MySQL CHECK Constraint

Summary: in this tutorial, you will learn how to use MySQL `CHECK` constraint to ensure that values stored in a column or group of columns satisfy a Boolean expression.

> MySQL 8.0.16 implemented the SQL check constraint. If you use MySQL with the earlier versions, you can emulate a `CHECK` constraint using a view `WITH CHECK OPTION` or a trigger.

Introduction to the MySQL CHECK constraint

Before MySQL 8.0.16, the `CREATE TABLE` allows you to include a table `CHECK` constraint. However, MySQL ignores all the `CHECK` constraints:

```
CHECK(expression)
```

As of MySQL 8.0.16, the `CREATE TABLE` supported essential features of table and column `CHECK` constraints for all storage engines.

Here is the basic syntax:

```
CONSTRAINT constraint_name
CHECK (expression)
[ENFORCED | NOT ENFORCED]
```

In this syntax:

First, specify the name for the check constraint that you want to create after the `CONSTRAINT` keyword. If you omit the constraint name, MySQL automatically generates a name with the following convention:

```
table_name_chk_n
```

In this convention, `n` is an ordinal number such as 1,2, and 3. For example, the automatically generated names of `CHECK` constraints of the `parts` table will be `parts_chk_1`, `parts_chk_2`, and so on.

Second, specify a Boolean `expression` which must be evaluated to `TRUE` or `UNKNOWN` for each row of the table inside the parentheses after the `CHECK` keyword.

If the expression evaluates to `FALSE`, the values violate the constraint or a constraint violation occurs.

> Note that MySQL treats 1 as true and 0 as false.

Third, optionally specify the enforcement clause to indicate whether the check constraint is enforced:

- Use `ENFORCED` or omit the `ENFORCED` clause to create and enforce the constraint.
- Use `NOT ENFORCED` to create the constraint but not enforce it.

As mentioned earlier, you can define a `CHECK` constraint as a table constraint or column constraint.

A table `CHECK` constraint can reference multiple columns whereas the column `CHECK` constraint can refer to the only column where it is defined.

MySQL CHECK constraint examples

Let's take some examples of using the `CHECK` constraints.

1) Creating CHECK constraints as column constraints

The following CREATE TABLE statement creates a new table called `parts`:

```
CREATE TABLE parts (
    part_no VARCHAR(18) PRIMARY KEY,
    description VARCHAR(40),
    cost DECIMAL(10,2 ) NOT NULL CHECK (cost >= 0),
    price DECIMAL(10,2) NOT NULL CHECK (price >= 0)
);
```

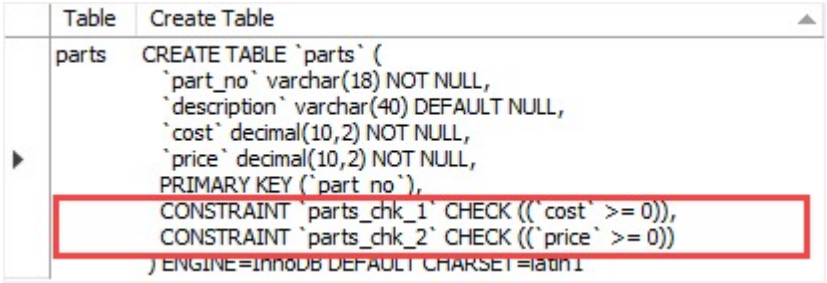
The parts table has two column `CHECK` constraints: one for the `cost` column and the other for the `price` column.

Because we did not explicitly specify the names of the `CHECK` constraints, MySQL automatically generated names for them.

To view the table definition with the `CHECK` constraint name, you use the `SHOW CREATE TABLE` statement:

```
SHOW CREATE TABLE parts;
```

Output:



The output indicates that MySQL generated the names (`parts_chk_1` and `parts_chk_2`) for the check constraints.

After creating `CHECK` constraints, if you insert or update a value that causes the Boolean expression to be false, MySQL rejects the change and issues an error.

This statement inserts a new row into the `parts` table:

```
INSERT INTO parts(part_no, description,cost,price)
VALUES('A-001','Cooler',0,-100);
```

MySQL issued an error:

```
Error Code: 3819. Check constraint 'parts_chk_2' is violated.
```

Because the value of the `price` column is negative which causes the expression `price > 0` evaluates to `FALSE` that results in a constraint violation.

2) Creating CHECK constraints as a table constraints

First, drop the `parts` table:

```
DROP TABLE IF EXISTS parts;
```

Then, create a new `parts` table with one more table `CHECK` constraint:

```
CREATE TABLE parts (
    part_no VARCHAR(18) PRIMARY KEY,
    description VARCHAR(40),
    cost DECIMAL(10,2 ) NOT NULL CHECK (cost >= 0),
    price DECIMAL(10,2) NOT NULL CHECK (price >= 0),
    CONSTRAINT parts_chk_price_gt_cost
        CHECK(price >= cost)
);
```

The following new clause defines a table `CHECK` constraint that ensures the price is always greater than or equal to the cost:

```
CONSTRAINT parts_chk_price_gt_cost CHECK(price >= cost)
```

Because we explicitly specify the name of the `CHECK` constraint, MySQL creates the new constraint with the specified name.

