How to Copy Table in MySQL: Duplicate Structure, Data, and Indexes Correctly

Copying tables in MySQL is a routine operation performed by DBAs, developers, and analysts dozens of times a day for various reasons and with various purposes. In this tutorial, we provide a detailed overview of the most common methods to copy a MySQL table structure and data.

Read the guide to find out:

* Ways to copy MySQL table definition in MySQL
* SQL examples to copy a table in MySQL
* How to copy table data and structure to another database
* How to duplicate a MySQL table including indexes and data
* How to insert table data into an already existing table
* The fastest way to copy a table—by using the MySQL GUI tool

How to copy a table in MySQL

MySQL doesn't have the copy table statement, which means you'll have to use sideways approaches to perform the operation. There are three popular ways to clone a table in MySQL.

1. CREATE TABLE ... AS SELECT statement to copy the source table column attributes and data.

CREATE TABLE new\_table\_name

AS

SELECT

\*

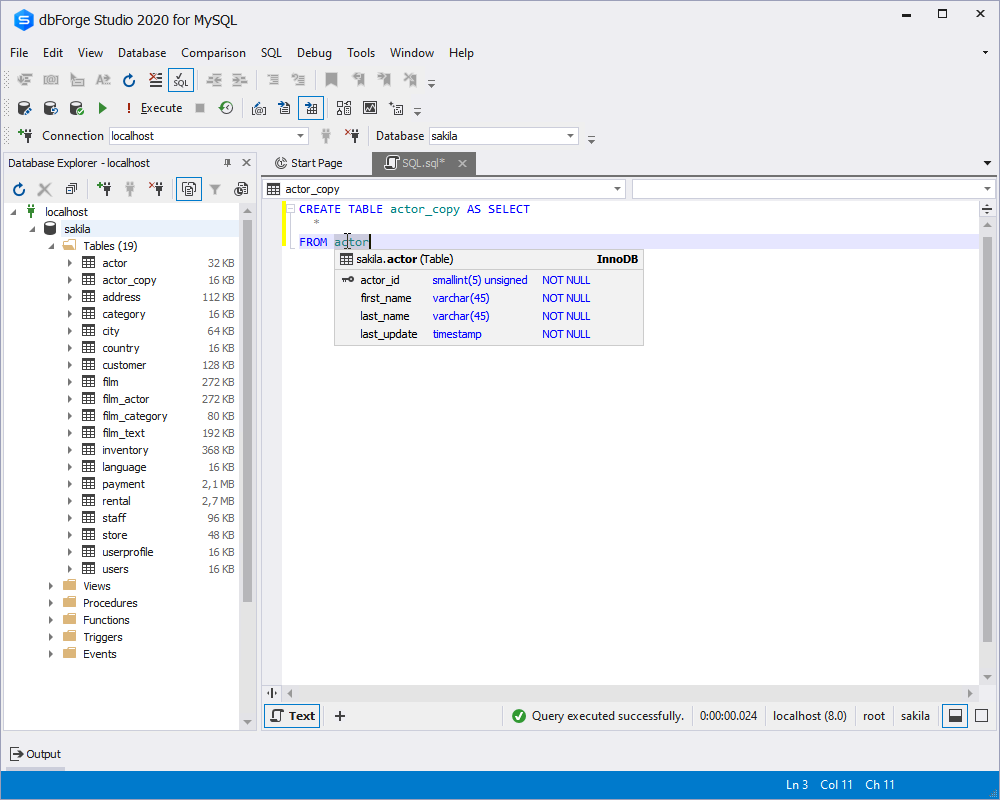
FROM old\_table\_name;

2. CREATE TABLE ... LIKE statement to create an empty table based on the definition of the original table, including column attributes and indexes.

