Step #1: Add MySQL APT Repository

Download the MySQL repository by executing the following command

Update the system Packages:

```
sudo apt update
```

We need to Install wget using below command:

```
sudo apt install wget -y
```

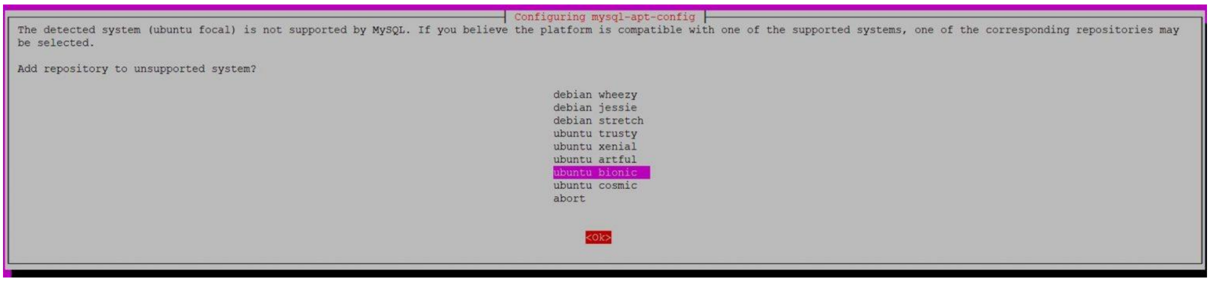
Download the repository using the below command:

```
wget https://dev.mysql.com/get/mysql-apt-config_0.8.12-1_all.deb
```

Once downloaded install repository using below command:

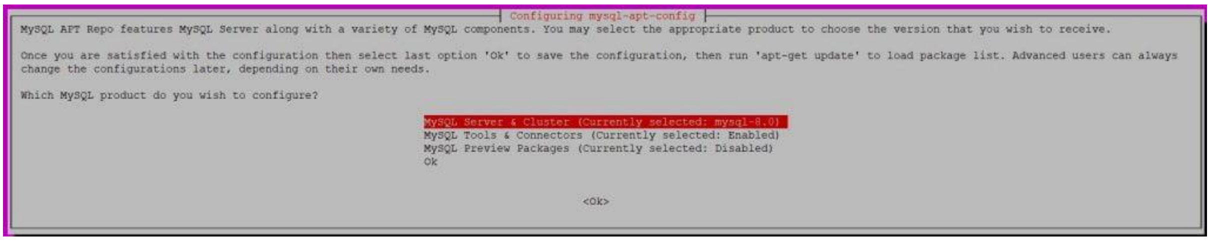
```
sudo dpkg -i mysql-apt-config_0.8.12-1_all.deb
```

In the prompt, choose Ubuntu Bionic and click Ok



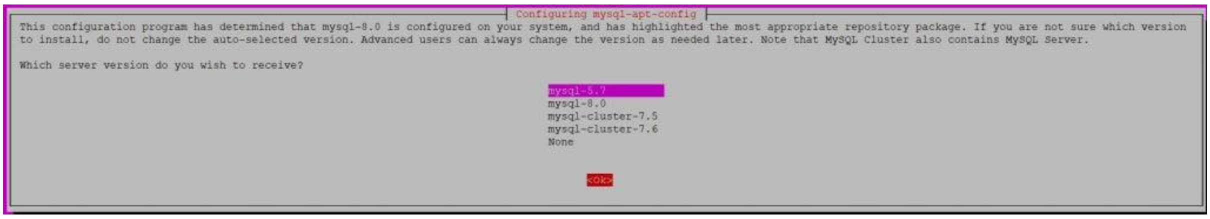
```
Configuring mysql-apt-config
The detected system (ubuntu focal) is not supported by MySQL. If you believe the platform is compatible with one of the supported systems, one of the corresponding repositories may
be selected.

Add repository to unsupported system?

                                   debian wheezy
                                   debian jessie
                                   debian stretch
                                   ubuntu trusty
                                   ubuntu xenial
                                   ubuntu artful
                                   ubuntu bionic
                                   ubuntu cosmic
                                   abort

                                      <Ok>
```

The next prompt shows MySQL 8.0 chosen by default. Choose the first option and click OK



