MySQL Day 1

Day 1: Installation & Basics

Author : Ajit Yadav (MSSQL , MYSQL , PostgreSQL DBA)

Day 1 Objectives

By the end of today, you should be able to:
✓ Install MySQL Server
✓ Start/Stop MySQL service
✓ Login using CLI
✓ Check version
✓ Secure the server
✓ Understand config file location

Step 1: Install MySQL (Linux)

For Ubuntu / Debian

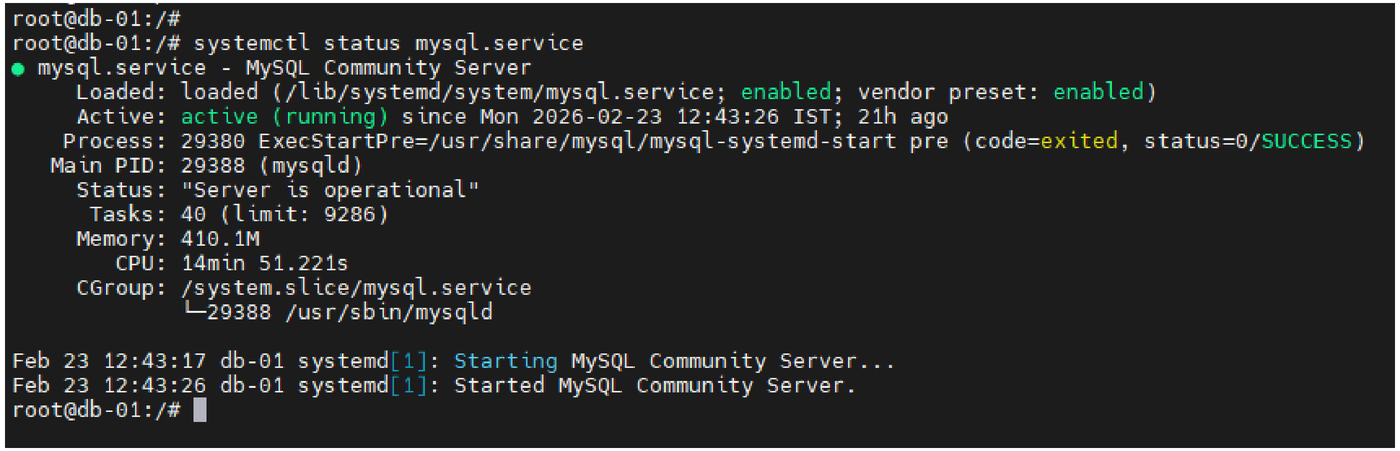
```
root@db-01:/# sudo apt update
root@db-01:/# sudo apt install mysql-server -y
```

For RHEL / CentOS / Rocky

```
root@db-01:/# sudo yum install mysql-server -y
```

After install :

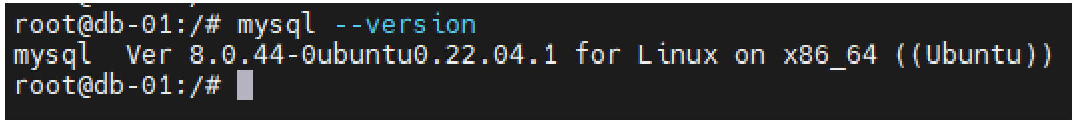
```
root@db-01:/# sudo systemctl start mysqld
root@db-01:/# sudo systemctl enable mysqld

root@db-01:/# sudo systemctl status mysqld
```



```
root@db-01:/#
root@db-01:/# systemctl status mysql.service
● mysql.service - MySQL Community Server
     Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2026-02-23 12:43:26 IST; 21h ago
    Process: 29380 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
   Main PID: 29388 (mysqld)
     Status: "Server is operational"
      Tasks: 40 (limit: 9286)
     Memory: 410.1M
        CPU: 14min 51.221s
     CGroup: /system.slice/mysql.service
             └─29388 /usr/sbin/mysqld

Feb 23 12:43:17 db-01 systemd[1]: Starting MySQL Community Server...
Feb 23 12:43:26 db-01 systemd[1]: Started MySQL Community Server.
root@db-01:/# 
```

