# Install MySql on your Ubuntu server

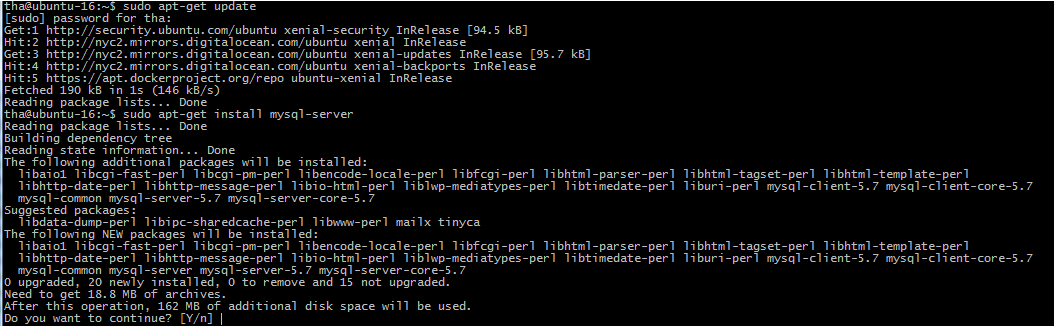
This guide is based on these web resources:

<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-16-04> and

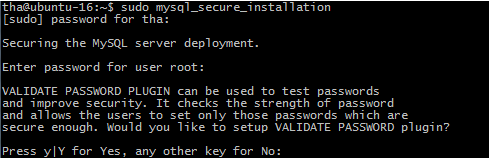
<https://www.digitalocean.com/community/tutorials/a-basic-mysql-tutorial> and

<https://www.digitalocean.com/community/tutorials/how-to-create-a-new-user>

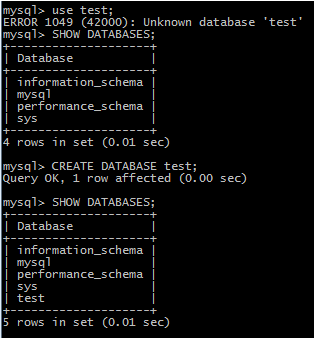
## Install MySql

1. In the command line: log into your server with a user account that is not root but has sudo privileges (like you did in the tutorial: getting started with Digital Ocean). Then
2. sudo apt-get update
3. sudo apt-get install mysql-server
4. Write Y for yes
5. Set the password for the root user

(remember this password for later)

1. sudo mysql\_secure\_installation
2. write no to validate password plugin:
3. For the rest of the questions write yes
4. All done!

## Get into MySql console

1. mysql -u root -p
2. you now have the mysql> prompt
3. try writing CREATE DATABASE test;
4. now try: SHOW DATABASES;
5. 

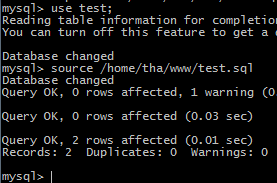
## Create a new user and grant privileges

1. CREATE USER 'newuser'@'localhost' IDENTIFIED BY 'password'; (replace the writing in red with your own username and password.
2. GRANT ALL PRIVILEGES ON \* . \* TO 'newuser'@'localhost'; (This will give the new user read/write privileges on all databases.
3. That´s it: write: exit to get back to ubuntu prompt.

## Run script file on the MySql Server

1. Create a script file (E.g. one that can add tables to the test database) and test it locally
2. Use scp to upload the file to Ubuntu (cd into the local folder holding the script/sql file first)



1. Log into Ubuntu and run the mysql prompt: mysql -u root –p

use test;

source <path to sql file>

1. Check to see if script has run by running a select statement. E.g : SELECT \* FROM customers;

## How to connect to your database using MySql Workbench

By default MySQL does not allow remote connections.

