

# How To Import and Export Databases in MySQL or MariaDB

Posted December 21, 2016 46.4k [MYSQL](#) [MARIADB](#) [BACKUPS](#) [CENTOS](#) [DEBIAN](#) [UBUNTU](#) [UBUNTU 16.04](#)

## Introduction

Being able to import and export your database is an important skill to have. You can use data dumps for backup and restoration purposes, so you can recover older copies of your database in case of an emergency, or you can use them to migrate data to a new server or development environment.

Working with database dumps in MySQL and MariaDB is straightforward. This tutorial will cover how to export the database as well as import it from a dump file in MySQL and MariaDB.

## Prerequisites

To import and/or export a MySQL or MariaDB database, you will need:

- Access to the Linux server running MySQL or MariaDB
- The database name and user credentials for it

## Exporting the Database

The `mysqldump` console utility is used to export databases to SQL text files. These files can easily be transferred and moved around. You will need the database name itself as well as the username and password to an account with privileges allowing at least full read only access to the database.

Export your database using the following command.

```
mysqldump -u username -p database_name > data-dump.sql
```

- `username` is the username you can log in to the database with
- `database_name` is the name of the database that will be exported
- `data-dump.sql` is the file in the current directory that the output will be saved to

The command will produce no visual output, but you can inspect the contents of `filename.sql` to check if it's a legitimate SQL dump file by using:

```
head -n 5 data-dump.sql
```

The top of the file should look similar to this, mentioning that it's a MySQL dump for a database named `database_name`.

```
SQL dump fragment
-- MySQL dump 10.13  Distrib 5.7.16, for Linux (x86_64)
--
-- Host: localhost    Database: database_name
-- -----
-- Server version    5.7.16-0ubuntu0.16.04.1
```

If any errors happen during the export process, `mysqldump` will print them clearly to the screen instead.

## Importing the Database

To import an existing dump file into MySQL or MariaDB, you will have to create the new database. This is where the contents of the dump file will be imported.

First, log in to the database as **root** or another user with sufficient privileges to create new databases.

- `mysql -u root -p`

This will bring you into the MySQL shell prompt. Next, create a new database called `new_database`.

- `CREATE DATABASE new_database;`

You'll see this output confirming it was created.

```
Output
Query OK, 1 row affected (0.00 sec)
```

Now exit the MySQL shell by pressing `CTRL+D`. On the normal command line, you can import the dump file with the following command:

- `mysql -u username -p new_database < data-dump.sql`

- `username` is the username you can log in to the database with
- `newdatabase` is the name of the freshly created database
- `data-dump.sql` is the data dump file to be imported, located in the current directory

The successfully-run command will produce no output. If any errors occur during the process, `mysql` will print them to the terminal instead. You can check that the database was imported by logging in to the MySQL shell again and inspecting the data. This can be done by selecting the new database with `USE new_database` and then using `SHOW TABLES;` or a similar command to look at some of the data.

## Conclusion

You now know how to create database dumps from MySQL databases as well as how to import them again. `mysqldump` has multiple additional settings that may be used to alter how the dumps are created, which you can learn more about from the [official mysqldump documentation page](#).