `suppliers` :

```
CREATE TABLE suppliers (
    supplierNumber INT AUTO_INCREMENT,
    supplierName VARCHAR(50) NOT NULL,
    phone VARCHAR(50),
    addressLine1 VARCHAR(50),
    addressLine2 VARCHAR(50),
    city VARCHAR(50),
    state VARCHAR(50),
    postalCode VARCHAR(50),
    country VARCHAR(50),
    customerNumber INT,
    PRIMARY KEY (supplierNumber)
);
```

Note that you will learn how to create a new table (https://www.mysqltutorial.org/mysql-create-table/) in the subsequent tutorial. For now, you just need to execute this statement to create the `suppliers` table.

Suppose all customers from `California, USA` become the company's suppliers. The following query finds all customers who locate in California, USA:

```
SELECT
    customerNumber,
    customerName,
    phone,
    addressLine1,
    addressLine2,
    city,
    state,
    postalCode,
    country
FROM
    customers
WHERE
```

```
    country = 'USA' AND
    state = 'CA';
```

Second, use the `INSERT INTO ... SELECT` statement to insert customers who locate in `California USA` from the `customers` table into the `suppliers` table:

```
INSERT INTO suppliers (
    supplierName,
    phone,
    addressLine1,
    addressLine2,
    city,
    state,
    postalCode,
    country,
    customerNumber
)
SELECT
    customerName,
    phone,
    addressLine1,
    addressLine2,
    city,
    state ,
    postalCode,
    country,
    customerNumber
FROM
```

```
    customers
WHERE
    country = 'USA' AND
    state = 'CA';
```

It returned the following message indicating that 11 rows have been inserted successfully.

```
11 row(s) affected Records: 11  Duplicates: 0  Warnings: 0
```

Third, verify the insert by querying data from the `suppliers` table:

```
SELECT * FROM suppliers;
```

Here is the output:

Using SELECT statement in the VALUES list

First, create a new table (https://www.mysqltutorial.org/mysql-create-table/) called `stats`:

```
CREATE TABLE stats (
    totalProduct INT,
    totalCustomer INT,
    totalOrder INT
);
```

Second, use the `INSERT` (https://www.mysqltutorial.org/mysql-insert-statement.aspx) statement to insert values that come from the `SELECT` statements:

```
INSERT INTO stats(totalProduct, totalCustomer, totalOrder)
VALUES(
        (SELECT COUNT(*) FROM products),
        (SELECT COUNT(*) FROM customers),
        (SELECT COUNT(*) FROM orders)
);
```

In this example:

- First, use the `SELECT` statements with the `COUNT()` functions to get the total products, employees, and orders.
- Second, use the values returned from the `SELECT` (https://www.mysqltutorial.org/mysql-select-statement-query-data.aspx) statement in place of values in the `VALUES` clause of the `INSERT` (https://www.mysqltutorial.org/mysql-insert-statement.aspx) statement.

Third, query data from the table `stats`:

```
SELECT * FROM stats;
```

In this tutorial, you have learned how to use the MySQL `INSERT INTO SELECT` statement to insert data into a table from a result set.