To change this you will need to modify the configuration file. This file is to be found in one of two places

“/etc/mysql/my.cnf” (if this file doesn’t exist or has no content – instead go to)

“/etc/mysql/mysql.conf.d/mysqld.cnf”

Find section: [mysqld] and the line that says “bind-address”

bind-address = localhost

Comment out that line entirely (use the hash-tag). - Then it will listen on all IPs and ports which you need because you will be connecting remotely to it over public IPv4.

After that add a user to access your database like this:

mysql -u root –p

CREATE USER 'testuser'@'localhost' IDENTIFIED BY 'password123';

CREATE USER 'testuser'@'%' IDENTIFIED BY 'password123';

GRANT ALL ON test.\* TO 'testuser'@'localhost';

GRANT ALL ON test.\* TO 'testuser'@'%';

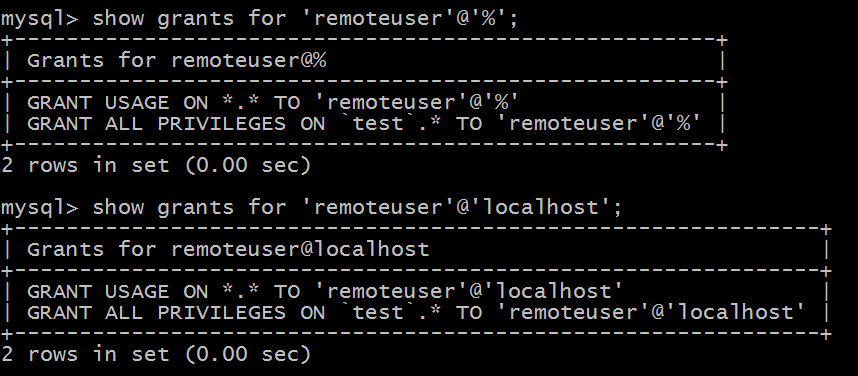
Replace % with your local IP address of your laptop/desktop or if it is dynamic (like it would be if you are sitting on the school ip) you can add them either by: '192.168.0.%' as a dynamic C-class or just keep the '%' if you want to be able to connect from anywhere (this is less secure).

Also replace the writing in red with your own username and password.

You can check the privileges like this

show grants for 'testuser’@’%';

show grants for 'testuser’@’localhost';

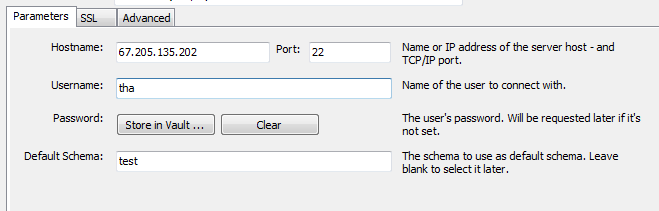


Now restart MySQL:

exit (To get back to the Ubuntu prompt)

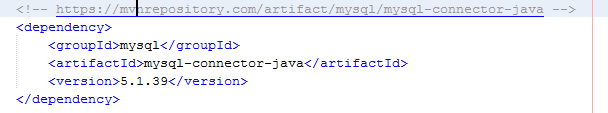
service mysql restart

Now on your computer open up MySql Workbench and “Connect to database”



Put your credentials in here and log in.

## How to connect to your database from your java web application

1. Create a table in you mysql database and put in some entries (data)
2. Create a Maven Web project in Netbeans
3. In the Pom.xml add this depency: ([Get it here](https://mvnrepository.com/artifact/mysql/mysql-connector-java/5.1.39))

Machine generated alternative text: public class DB {
public static final String driver = “com.mysql.jdbc.Driver”;
public static final String uri = “jdbc:mysql://localhost/test”;
public static final String username = “admin”;
public static final String fazf’Cm = “passwzzz’;
public static Connection getConnection() {
Connection conn = null;
try{
Class. iforName (driver);
conn = DriverManager. getConnection(DB. uri, DB. username, DB.password);

1. Create a Servlet and read from the test database and print the content to the response output stream.
2. Check the result in a browser.