LAB - DELETE

In this lab, you will learn how to use SQL Server `DELETE` statement to remove one or more rows from a table.

To remove one or more rows from a table completely, you use the `DELETE` statement. The following illustrates its syntax:

```
DELETE [ TOP ( expression ) [ PERCENT ] ]
FROM table_name
[WHERE search_condition];
```

First, you specify the name of the table from which the rows are to be deleted in the `FROM` clause.

For example, the following statement will delete all rows from the `target_table`:

```
DELETE FROM target_table;
```

Second, to specify the number or percent of random rows that will be deleted, you use the `TOP` clause.

For example, the following `DELETE` statement removes 10 random rows from the `target_table`:

```
DELETE TOP 10 FROM target_table;
```

Because the table stores its rows in unspecified order, we do not know which rows will be deleted but we know for sure that the number of rows will be deleted is 10.

Similarly, you can delete the 10 percent of random rows by using the following `DELETE` statement:

```
DELETE TOP 10 PERCENT FROM target_table;
```

Third, practically speaking, you will rarely remove all rows from a table but only one or several rows. In this case, you need to specify the `search_condition` in the `WHERE` clause to limit the number of rows that are deleted.

The rows that cause the `search_condition` evaluates to true will be deleted.

The `WHERE` clause is optional. If you skip it, the `DELETE` statement will remove all rows from the table.

Examples

Let’s create a new table for the demonstration.

The following statement creates a table named `production.product_history` with the data copied from the `production.products` table:

```
SELECT *
INTO production.product_history
FROM
    production.products;
```

The following query returns all rows from the `product_history` table:

```
SELECT * FROM production.product_history;
```

As can be seen clearly in the output, we have 321 rows in total.

1) Delete the number of random rows example

The following `DELETE` statement removes 21 random rows from the `product_history` table:

```
DELETE TOP (21)
FROM production.product_history;
```

Here is the message issued by the SQL Server:

```
(21 rows affected)
```

It means that 21 rows have been deleted.

2) Delete the percent of random rows example

The following `DELETE` statement removes 5 percent of random rows from the `product_history` table:

```
DELETE TOP (5) PERCENT
FROM production.product_history;
```

SQL Server issued the following message indicating that 15 rows (300 x 5% = 15) have been deleted.

```
(15 rows affected)
```

3) Delete some rows with a condition example

The following `DELETE` statement removes all products whose model year is 2017:

```
DELETE
FROM
    production.product_history
WHERE
    model_year = 2017;
```

Here is the output message:

```
(75 rows affected)
```

4) Delete all rows from a table example

The following `DELETE` statement removes all rows from the `product_history` table:

```
DELETE FROM  production.product_history;
```

Note that if you want to remove all rows from a big table, you should use the `TRUNCATE TABLE` statement which is faster and more efficient.

In this lab, you have learned how to use the SQL Server `DELETE` statement to remove one or more rows from a table.