CREATE TABLE new\_table\_name LIKE old\_table\_name;

3. SHOW CREATE TABLE to generate a create table script for the original table.

SHOW CREATE TABLE customer\G



MуSQL copy table structure only

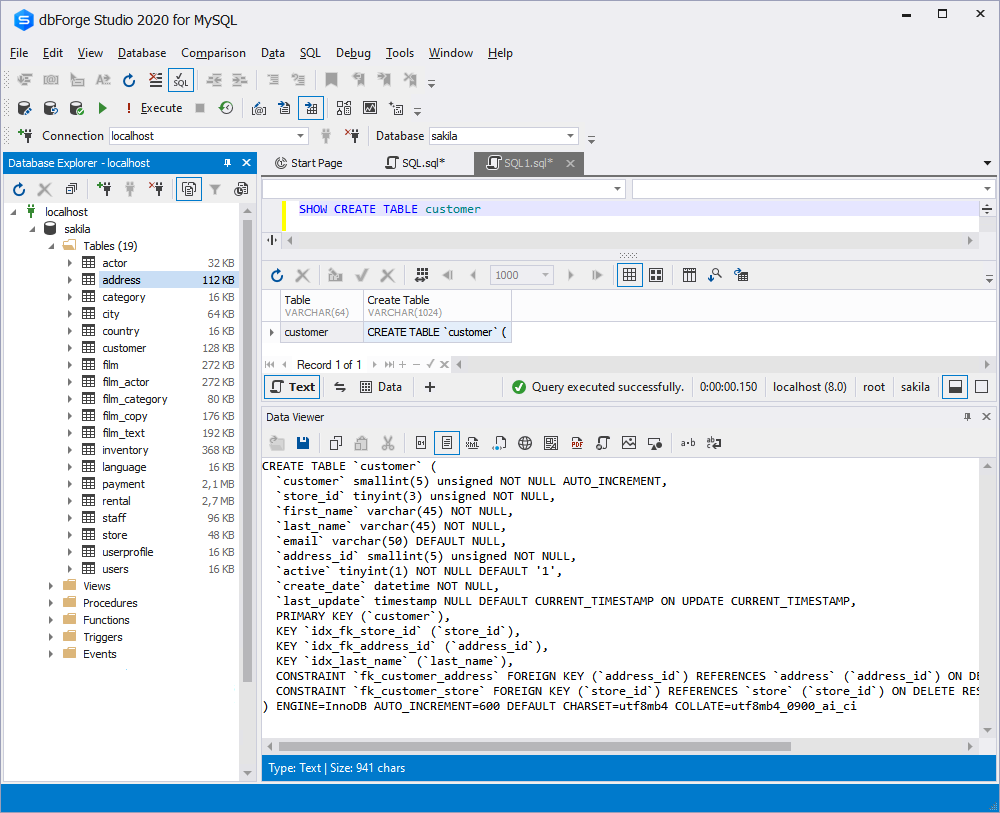
Let's now take a closer look at ways to create a table in MySQL by using SQL statements.

If you need to duplicate the table structure, but not its data, it is better to use the CREATE TABLE ... LIKE statement.

Suppose, we need to clone the structure of the customer table. The query to copy structure is as follows:

CREATE TABLE customer\_new LIKE customer

In case you need to copy the table structure including primary keys, foreign keys and constraints, run the SHOW CREATE TABLE ... statement, then copy the script for the original table, change the name of the table and execute the script.



Copy table data to another table

To copy data from one table to another, use the INSERT INTO statement. Note, that you can specify the columns to be copied.