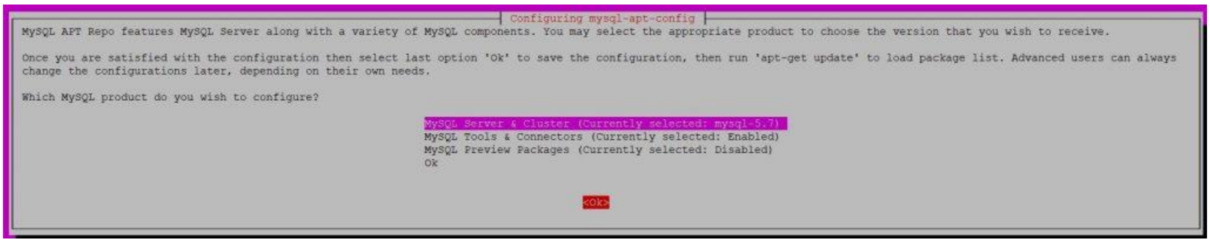
```
Configuring mysql-apt-config
MySQL APT Repo features MySQL Server along with a variety of MySQL components. You may select the appropriate product to choose the version that you wish to receive.

Once you are satisfied with the configuration then select last option 'Ok' to save the configuration, then run 'apt-get update' to load package list. Advanced users can always
change the configurations later, depending on their own needs.

Which MySQL product do you wish to configure?

                         MySQL Server & Cluster (Currently selected: mysql-8.0)
                         MySQL Tools & Connectors (Currently selected: Enabled)
                         MySQL Preview Packages (Currently selected: Disabled)
                         Ok

                                      <Ok>
```

In the next prompt, select MySQL 5.7 server and click OK.



```
Configuring mysql-apt-config
This configuration program has determined that mysql-8.0 is configured on your system, and has highlighted the most appropriate repository package. If you are not sure which version
to install, do not change the auto-selected version. Advanced users can always change the version as needed later. Note that MySQL Cluster also contains MySQL Server.

Which server version do you wish to receive?

                                        mysql-5.7
                                        mysql-8.0
                                        mysql-cluster-7.5
                                        mysql-cluster-7.6
                                        None

                                           <Ok>
```

The next prompt selects MySQL5.7 by default. Choose the last option Ok and click OK



```
Configuring mysql-apt-config
MySQL APT Repo features MySQL Server along with a variety of MySQL components. You may select the appropriate product to choose the version that you wish to receive.

Once you are satisfied with the configuration then select last option 'Ok' to save the configuration, then run 'apt-get update' to load package list. Advanced users can always
change the configurations later, depending on their own needs.

Which MySQL product do you wish to configure?

                                MySQL Server & Cluster (Currently selected: mysql-5.7)
                                MySQL Tools & Connectors (Currently selected: Enabled)
                                MySQL Preview Packages (Currently selected: Disabled)
                                Ok

                                           <Ok>
```

Step #2: Update MySQL Repository:

Update your system package:

```
sudo apt-get update
```

Let’s search for MySQL 5,7 using below command:

```
sudo apt-cache policy mysql-server
```

Step #3: Install MySQL 5.7 on Ubuntu 22.04 LTS

Before Installing MySQL 5.7 you need to run this command:

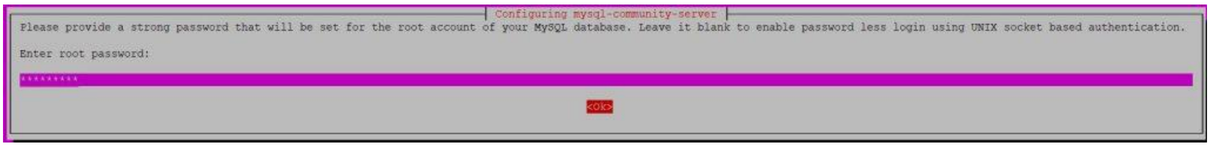
```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys
467B942D3A79BD29


sudo apt update
```

we are going to install MySQL 5.7 client, MySQL 5.7 server with the below command:

```
sudo apt install -f mysql-client=5.7* mysql-community-server=5.7*
mysql-server=5.7*
```

Hit the y key to start installation of MySQL 5.7 on Ubuntu 22.04 LTS

Enter and re-enter root password:



```
Configuring mysql-community-server
Please provide a strong password that will be set for the root account of your MySQL database. Leave it blank to enable password less login using UNIX socket based authentication.

Enter root password:

*********

<Ok>
```

Step #4: Secure MySQL 5.7 Installation on Ubuntu 22.04LTS:

Run the following commands

```
sudo mysql_secure_installation
```

Provide the root password set above and you will have output like this:

```
ubuntu@ip-172-31-44-141:~$ sudo mysql_secure_installation



Securing the MySQL server deployment.



Enter password for user root:



VALIDATE PASSWORD PLUGIN 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 plugin?
```

```
Press y|Y for Yes, any other key for No: y


There are three levels of password validation policy:


LOW    Length >= 8

MEDIUM Length >= 8, numeric, mixed case, and special characters

STRONG Length >= 8, numeric, mixed case, special characters and dictionary                file


Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 1

Using existing password for root.


Estimated strength of the password: 25

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


 ... skipping.

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
```

