

AI-03005 mySQL Information

Owner	© Rigel Arcayan
Tags	Project
Created time	@September 16, 2025 8:04 PM

What is mySQL? What is the history of the database creation and development? Create a comprehensive guide in the installation, deployment, management. List all the commands used in the Ubuntu server environment. Create a guide on how to perform basic database creation, user creation, passwords, logins, reset account, drop db etc. create a guide on how to install and configure phasing , it's prerequisites and needed dependencies.

A **MySQL server** is an open-source relational database management system (RDBMS) that uses Structured Query Language (SQL) to store, manage, and retrieve data. It's a key component of the popular LAMP (Linux, Apache, MySQL, PHP/Perl/Python) stack used for web development.

History of MySQL

MySQL was created in **1995** by the Swedish company MySQL AB, founded by Michael Widenius, David Axmark, and Allan Larsson. The name is a combination of "My," the name of Widenius's daughter, and "SQL." The initial goal was to provide an efficient and reliable data management system for both home and professional users.

- **Open Source Model:** In **2000**, MySQL AB released the software under the terms of the GNU General Public License (GPL), making it open source. This move allowed third-party developers to contribute to its development, leading to its rapid growth and widespread adoption.
- **Acquisitions:** The company was acquired by **Sun Microsystems** in **2008**, which in turn was acquired by **Oracle Corporation** in **2009**. Oracle has continued to develop and support MySQL, releasing new versions and adding features.

- **Forks:** Due to concerns about Oracle's ownership and the future of the open-source project, some developers forked the code to create new, independent versions. The most notable of these is **MariaDB**, created by Michael Widenius himself.

Installation, Deployment, and Management on Ubuntu

This guide covers the basic steps for installing, deploying, and managing MySQL on an Ubuntu server.

1. Installation

The easiest way to install MySQL on Ubuntu is by using the `apt` package manager.

- **Update the package list:**Bash

```
sudo apt update
```

- **Install the MySQL server package:**Bash

```
sudo apt install mysql-server
```

- During the installation, you'll be prompted to set a **root password** for the MySQL server. Make sure to use a strong password.

2. Initial Configuration

After installation, it's a good practice to run the security script provided by MySQL to secure the installation.

- **Run the security script:**Bash

```
sudo mysql_secure_installation
```

This script will guide you through several steps, including:

- **Password Validation Policy:** You can choose a password validation level (LOW, MEDIUM, or HIGH) to enforce strong passwords for new users.
- **Change the root password:** If you didn't set one during installation, you can do it here.

- **Remove anonymous users:** This prevents unauthorized access to the database.
- **Disallow remote root login:** A crucial security measure to prevent remote attackers from gaining control of your database.
- **Remove the test database:** This removes a default database that's often used by attackers.

3. Managing the MySQL Service

You can manage the MySQL service using `systemctl`.

- **Start the service:**Bash

```
sudo systemctl start mysql
```

- **Stop the service:**Bash

```
sudo systemctl stop mysql
```

- **Restart the service:**Bash

```
sudo systemctl restart mysql
```

- **Check the status of the service:**Bash

```
sudo systemctl status mysql
```

- **Enable MySQL to start on boot:**Bash

```
sudo systemctl enable mysql
```

Basic MySQL Commands

To interact with the MySQL server, you'll need to use the MySQL shell.

1. Login to MySQL

- **Login as the root user:**Bash

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

Enter the root password you set during installation when prompted.

2. Database Commands

Once you're in the MySQL shell (`mysql>`), you can run SQL commands.

- **Show all databases:**SQL

```
SHOW DATABASES;
```

- **Create a new database:**SQL

```
CREATE DATABASE database_name;
```

- **Select a database to work with:**SQL

```
USE database_name;
```

- **Drop (delete) a database:**SQL

```
DROP DATABASE database_name;
```

3. User Commands

- **Create a new user:**SQL

```
CREATE USER 'username'@'localhost' IDENTIFIED BY 'password';
```

Replace `username` and `password` with your desired credentials. The `'@localhost'` part specifies that the user can only connect from the local machine. To allow connections from anywhere, use `'username'@'%'`.

- **Grant privileges to a user:**SQL

```
GRANT ALL PRIVILEGES ON database_name.* TO 'username'@'localhost';
```

This command gives the user all permissions on the specified database. You can replace `ALL PRIVILEGES` with specific ones like `SELECT`, `INSERT`, `UPDATE`, `DELETE`, etc.

- **Flush privileges to apply changes:**SQL

```
FLUSH PRIVILEGES;
```

- **Show user privileges:SQL**

```
SHOW GRANTS FOR 'username'@'localhost';
```

- **Revoke privileges from a user:SQL**

```
REVOKE ALL PRIVILEGES ON database_name.* FROM 'username'@'localhost';
```

- **Drop (delete) a user:SQL**

```
DROP USER 'username'@'localhost';
```

- **Change a user's password:SQL**

```
ALTER USER 'username'@'localhost' IDENTIFIED BY 'new_password';
```

Installing and Configuring phpMyAdmin

phpMyAdmin is a popular web-based tool for managing MySQL databases. It provides a graphical user interface (GUI) that simplifies many common database tasks.

Prerequisites

Before installing phpMyAdmin, you need a functioning **web server** (like Apache) and **PHP** installed with necessary extensions.

- **Install PHP and required extensions:Bash**

```
sudo apt install php libapache2-mod-php php-mysql php-json php-gd php-mbstring php-zip
```

- **Enable the PHP module for Apache:Bash**

```
sudo a2enmod php
```

- **Restart Apache to apply changes:Bash**

```
sudo systemctl restart apache2
```

Step-by-Step Installation

1. Install phpMyAdmin:Bash

```
sudo apt install phpmyadmin
```

2. During the installation, a blue configuration screen will appear.

- **Select your web server:** Use the spacebar to select `apache2` and press Enter.
- **Configure the database for phpMyAdmin:** Select **Yes** to configure the database using `dbconfig-common` .
- **Set a password for the phpmyadmin user:** The installer will create a new user for phpMyAdmin. Enter a strong password for this user.

3. Create a link to the phpMyAdmin installation directory:Bash

```
sudo ln -s /usr/share/phpmyadmin /var/www/html/phpmyadmin
```

4. Access phpMyAdmin in your browser:

Open your web browser and navigate to `http://your_server_ip_address/phpmyadmin` or `http://localhost/phpmyadmin` if you're on the local machine.

5. Log in:

Use the root user and the MySQL root password you set earlier. You can also log in with any other MySQL user you've created.