



MySQL DROP COLUMN

Summary: in this tutorial, you will learn how to drop a column from a table using the MySQL `DROP COLUMN` statement.

Introduction to MySQL DROP COLUMN statement

In some situations, you want to remove one or more columns from a table. In such cases, you use the following `ALTER TABLE DROP COLUMN` statement:

```
ALTER TABLE table_name
DROP COLUMN column_name;
```

In this syntax:

- First, specify the name of the table that contains the column which you want to drop after the `ALTER TABLE` keywords.
- Second, specify the name of the column that you want to drop in the `DROP COLUMN` clause.

Note that the keyword `COLUMN` keyword in the `DROP COLUMN` clause is optional so you can use the shorter statement as follows:

```
ALTER TABLE table_name
DROP column_name;
```

To remove multiple columns from a table using a single `ALTER TABLE` statement, you use the following syntax:

```
ALTER TABLE table_name
DROP COLUMN column_name_1,
DROP COLUMN column_name_2,
...;
```

There are some important points you should remember before removing a column from a table:

- Removing a column from a table makes all database objects such as [stored procedures](https://www.mysqltutorial.org/mysql-stored-procedure-tutorial.aspx) (<https://www.mysqltutorial.org/mysql-stored-procedure-tutorial.aspx>) , [views](https://www.mysqltutorial.org/mysql-views-tutorial.aspx) (<https://www.mysqltutorial.org/mysql-views-tutorial.aspx>) , and [triggers](https://www.mysqltutorial.org/mysql-triggers.aspx) (<https://www.mysqltutorial.org/mysql-triggers.aspx>) that referencing the dropped column invalid. For example, you may have a stored procedure that refers to a column. When you remove the column, the stored procedure becomes invalid. To fix it, you have to manually change the stored procedure's code.
- The code from other applications that depends on the dropped column must be also changed, which takes time and efforts.
- Dropping a column from a large table can impact the performance of the database during the removal time.

MySQL DROP COLUMN examples

First, [create a table](https://www.mysqltutorial.org/mysql-create-table/) (<https://www.mysqltutorial.org/mysql-create-table/>) named `posts` for the demonstration.

```
CREATE TABLE posts (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    title VARCHAR(255) NOT NULL,  
    excerpt VARCHAR(400),  
    content TEXT,  
    created_at DATETIME,  
    updated_at DATETIME  
);
```

Next, use the `ALTER TABLE DROP COLUMN` statement to remove the `excerpt` column:

```
ALTER TABLE posts  
DROP COLUMN excerpt;
```

Then, view the table structure using the `DESCRIBE` statement:

```
DESCRIBE posts;
```

After that, use the `ALTER TABLE DROP COLUMN` statement to drop the `created_at` and `updated_at` columns:

```
ALTER TABLE posts
DROP COLUMN created_at,
DROP COLUMN updated_at;
```

Finally, use the `DESCRIBE` statement to verify the removal:

```
DESCRIBE posts;
```

MySQL drop a column which is a foreign key example

If you remove the column that is a [foreign key](https://www.mysqltutorial.org/mysql-foreign-key/) (<https://www.mysqltutorial.org/mysql-foreign-key/>), MySQL will issue an error. Consider the following example.

First, [create a table](https://www.mysqltutorial.org/mysql-create-table/) (<https://www.mysqltutorial.org/mysql-create-table/>) named `categories` :

```
CREATE TABLE categories (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(255)
);
```

Second, [add a column](https://www.mysqltutorial.org/mysql-add-column/) (<https://www.mysqltutorial.org/mysql-add-column/>) named `category_id` to the `posts` table.

```
ALTER TABLE posts  
ADD COLUMN category_id INT NOT NULL;
```

Third, make the `category_id` column as a foreign key column of that references to the `id` column of the `categories` table.

```
ALTER TABLE posts  
ADD CONSTRAINT fk_cat  
FOREIGN KEY (category_id)  
REFERENCES categories(id);
```

Fourth, drop the `category_id` column from the `posts` table.

```
ALTER TABLE posts  
DROP COLUMN category_id;
```

MySQL issued an error message:

```
Error Code: 1553. Cannot drop index 'fk_cat': needed in a foreign key constraint
```

To avoid this error, you must remove the foreign key constraint before dropping the column.

In this tutorial, we have shown you how to use MySQL `DROP COLUMN` statement to remove one or more columns from a table.