Step 2: Check MySQL Version

mysql --version



```
root@db-01:/# mysql --version
mysql  Ver 8.0.44-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
root@db-01:/#
```

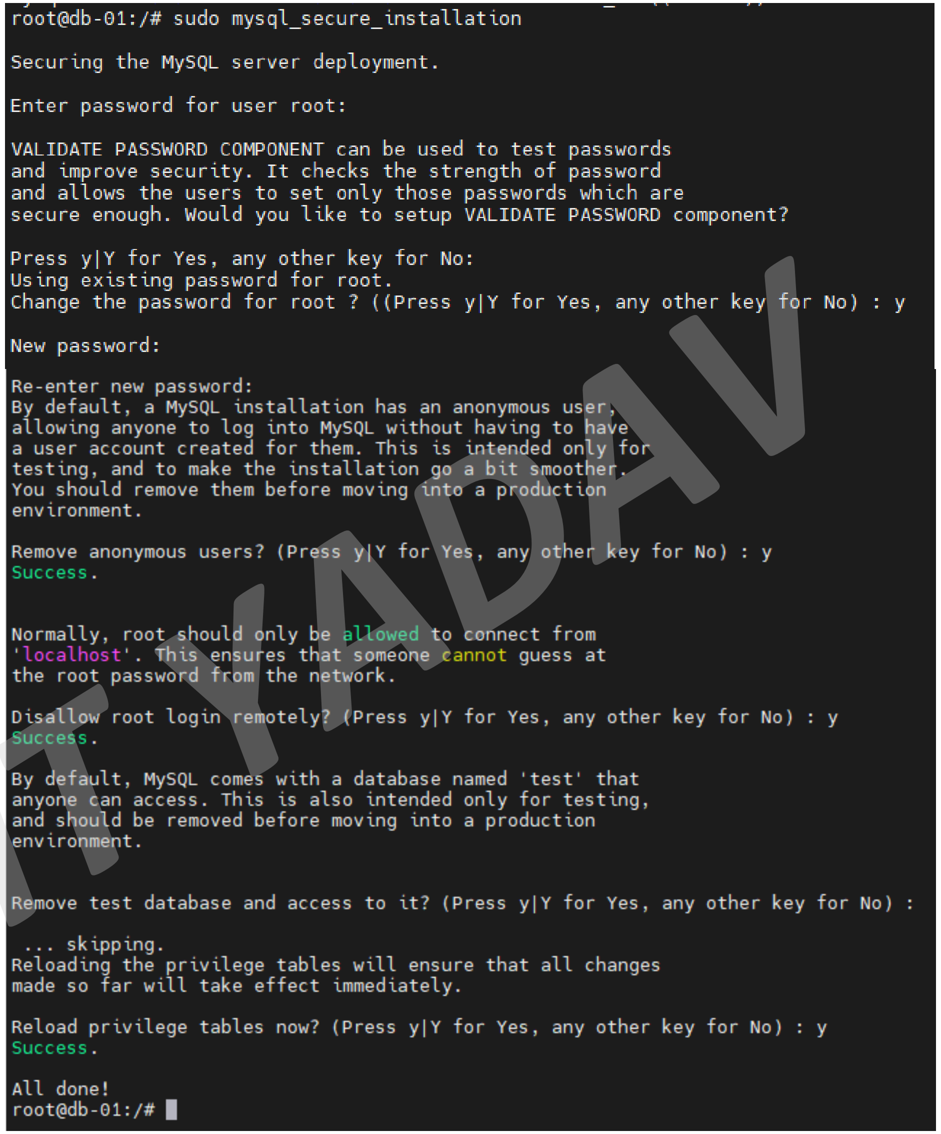
Step 3: Secure Installation

sudo mysql_secure_installation

Follow steps:

- ✓ Set root password
- ✓ Remove anonymous users → YES
- ✓ Disable remote root login → YES
- ✓ Remove test DB → YES
- ✓ Reload privileges → YES

This is mandatory for production.



