

MariaDB: Getting started

How to start using MariaDB

Welcome on Stackhero's documentation!

Stackhero provides MariaDB instances that are ready for production in just 2 minutes!

Including TLS encryption (aka HTTPS), customizable domain name, PhpMyAdmin, backups and updates in just a click.

Try our [managed MariaDB cloud](#) in just 2 minutes

- [How to connect from PHP to MariaDB](#)
 - [Connect PHP to MariaDB with object-oriented style](#)
 - [Connect PHP to MariaDB with procedural style](#)
 - [Connect PHP to MariaDB with using credentials from environment variables](#)
- [Connect from Laravel to MariaDB](#)
- [How to connect from Node.js to MariaDB](#)
- [Connect from Node.js/NestJS/TypeORM to MariaDB](#)
- [Create a user on MariaDB using PhpMyAdmin](#)
- [Differences between MariaDB and MySQL](#)

How to connect from PHP to MariaDB

In these examples, we will use the "MySQL Improved Extension" known as `MySQLi`.

Connect PHP to MariaDB with object-oriented style

```
<?php

$hostname = '<STACKHERO_MARIADB_HOST>';
$user = 'root';
$password = '<STACKHERO_MARIADB_ROOT_PASSWORD>';
$database = 'mysql'; // You shouldn't use the "mysql" database. This is just for
the example. The recommended way is to create a dedicated database (and user) in
PhpMyAdmin and use it then here.

$mysqli = mysqli_init();
$mysqliConnected = $mysqli->real_connect($hostname, $user, $password, $database,
NULL, NULL, MYSQLI_CLIENT_SSL);
if (!$mysqliConnected) {
    die("Connect Error: " . $mysqli->connect_error());
}
```

```
}  
  
echo 'Success... ' . $mysqli->host_info . "\n";  
  
$mysqli->close();  
  
?>
```

Connect PHP to MariaDB with procedural style

```
<?php  
  
$hostname = '<STACKHERO_MARIADB_HOST>';  
$user = 'root';  
$password = '<STACKHERO_MARIADB_ROOT_PASSWORD>';  
$database = 'mysql'; // You shouldn't use the "mysql" database. This is just for  
the example. The recommended way is to create a dedicated database (and user) in  
PhpMyAdmin and use it then here.  
  
$mysqli = mysqli_init();  
$mysqliConnected = mysqli_real_connect($mysqli, $hostname, $user, $password,  
$database, NULL, NULL, MYSQLI_CLIENT_SSL);  
if (!$mysqliConnected) {  
    die("Connect Error: " . mysqli_connect_error($mysqli));  
}  
  
echo 'Success... ' . mysqli_get_host_info($mysqli) . "\n";  
  
mysqli_close($mysqli);  
  
?>
```

Connect PHP to MariaDB with using credentials from environment variables

We recommend to not keep your credentials in your source code but use environment variables in place.

Here is how to retrieve these credentials:

```
$hostname = parse_url(getenv('STACKHERO_MARIADB_HOST'));  
$user = parse_url(getenv('STACKHERO_MARIADB_USER'));  
$password = parse_url(getenv('STACKHERO_MARIADB_PASSWORD'));  
$database = parse_url(getenv('STACKHERO_MARIADB_USER'));
```

Connect from Laravel to MariaDB

Edit the file `config/database.php` and replace the `mysql` part with the following configuration:

```
'mysql' => [
  'driver' => 'mysql',
  'host' => env('STACKHERO_MARIADB_HOST'),
  'port' => 3306,
  'username' => env('STACKHERO_MARIADB_USER'),
  'password' => env('STACKHERO_MARIADB_PASSWORD'),
  'database' => env('STACKHERO_MARIADB_USER'),
  'charset' => 'utf8mb4',
  'collation' => 'utf8mb4_unicode_ci',
  'prefix' => '',
  'prefix_indexes' => true,
  'strict' => true,
  'engine' => null,
  'sslmode' => 'require',
  'options' => extension_loaded('pdo_mysql') ? array_filter([
    PDO::MYSQL_ATTR_SSL_CA => env('MYSQL_ATTR_SSL_CA'),
  ]) : [],
],
```

How to connect from Node.js to MariaDB

In this example we will use `mysql2` package with promises support.