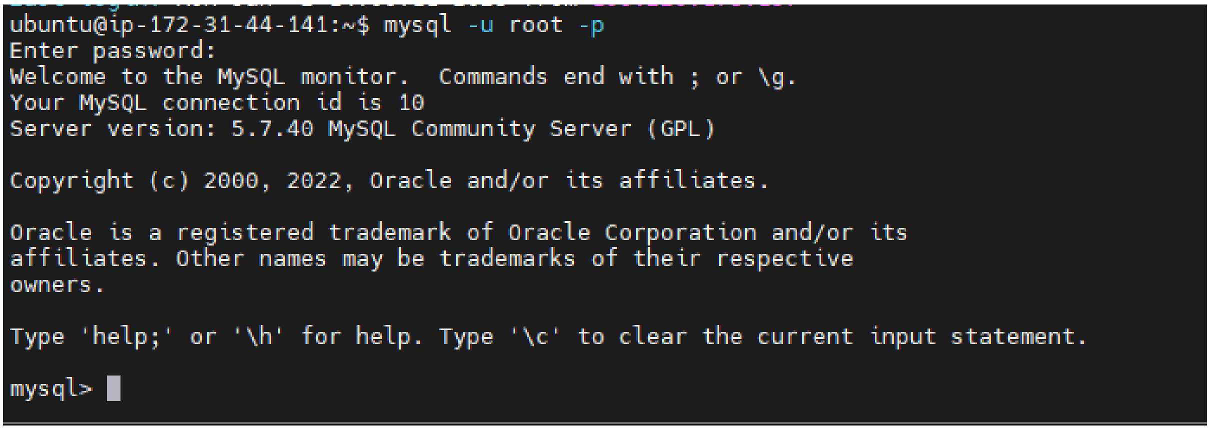
```
environment.


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) : y
 - Dropping test database...
```

```
Success.


 - Removing privileges on test database...

Success.


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!
```

Step #5: Check MySQL 5.7 version

To confirm the installed version,Login to MySQL 5.7 Server using the below command.

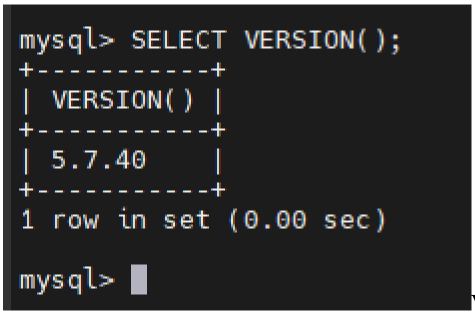
```
mysql -u root -p
```



```
ubuntu@ip-172-31-44-141:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.7.40 MySQL Community Server (GPL)

Copyright (c) 2000, 2022, 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> 
```

mysql 5.7

Check the version using below command:



```
mysql> SELECT VERSION();
+-----------+
| VERSION() |
+-----------+
| 5.7.40    |
+-----------+
1 row in set (0.00 sec)

mysql>
```

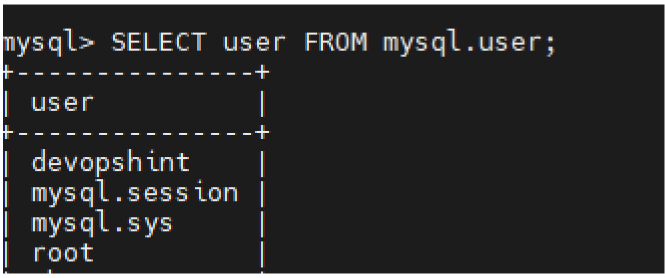
version

Here we have covered Install MySQL 5.7 on Ubuntu 22.04 LTS.

Step #6: Create MySQL User

Let’s create user using below command

```
CREATE USER 'devopshint'@'%' IDENTIFIED BY 'Devops@123';
```



```
mysql> SELECT user FROM mysql.user;
+---------------+
| user          |
+---------------+
| devopshint    |
| mysql.session |
| mysql.sys     |
| root          |
```