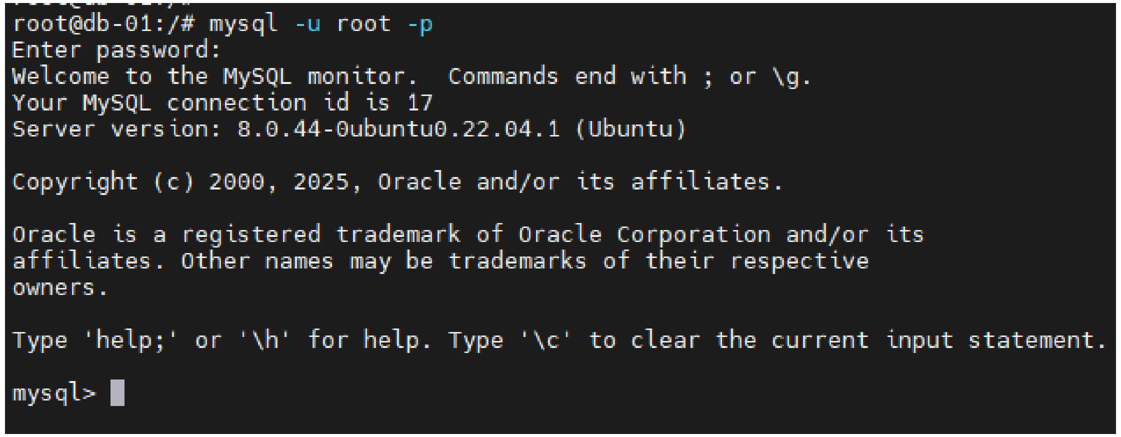
```
root@db-01:/# sudo mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root:

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No:
Using existing password for root.
Change the password for root ? ((Press y|Y for Yes, any other key for No) : y

New password:

Re-enter new password:
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.


Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.


Remove test database and access to it? (Press y|Y for Yes, any other key for No) :

 ... skipping.
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
root@db-01:/#
```

Step 4: Login to MySQL (CLI)

mysql -u root -p
Enter password :

and u will see the mysql cli interface

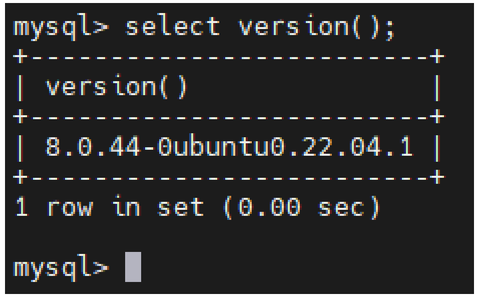
.



```
root@db-01:/# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 8.0.44-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

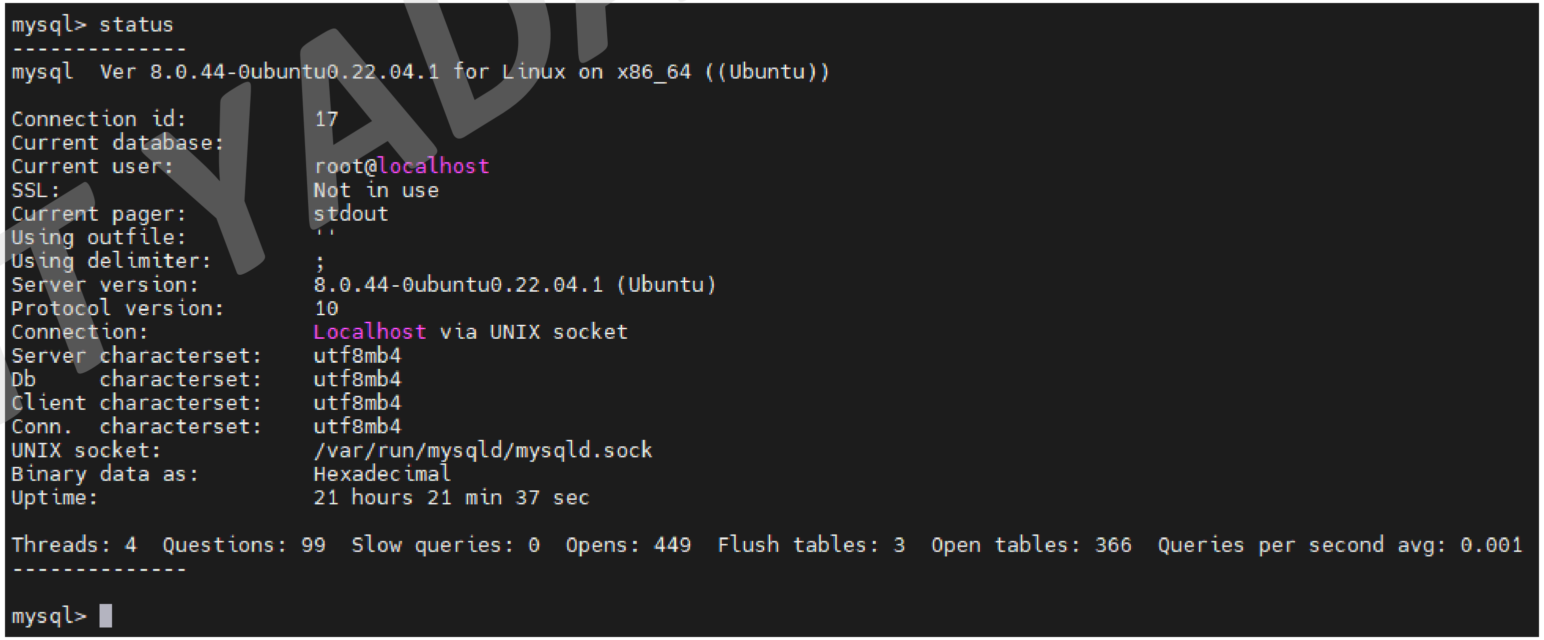
Step 5: Basic MySQL Commands

Check server info:



```
mysql> select version();
+-------------------------+
| version()               |
+-------------------------+
| 8.0.44-0ubuntu0.22.04.1 |
+-------------------------+
1 row in set (0.00 sec)

mysql>
```

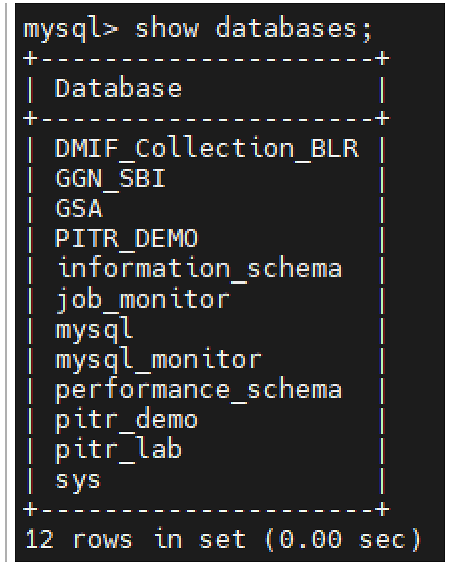
status   # to see the mysql database and server details



```
mysql> status
--------------
mysql  Ver 8.0.44-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))

Connection id:          17
Current database:
Current user:           root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server version:         8.0.44-0ubuntu0.22.04.1 (Ubuntu)
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    utf8mb4
Db     characterset:    utf8mb4
Client characterset:    utf8mb4
Conn.  characterset:    utf8mb4
UNIX socket:            /var/run/mysqld/mysqld.sock
Binary data as:         Hexadecimal
Uptime:                 21 hours 21 min 37 sec

Threads: 4  Questions: 99  Slow queries: 0  Opens: 449  Flush tables: 3  Open tables: 366  Queries per second avg: 0.001
--------------

mysql>
```

List of databases :

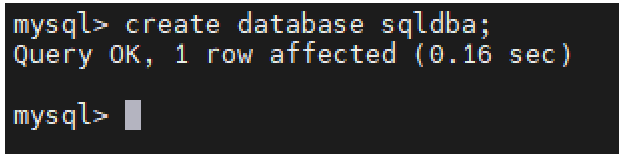
show databases