To install it simply run this command: `npm install --save mysql2`

```
const mysql = require('mysql2/promise');

(async () => {
  const db = await mysql.createConnection({
    host: '<STACKHERO_MARIADB_HOST>',
    user: 'root',
    password: '<STACKHERO_MARIADB_ROOT_PASSWORD>'
  });

  // Create database stackherotest if not exists yet
  await db.query('CREATE DATABASE IF NOT EXISTS stackherotest');

  // Create table users if not exists yet
  await db.query('CREATE TABLE IF NOT EXISTS `stackherotest`.`users` '
    + '('
    + '`userId` INT UNSIGNED NOT NULL,'
    + '`name` VARCHAR(128) NOT NULL,'
    + '`address` TEXT NOT NULL,'
    + '`email` VARCHAR(265) NOT NULL'
```

```

+ ' '
+ 'ENGINE = InnoDB;');

// Insert a fake user
await db.query(
  'INSERT INTO `stackherotest`.`users` (`userId`, `name`, `address`, `email`)
VALUES ?',
  [
    [
      Math.round(Math.random() * 100000), // Generate a fake userId
      'User name', // column 'name'
      'User address', // column 'address'
      'user@email.com' // column 'email'
    ]
  ]
);

// Count number of rows in table users
const [ usersCount ] = await db.query('SELECT COUNT(*) AS `cpt` FROM
`stackherotest`.`users`');
console.log(`There is now ${usersCount[0].cpt} entries in table "users"`);

// Close the connection to MariaDB
await db.end();

}()).catch(error => {
  console.error('');
  console.error('🚨 An error occurred!');
  console.error(error);
  process.exit(1);
});

```

Connect from Node.js/NestJS/TypeORM to MariaDB

To connect from Node.js/NestJS/TypeORM, you have to add the `ssl` option like in this example:

```

TypeOrmModule.forRoot({
  type: 'mysql',
  host: '<STACKHERO_MARIADB_HOST>',
  port: 3306,
  username: 'root',
  password: '<STACKHERO_MARIADB_ROOT_PASSWORD>',
  database: 'root',
  entities: [],
  synchronize: true,