INSERT INTO target\_table (`column1`, `column2`)

SELECT

`column1`,

`column2`

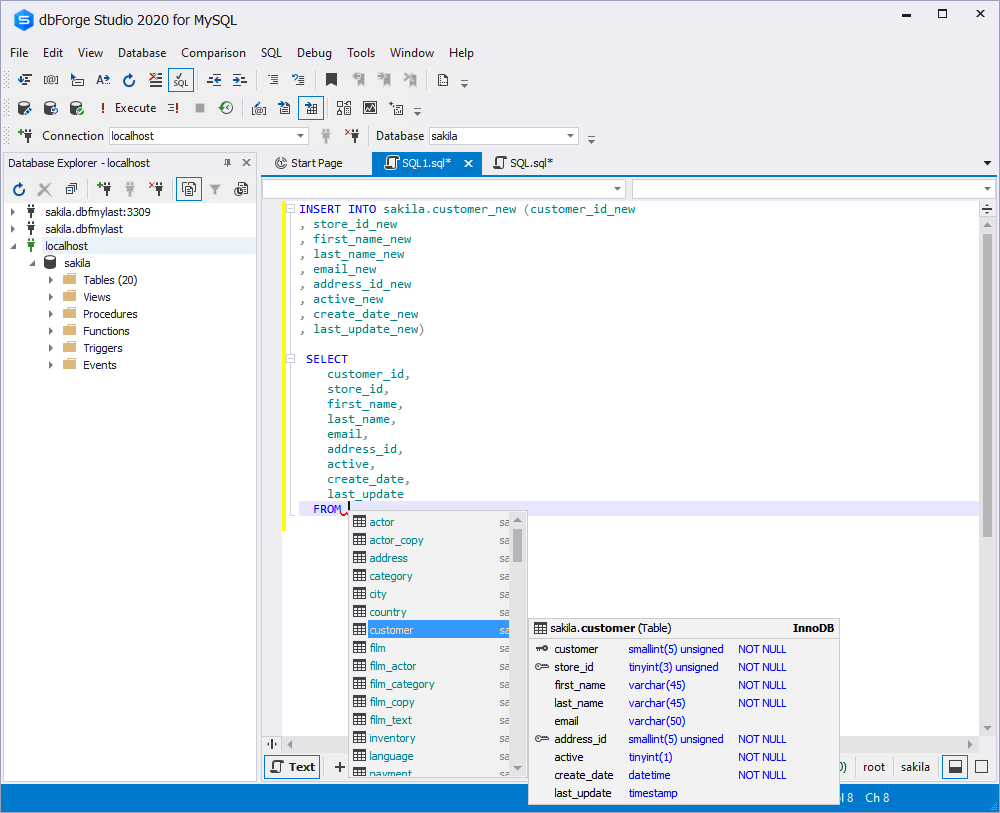
FROM source\_table;

This method works perfectly well if you have already copied the table structure like we did in the example above and now need to copy values.

In case you want to duplicate all rows from the source table, use the following syntax:

CREATE TABLE new\_table\_name LIKE old\_table\_name;

INSERT new\_table\_name SELECT \* FROM old\_table\_name;



Copy a table from one database to another

In MySQL, the easiest way to copy a table with its data between two databases is to use the CREATE TABLE AS statement, but note, that you need to provide the target database name as a table prefix.

CREATE TABLE new-database-name.new-table-name

AS

SELECT \* FROM old-database.old-table-name;

If you need to copy to another database schema only the table structure, use the CREATE TABLE LIKE statement, but again don't forget to specify the database names. Remember that by a schema in MySQL is typically meant a system schema. Simply put, a schema in MySQL corresponds to the concept of a database in SQL Server.

CREATE TABLE new-database-name.new-table-name

LIKE old-database.old-table-name;