users

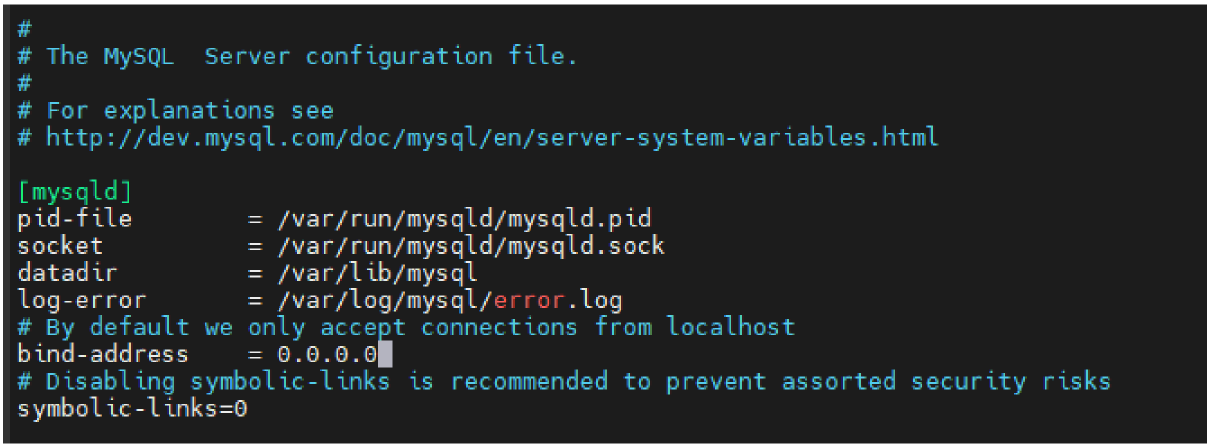
Step #7: Enable MySQL remote access

By default, In MySQL database server remote access is disabled for security reason.

To enable remote connections of MySQL Server, we have to change bind-address in MySQL configuration file.

Open the /etc/mysql/mysql.conf.d/mysqld.cnf file

```
sudo nano /etc/mysql/mysql.conf.d/mysqld.cnf
```

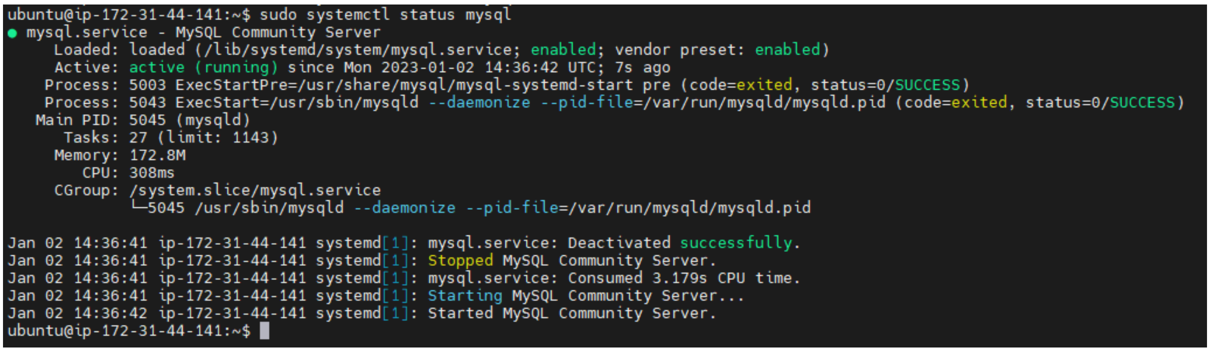


```
#
# The MySQL  Server configuration file.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

[mysqld]
pid-file        = /var/run/mysqld/mysqld.pid
socket          = /var/run/mysqld/mysqld.sock
datadir         = /var/lib/mysql
log-error       = /var/log/mysql/error.log
# By default we only accept connections from localhost
bind-address    = 0.0.0.0
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0
```

bindip

Save the file and restart mysql

```
sudo systemctl restart mysql
```



```
ubuntu@ip-172-31-44-141:~$ sudo systemctl status mysql
● mysql.service - MySQL Community Server
     Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2023-01-02 14:36:42 UTC; 7s ago
    Process: 5003 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
    Process: 5043 ExecStart=/usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid (code=exited, status=0/SUCCESS)
   Main PID: 5045 (mysqld)
      Tasks: 27 (limit: 1143)
     Memory: 172.8M
        CPU: 308ms
     CGroup: /system.slice/mysql.service
             └─5045 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid

Jan 02 14:36:41 ip-172-31-44-141 systemd[1]: mysql.service: Deactivated successfully.
Jan 02 14:36:41 ip-172-31-44-141 systemd[1]: Stopped MySQL Community Server.
Jan 02 14:36:41 ip-172-31-44-141 systemd[1]: mysql.service: Consumed 3.179s CPU time.
Jan 02 14:36:41 ip-172-31-44-141 systemd[1]: Starting MySQL Community Server...
Jan 02 14:36:42 ip-172-31-44-141 systemd[1]: Started MySQL Community Server.
ubuntu@ip-172-31-44-141:~$ 
```