```

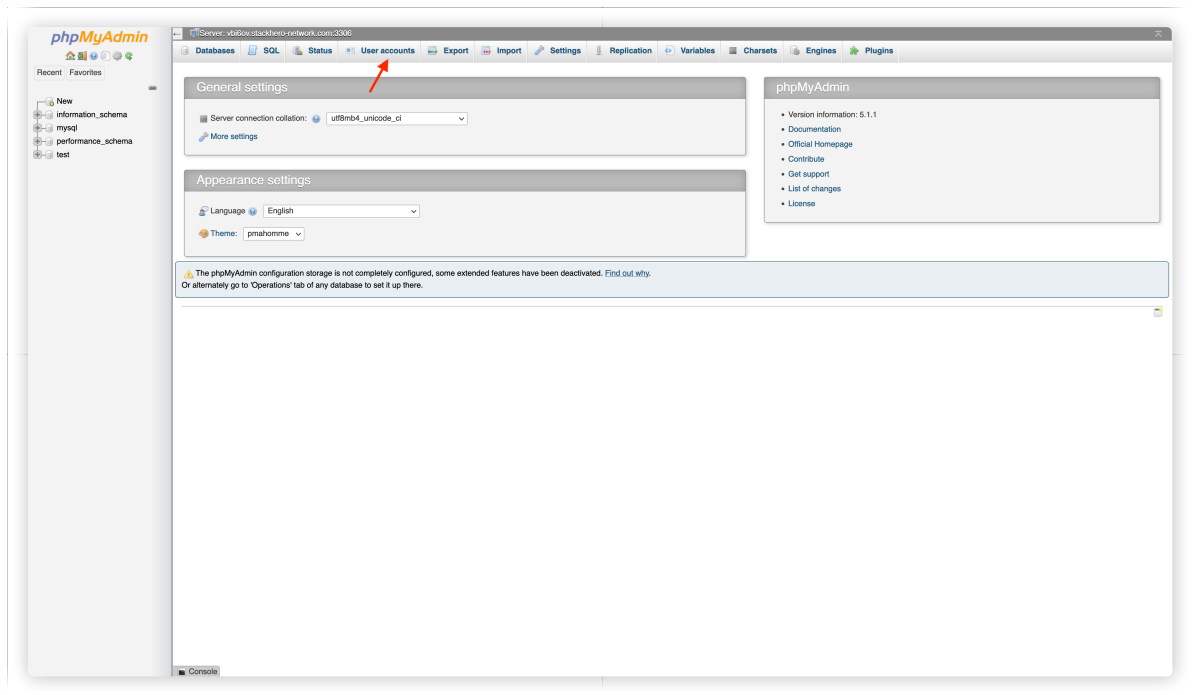
```
ssl: {}  
})
```

Create a user on MariaDB using PhpMyAdmin

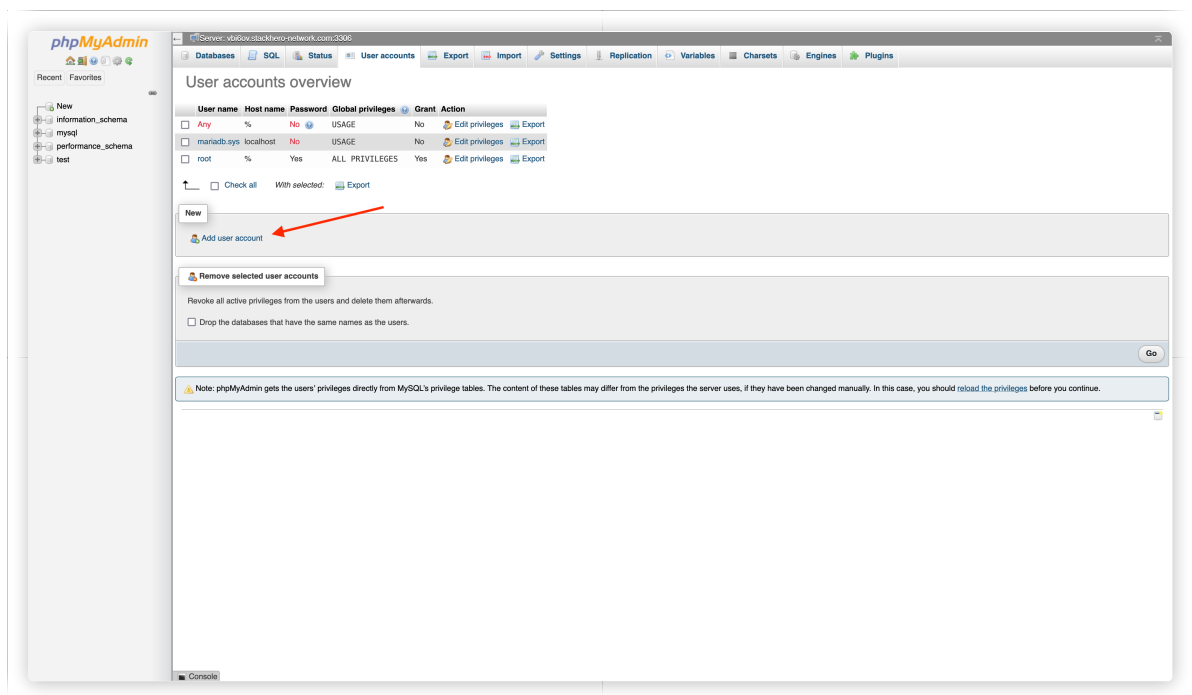
A best practice is to create a user other than the default "root" for your application.

The easiest way to do that is to use PhpMyAdmin.