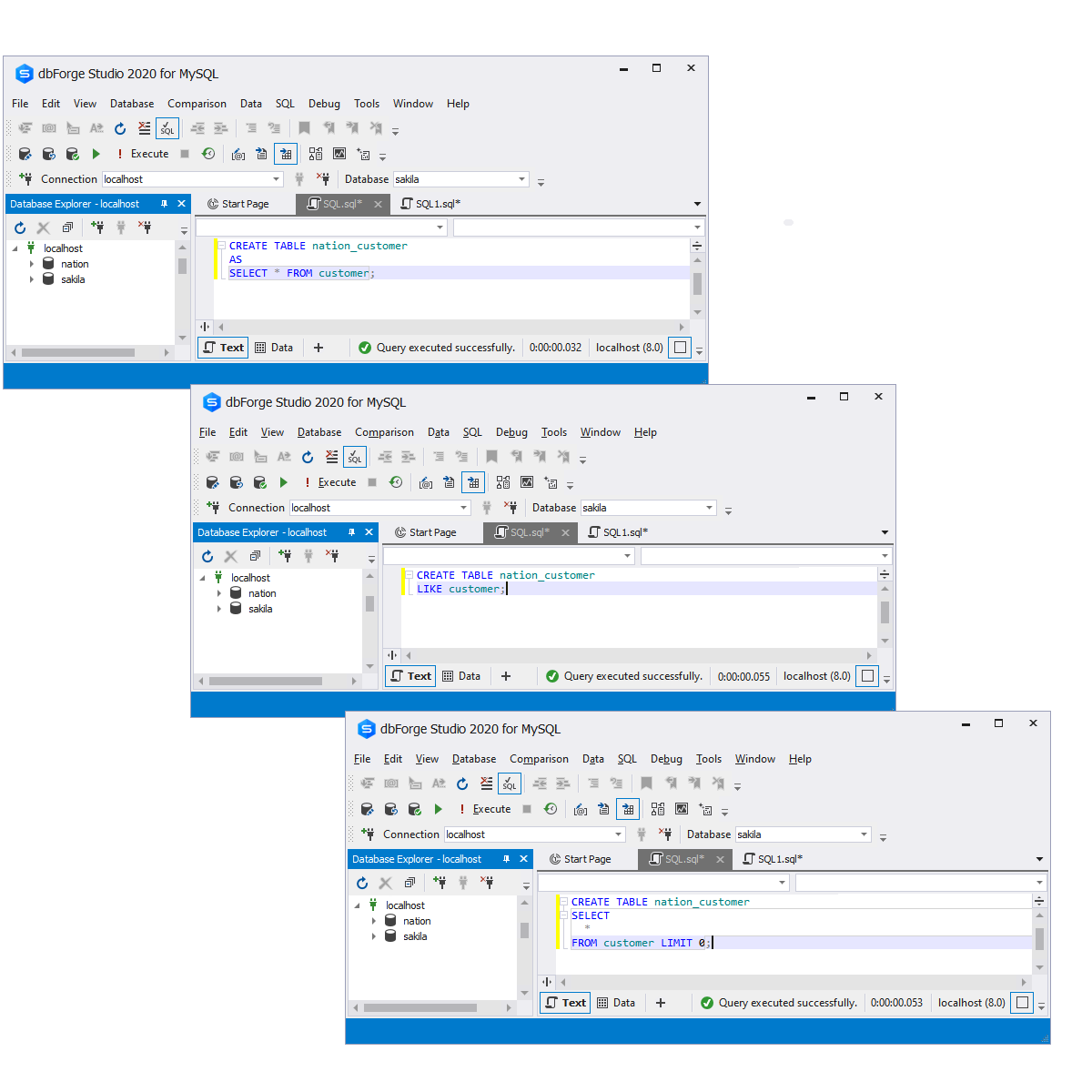
However, by running the query above, you will create a clone of the source table with all the column attributes, indexes, and keys. To copy the table structure only, you can run use a LIMIT clause in the query to make MySQL left aside the data.

CREATE TABLE new-database-name.new-table-name

SELECT

\*

FROM from old-database.old-table-name LIMIT 0;



Copy a table to a new table: Example

In MySQL, you don't have to copy the entire table, you can copy the specific columns only. For this, you can use the CREATE TABLE and the SELECT statement as shown below:

CREATE TABLE new\_table\_name

SELECT column1, column2, column3

FROM

old\_table\_table;

Suppose, we want to create a new table film\_copy which will consist of three columns: film\_id, film\_title, and film\_description.

CREATE TABLE film\_copy

SELECT

f.film\_id,

f.title,

f.description

FROM film f;