```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| DMIF_Collection_BLR |
| GGN_SBI            |
| GSA                |
| PITR_DEMO          |
| information_schema |
| job_monitor        |
| mysql              |
| mysql_monitor      |
| performance_schema |
| pitr_demo          |
| pitr_lab           |
| sys                |
+--------------------+
12 rows in set (0.00 sec)
```

Create test database :

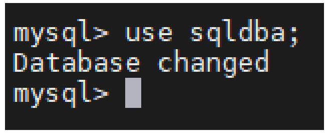
create database sqldba;



```
mysql> create database sqldba;
Query OK, 1 row affected (0.16 sec)

mysql> 
```

connect to that database:

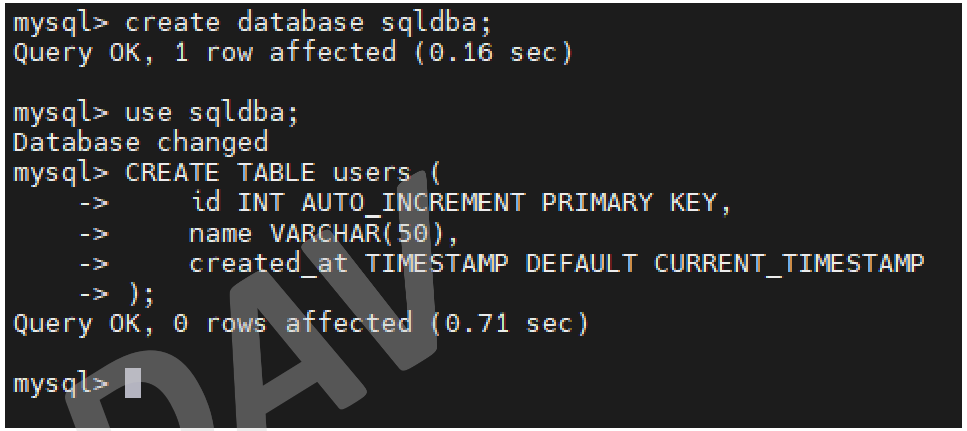
use sqldba;



```
mysql> use sqldba;
Database changed
mysql> 
```

create table inside that db :

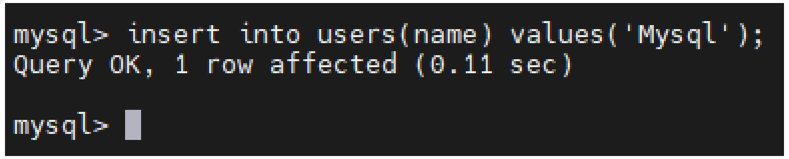
```
CREATE TABLE users (
   id INT AUTO_INCREMENT PRIMARY KEY,
   name VARCHAR(50),
   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```



```
mysql> create database sqldba;
Query OK, 1 row affected (0.16 sec)

mysql> use sqldba;
Database changed
mysql> CREATE TABLE users (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     name VARCHAR(50),
    ->     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    -> );
Query OK, 0 rows affected (0.71 sec)

mysql> 
```

insert record :

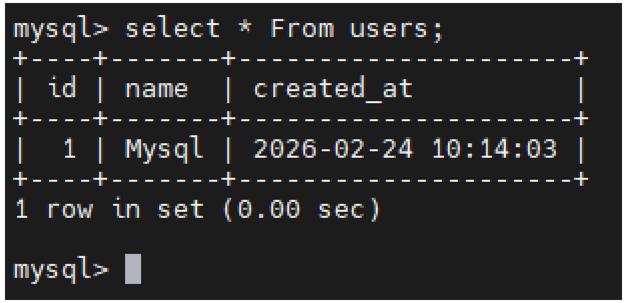
INSERT INTO users(name) VALUES ('Mysql');



```
mysql> insert into users(name) values('Mysql');
Query OK, 1 row affected (0.11 sec)

mysql> 
```

view data :

SELECT* FROM users;



```
mysql> select * From users;
+----+-------+---------------------+
| id | name  | created_at          |
+----+-------+---------------------+
|  1 | Mysql | 2026-02-24 10:14:03 |
+----+-------+---------------------+
1 row in set (0.00 sec)

mysql> 
```