1. In PhpMyAdmin, click on **User accounts** on top.

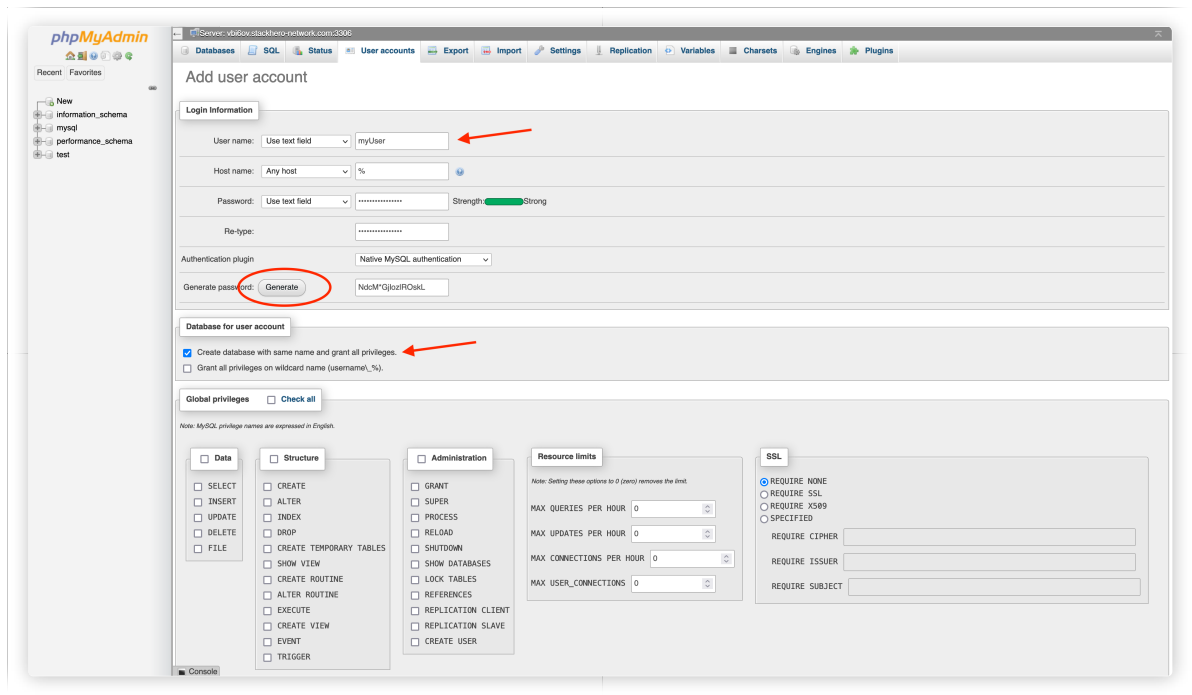


2. Click on **Add user account**.



3. Fill the user creation form:

- Define an account name (generally your application name)
- Click on "Generate password" to get a secured password (copy it to your clipboard)
- Check the "Create database with same name and grant all privileges"



Once validated, the user will be created as its database which will get the same name as the username.

Differences between MariaDB and MySQL


MySQL has been bought by Oracle in 2010. The open source community decided to create a fork of it, called MariaDB, to keep it totally independent from big companies.


For general purposes, MariaDB and MySQL are quite the same.


Other articles about MariaDB that might interest you

- [Advanced usage](#)
How to configure, optimize, export or import your MariaDB data
- [Troubleshooting](#)
MariaDB common issues and solutions

Our Managed Services


 Directus


 Docker

 Elasticsearch


[Terms of Service](#)

 GitLab


 Grafana

 Graylog


[Privacy Policy](#)


 InfluxDB


 Kibana

 MariaDB


[Documentations](#)

 Matomo


 Mattermost

 Memcached


[Support](#)


 Mercure-Hub


 MinIO


 Mosquitto

[Status](#)


 MySQL

 Nextcloud


 Node-RED

 [English](#)

 [Global](#)


 Node.js


 PHP

 Postfix




Directus, Docker, Elasticsearch, GitLab, Grafana, Graylog, InfluxDB, Kibana, MariaDB, Matomo, Mattermost, Memcached, Mercure-Hub, MinIO, MongoDB, Mosquitto, MySQL, Nextcloud, Node-RED, Node.js, PHP, Postfix, PostgreSQL, Prometheus, Python, RabbitMQ, Redis[®]*, RethinkDB are trademarks and property of their respective owners. All product and service names used on this website are for identification purposes of their open sourced products only and do not imply endorsement. Stackhero is not affiliated to these trademarks or companies.


 PostgreSQL


 Prometheus

 Python

*Redis is a registered trademark of Redis Ltd. Any rights therein are reserved to Redis Ltd. Any use by Stackhero is for referential purposes only and does not indicate any sponsorship, endorsement or affiliation between Redis and Stackhero

 RabbitMQ

 Redis[®]*

 RethinkDB

Some icons of this website are made by Dmitry Miroliubov.

© Stackhero. All rights reserved.