Here is the definition of the `parts` table:

```
SHOW CREATE TABLE parts;
```

The table `CHECK` constraint appears at the end of the table definition after the column list.

This statement attempts to insert a new part whose price is less than the cost:

```
INSERT INTO parts(part_no, description,cost,price)
VALUES('A-001','Cooler',200,100);
```

Here is the error due to the constraint violation:

```
Error Code: 3819. Check constraint 'parts_chk_price_gt_cost' is violated.
```

Adding a check constraint to a table

To add a check constraint to an existing table, you use the `ALTER TABLE ... ADD CHECK` statement:

```
ALTER TABLE table_name
ADD CHECK (expression);
```

If you want to explicitly specify the name of the `CHECK` constraint, you can use the `ALTER TABLE ... ADD CONSTRAINT ... CHECK` statement:

```
ALTER TABLE table_name
ADD CONSTRAINT contraint_name
CHECK (expression);
```

For example, the following statement adds a `CHECK` constraint to the `parts` table:

```
ALTER TABLE parts
ADD CHECK (part_no <> description);
```

This `CHECK` constraint prevents you from having the `part_no` identical to the `description`.

For example, the following `INSERT` statement will be rejected:

```
INSERT INTO parts
VALUES('A','A',100,120);
```

Output:

```
ERROR 3819 (HY000): Check constraint 'parts_chk_3' is violated.
```

Removing a check constraint from a table

To remove a `CHECK` constraint from a table, you use the `ALTER TABLE ... DROP CHECK` statement:

```
ALTER TABLE table_name
DROP CHECK constraint_name;
```

For example, the following statement removes the `CHECK` constraint `parts_chk_3` from the parts table:

```
ALTER TABLE parts
DROP CHECK parts_chk_3;
```

Summary

- Use `CHECK` constraints to ensure values stored in a column satisfy a Boolean condition.
- Use the `CHECK(expression)` to define a `CHECK` constraint.
- Use the `ALTER TABLE ... ADD CHECK` to add a `CHECK` constraint to a table.

- Use the `ALTER TABLE ... DROP CHECK` to remove a `CHECK` constraint from a table.

Was this tutorial helpful?

PREVIOUSLY
MySQL DEFAULT

UP NEXT
MySQL INSERT

Search ...

GETTING STARTED

What Is MySQL?

Install MySQL Database Server

Connect to MySQL Server

Download MySQL Sample Database

Load Sample Database

QUERYING DATA

SELECT FROM

SELECT

ORDER BY

WHERE

SELECT DISTINCT

AND

OR

IN

NOT IN

BETWEEN

LIKE

LIMIT

IS NULL

Table & Column Aliases

Joins

INNER JOIN

LEFT JOIN

RIGHT JOIN

Self Join

CROSS JOIN

GROUP BY

HAVING

HAVING COUNT

ROLLUP

Subquery

Derived Tables

EXISTS

EXCEPT

INTERSECT

MANAGING DATABASES

Select a Database

Create Databases

Drop Databases

MANAGING TABLES

Create Tables

AUTO_INCREMENT

Rename Tables

Add Columns

Drop Columns

Drop Tables

Temporary Tables

Generated Columns

MYSQL CONSTRAINTS

Primary Key

Foreign Key

Disable Foreign Key Checks

UNIQUE Constraint

NOT NULL Constraint

DEFAULT Constraint

CHECK Constraint

INSERT DATA

Insert Into

Insert Multiple Rows

INSERT INTO SELECT

Insert On Duplicate Key Update

INSERT IGNORE

Insert DateTimes

Insert Dates

UPDATE DATA

UPDATE

UPDATE JOIN

DELETE DATA

DELETE JOIN

ON DELETE CASCADE

TRUNCATE TABLE

MYSQL TRANSACTIONS

Table Locking

MYSQL DATA TYPES

BIT

INT

BOOLEAN

DECIMAL

DATETIME

TIMESTAMP

DATE

TIME

CHAR

VARCHAR

TEXT

BINARY

VARBINARY

ENUM

BLOB

MYSQL GLOBALIZATION

MySQL Character Sets

MySQL Collation

MYSQL IMPORT & EXPORT

Import a CSV File Into a Table

Export a Table to a CSV File

MySQLTutorial.org helps you master MySQL quickly, easily, and with enjoyment. Our tutorials make learning MySQL a breeze.

All MySQL tutorials are clear, practical and easy-to-follow.

More About Us

MySQL Commands

innodb_dedicated_server: Configure InnoDB Dedicated Server

innodb_flush_method: Configure InnoDB Flush Method

innodb_log_buffer_size: Configure InnoDB Log Buffer Size

innodb_buffer_pool_chunk_size: Configure Buffer Pool Chunk Size

innodb_buffer_pool_instances: Configuring Multiple Buffer Pool Instances for Improved Concurrency in MySQL

innodb_buffer_pool_size: Configure InnoDB Buffer Pool Size

MySQL InnoDB Architecture

How to Kill a Process in MySQL

Contact Us

About

Privacy Policy

OTHERS

MySQL Cheat Sheet

MySQL Resources

MySQL Books