

Foreign Key with Set Null in SQL Server

The following tutorial will explain the Foreign Key foreign key with the Set Null On Delete constraint in SQL Server.

The following tutorial will explain the Foreign Key foreign key with the Set Null On Delete constraint in SQL Server.

What is the foreign key with Set NULL on Delete in SQL Server?

Foreign keys are bound to set Null on Delete meaning that when a record in the parent table is deleted, the corresponding record in the child table in the foreign key value field will be set to NULL. The table written in the child table will not be deleted from SQL Server.

Foreign keys provided that the NULL value when deleted can be created with the CREATE TABLE or ALTER TABLE statement.

Create foreign key with constraint to set NULL value when deleted with CREATE TABLE statement

Syntax

```
CREATE TABLE bang_con
(
cot1 kieudulieu [ NULL | NOT NULL ],
cot2 kieudulieu [ NULL | NOT NULL ],
...

CONSTRAINT fk_ten
FOREIGN KEY (cot_con1, cot_con2, ... cot_con_n)
REFERENCES bang_me (cot_me1, cot_me2, ... cot_me_n)
ON DELETE SET NULL
[ ON UPDATE { NO ACTION | CASCADE | SET NULL | SET DEFAULT } ]
);
```

bang_con

The name of the child table you want to create.

cot1, cot2

The column you want to create in the table. Each column has 1 data type, must be specified to contain NULL or NOT NULL values, otherwise default is NULL.

Data types in SQL Server

fk_ten

The name of the foreign key constraint to create.

cot_con1, cot_con2, . cot_con_n

The column in bang_con wants to refer to the primary key in bang_me.

bang_me

The name of the parent table contains the primary key used in bang_con.

cot_me1, cot_me2, . cot_me_n

The column that makes up the primary key in bang_me. Foreign keys will create constraints between data and columns cot_con1, cot_con2, . cot_con_n in bang_con.

ON DELETE SET NULL

The data in the child table will be set to NULL when the data in the parent table is deleted. Sub-data is not deleted.

ON UPDATE

Option. Indicates what to do with the child data when the parent data is updated. There are options NO ACTION, CASCADE, SET NULL and SET DEFAULT.

NO ACTION

Use with ON DELETE or ON UPDATE, ie do nothing with child data when parent data is deleted or updated.

CASCADE

Use with ON DELETE or ON UPDATE, meaning that the child data is deleted or updated when the parent data is deleted or updated.

SET NULL

Use with ON DELETE or ON UPDATE, meaning that the child data is set to NULL when the parent data is deleted or updated.

SET DEFAULT

Use with ON DELETE or ON UPDATE, meaning that the child data is set to the default value when the parent data is deleted or updated.

For example

```
CREATE TABLE sanpham
( id_sanpham INT PRIMARY KEY,
  ten_sanpham VARCHAR(50) NOT NULL,
  phan_loai VARCHAR(25)
);
```

```
CREATE TABLE hangtonkho
( id_hangtonkho INT PRIMARY KEY,
  id_sanpham INT,
  soluong INT,
  luong_toithieu INT,
  luong_toida INT,
  CONSTRAINT fk_htk_id_sanpham
  FOREIGN KEY (id_sanpham)
  REFERENCES sanpham (id_sanpham)
  ON DELETE SET NULL
);
```

In this example, we created the parent table, sanpham, with the primary key including the information field id_sanpham. Then there is a child table named hangtonkho with a foreign key with deletion constraint. The CREATE TABLE statement creates a foreign key on the hangtonkho table named fk_htk_id_sanpham. The foreign key forms the relationship between the id_sanpham column in the hangtonkho table and id_sanpham in the sanpham table.

This foreign key is specified ON DELETE SET NULL to tell SQL Server that it is necessary to set the corresponding record value in the child table to NULL when the data in the parent table is deleted. In this example, when id_sanpham is deleted from the table, the corresponding record in the hangtonkho table using id_sanpham will be set to NULL.

It is important to note that because the id_sanpham column in the hangtonkho table will be set to NULL, it is necessary to ensure that this column is allowed to receive NULL values. If set to NOT NULL when created with CREATE TABLE, the following error will be encountered.

```
CREATE TABLE hangtonkho
( id_hangtonkho INT PRIMARY KEY,
  id_sanpham INT NOT NULL,
  soluong INT,
  luong_toithieu INT,
  luong_toida INT,
  CONSTRAINT fk_htk_id_sanpham
  FOREIGN KEY (id_sanpham)
  REFERENCES sanpham (id_sanpham)
  ON DELETE SET NULL
);
```

Msg 1761, Level 16, State 0, Line 1

Cannot create the foreign key 'fk_htk_id_sanpham' with the SET NULL referential

Msg 1750, Level 16, State 0, Line 1

Could th? t?o constraint ho?c ch? m?c. Xem l?i tr??c.

Make sure you define the id_sanpham column in the hangtonkho table to get NULL values ??as shown below.

```
CREATE TABLE hangtonkho
( id_hangtonkho INT PRIMARY KEY,
  id_sanpham INT,
  soluong INT,
  luong_toithieu INT,
  luong_toida INT,
  CONSTRAINT fk_htk_id_sanpham
  FOREIGN KEY (id_sanpham)
  REFERENCES sanpham (id_sanpham)
  ON DELETE SET NULL
);
```

Command (s) completed successfully.

Creating a foreign key with the constraint sets the NULL value when deleting with the ALTER TABLE statement

Syntax

```
ALTER TABLE state_con
ADD CONSTRAINT fk_ten
FOREIGN KEY (cot_con1, cot_con2, . cot_con_n)
REFERENCES bang_me (cot_me1, cot_me2, . cot_me_n)
ON DELETE SET NULL;
```

bang_con

The name of the child table you want to create.

fk_ten

The name of the foreign key constraint to create.

cot_con1, cot_con2, . cot_con_n

The column in bang_con wants to refer to the primary key in bang_me.

bang_me

The name of the parent table contains the primary key used in bang_con.

cot_me1, cot_me2, . cot_me_n

The column that makes up the primary key in bang_me. Foreign keys will create constraints between data and columns cot_con1, cot_con2, . cot_con_n in bang_con.

ON DELETE SET NULL

Specifies that the child data will be set to NULL value when the data in the parent table is deleted. Data in the child table will not be deleted.

For example

```
ALTER TABLE hangtonkho
ADD CONSTRAINT fk_htk_id_sanpham
FOREIGN KEY (id_sanpham)
REFERENCES sanpham (id_sanpham)
ON DELETE SET NULL;
```

In this example, the hangtonkho sub table is created with a foreign key named fk_htk_id_sanpham, referring to the parent table sanpham based on id_sanpham.

Specifying ON DELETE SET NULL for SQL Server to understand that when id_sanpham data in the parent table is deleted, the corresponding record in the hangtonkho sub table will be set to NULL value.

Previous lesson: Foreign Key (Cascade Delete) in SQL Server

The following article: Delete the foreign key in SQL Server

You finished reading the article "**Foreign Key with Set Null in SQL Server**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.