Introduction to MySQL

## What is MySQL?

* MySQL is a relational database management system
* MySQL is open-source
* MySQL is free
* MySQL is ideal for both small and large applications
* MySQL is very fast, reliable, scalable, and easy to use
* MySQL is cross-platform
* MySQL is compliant with the ANSI SQL standard
* MySQL was first released in 1995
* MySQL is developed, distributed, and supported by Oracle Corporation
* MySQL is named after co-founder Monty Widenius's daughter: My

## Who Uses MySQL?

* Huge websites like Facebook, Twitter, Airbnb, Booking.com, Uber, GitHub, YouTube, etc.
* Content Management Systems like WordPress, Drupal, Joomla!, Contao, etc.
* A very large number of web developers around the world

## Show Data On Your Web Site

To build a web site that shows data from a database, you will need:

* An RDBMS database program (like MySQL)
* A server-side scripting language, like PHP
* To use SQL to get the data you want
* To use HTML / CSS to style the page

## Installing MySQL on Windows

The default installation on any version of Windows is now much easier than it used to be, as MySQL now comes neatly packaged with an installer. Simply download the installer package, unzip it anywhere and run the setup.exe file.

The default installer setup.exe will walk you through the trivial process and by default will install everything under C:\mysql.

Test the server by firing it up from the command prompt the first time. Go to the location of the **mysqld server** which is probably C:\mysql\bin, and type −

mysqld.exe --console

## Installing MySQL on Windows

The default installation on any version of Windows is now much easier than it used to be, as MySQL now comes neatly packaged with an installer. Simply download the installer package, unzip it anywhere and run the setup.exe file.

The default installer setup.exe will walk you through the trivial process and by default will install everything under C:\mysql.

Test the server by firing it up from the command prompt the first time. Go to the location of the **mysqld server** which is probably C:\mysql\bin, and type −

mysqld.exe --console

**NOTE** − If you are on NT, then you will have to use mysqld-nt.exe instead of mysqld.exe

If all went well, you will see some messages about startup and **InnoDB**. If not, you may have a permissions issue. Make sure that the directory that holds your data is accessible to whatever user (probably MySQL) the database processes run under.

MySQL will not add itself to the start menu, and there is no particularly nice GUI way to stop the server either. Therefore, if you tend to start the server by double clicking the mysqld executable, you should remember to halt the process by hand by using mysqladmin, Task List, Task Manager, or other Windows-specific means.

## Verifying MySQL Installation

After MySQL, has been successfully installed, the base tables have been initialized and the server has been started: you can verify that everything is working as it should be via some simple tests.

### Use the mysqladmin Utility to Obtain Server Status

Use **mysqladmin** binary to check the server version. This binary would be available in /usr/bin on linux and in C:\mysql\bin on windows.

[root@host]# mysqladmin --version

It will produce the following result on Linux. It may vary depending on your installation −

mysqladmin Ver 8.23 Distrib 5.0.9-0, for redhat-linux-gnu on i386

If you do not get such a message, then there may be some problem in your installation and you would need some help to fix it.

### Execute simple SQL commands using the MySQL Client

You can connect to your MySQL server through the MySQL client and by using the **mysql** command. At this moment, you do not need to give any password as by default it will be set as blank.

You can just use following command −

[root@host]# mysql

It should be rewarded with a mysql> prompt. Now, you are connected to the MySQL server and you can execute all the SQL commands at the mysql> prompt as follows −

mysql> SHOW DATABASES;

+----------+

| Database |

+----------+

| mysql |

| test |

+----------+

2 rows in set (0.13 sec)

## Setting Up a MySQL User Account

For adding a new user to MySQL, you just need to add a new entry to the **user** table in the database **mysql**.

The following program is an example of adding a new user **guest** with SELECT, INSERT and UPDATE privileges with the password **guest123;** the SQL query is −

root@host# mysql -u root -p

Enter password:\*\*\*\*\*\*\*

mysql> use mysql;

Database changed

mysql> INSERT INTO user

(host, user, password,

select\_priv, insert\_priv, update\_priv)

VALUES ('localhost', 'guest',

PASSWORD('guest123'), 'Y', 'Y', 'Y');

Query OK, 1 row affected (0.20 sec)

mysql> FLUSH PRIVILEGES;

Query OK, 1 row affected (0.01 sec)

mysql> SELECT host, user, password FROM user WHERE user = 'guest';

+-----------+---------+------------------+

| host | user | password |

+-----------+---------+------------------+

| localhost | guest | 6f8c114b58f2ce9e |

+-----------+---------+------------------+

1 row in set (0.00 sec)

When adding a new user, remember to encrypt the new password using PASSWORD() function provided by MySQL. As you can see in the above example, the password mypass is encrypted to 6f8c114b58f2ce9e.

Notice the FLUSH PRIVILEGES statement. This tells the server to reload the grant tables. If you don't use it, then you won't be able to connect to MySQL using the new user account at least until the server is rebooted.

You can also specify other privileges to a new user by setting the values of following columns in user table to 'Y' when executing the INSERT query or you can update them later using UPDATE query.

* Select\_priv
* Insert\_priv
* Update\_priv
* Delete\_priv
* Create\_priv
* Drop\_priv
* Reload\_priv
* Shutdown\_priv
* Process\_priv
* File\_priv
* Grant\_priv
* References\_priv
* Index\_priv
* Alter\_priv