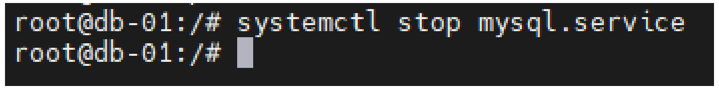
Step 6: MySQL Service Management

start :

sudo systemctl start mysqld

stop :

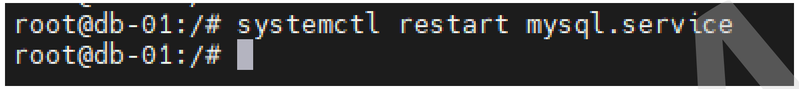
sudo systemctl stop mysqld



```
root@db-01:/# systemctl stop mysql.service
root@db-01:/# 
```

restart :

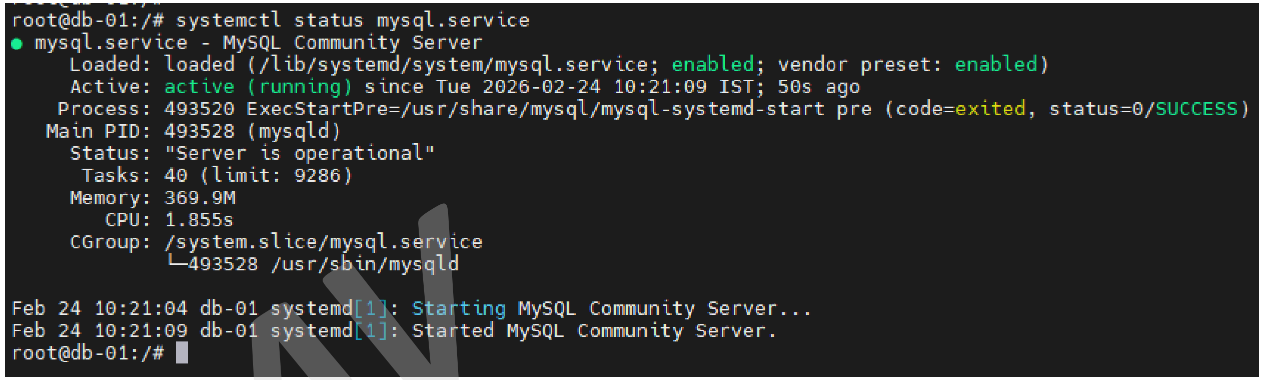
sudo systemctl restar mysqld



```
root@db-01:/# systemctl restart mysql.service
root@db-01:/# 
```

status :

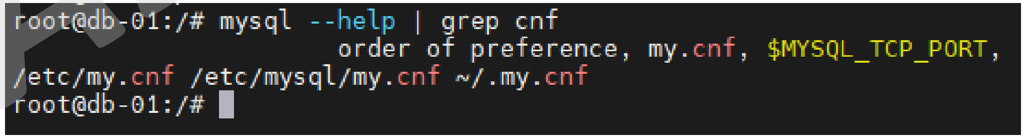
systemctl status mysqld



```
root@db-01:/# systemctl status mysql.service
● mysql.service - MySQL Community Server
     Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2026-02-24 10:21:09 IST; 50s ago
    Process: 493520 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
   Main PID: 493528 (mysqld)
     Status: "Server is operational"
      Tasks: 40 (limit: 9286)
     Memory: 369.9M
        CPU: 1.855s
     CGroup: /system.slice/mysql.service
             └─493528 /usr/sbin/mysqld

Feb 24 10:21:04 db-01 systemd[1]: Starting MySQL Community Server...
Feb 24 10:21:09 db-01 systemd[1]: Started MySQL Community Server.
root@db-01:/# 
```

Step 7: Configuration File Location

mysql --help | grep cnf



```
root@db-01:/# mysql --help | grep cnf
                        order of preference, my.cnf, $MYSQL_TCP_PORT,
/etc/my.cnf /etc/mysql/my.cnf ~/.my.cnf
root@db-01:/# 
```

📁 Ubuntu:

/etc/mysql/my.cnf

/etc/mysql/mysql.conf.d/mysqld.cnf

📁 RHEL:

/etc/my.cnf