How To Create and Manage Databases in MySQL and MariaDB on a Cloud Server

Authored by: ***ASPHostServer Administrator*** *[asphostserver@gmail.com]*
Saved From: *http://faq.asphosthelpdesk.com/article.php?id=186*

What are MySQL and MariaDB?

MySQL and MariaDB are relational database management systems. These tools can be used on your server to manage the data from many different programs. Both implement forms of the SQL querying language, and either can be used on a cloud server.

This guide will cover how to create a database using these tools. This is a fundamental skill needed to manage your data in an SQL environment. We will also cover several other aspects of database management.

For the purposes of this guide, we will be using an Ubuntu 12.04 server on a small server instance. However, everything should translate directly to other distributions.

How to Create a Database in MySQL and MariaDB

To begin, sign into MySQL or MariaDB with the following command:

```
mysql -u root -p
```

Enter the administrator password you set up during installation. You will be given a MySQL/MariaDB prompt.

We can now create a database by typing the following command:

```
CREATE DATABASE new_database;

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

To avoid errors in the event that the database name we've chosen already exists, use the following command:

```
CREATE DATABASE IF NOT EXISTS new_database;

Query OK, 1 row affected, 1 warning (0.01 sec)
```

The warning indicates that the database already existed and no new database was created.

If we leave the "IF NOT EXISTS" option off, and the database already exists, we will receive the following error:

```
ERROR 1007 (HY000): Can't create database 'other_database'; database exists
```

How to View Databases in MySQL and MariaDB

To view a list of the current databases that you have created, use the following command:

```
SHOW DATABASES;

+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| new_database       |
| other_database     |
| performance_schema |
+--------------------+
5 rows in set (0.00 sec)
```

The "information_schema", "performance_schema", and "mysql" databases are set up by default in most cases and should be left alone unless you know what you are doing.

How to Change Databases in MySQL and MariaDB

Any operations performed without explicitly specifying a database will be performed on the currently selected database.

Find out which database is currently selected with the following command:

```
SELECT database();

+------------+
| database() |
+------------+
| NULL       |
+------------+
1 row in set (0.01 sec)
```

We have received a result of "null". This means that no database is currently selected.

To select a database to use for subsequent operations, use the following command:

```
USE new_database;

Database changed
```

We can see that the database has been selected by re-issuing the command we ran previously:

```
SELECT database();

+--------------+
| database()   |
```

```
+--------------+
| new_database |
+--------------+
1 row in set (0.00 sec)
```

How to Delete a Database in MySQL and MariaDB

To delete a database in MySQL or MariaDB, use the following command:

```
DROP DATABASE new_database;

Query OK, 0 rows affected (0.00 sec)
```

This operation cannot be reversed! Make certain you wish to delete before pressing enter!

If this command is executed on a database that does not exist, the following error message will be given:

```
DROP DATABASE new_database;

ERROR 1008 (HY000): Can't drop database 'new_database'; database doesn't exist
```

To prevent this error, and ensure that the command executes successfully regardless of if the database exists, call it with the following syntax:

```
DROP DATABASE IF EXISTS new_database;

Query OK, 0 rows affected, 1 warning (0.00 sec)
```

The warning indicates that the database did not exist, but the command executes successfully anyways.

Conclusion

You now have the basic skills necessary to manage databases using MySQL and MariaDB. There are many things to learn, but you now have a good starting point to manage your databases.