D B M S

Introduction to RDBMS - MySQL

MySQL (MySQL RDBMS)

MySQL is (as of July 2013) the world's most widely used open-source **relational database management system (RDBMS)**, enabling the cost-effective delivery of reliable, high-performance and scalable Web-based and embedded database applications. It is widely-used as the database component of LAMP (Linux, Apache, MySQL, Perl/PHP/Python) web application software stack.

MySQL was developed by Michael Widenius and David Axmark in 1994. Presently MySQL is maintained by Oracle (formerly Sun, formerly MySQL AB).

What is MySQL

- MySQL is a database management system.
- MySQL databases are relational.
- MySQL software is Open Source.
- The MySQL Database Server is fast, reliable, scalable, and easy to use.
- MySQL Server works in client/server or embedded systems.
- Initial release : 23 May 1995
- Current stable release : 5.6.13 / 30 July 2013
- Written in : C, C++
- Operating system : Cross-platform
- Available in : English
- License of MySQL is available under GNU General Public License (version 2) or proprietary EULA.

Who uses MySQL

- Some of the most visited websites like Flickr, Facebook, Wikipedia, Google (not for search), YouTube.

- Content Management Systems like WordPress, phpBB, Drupal, Joomla, TYPO3, MODx.

- Last but not least, a large number of web developers across the world.

MySQL supported platforms

MySQL runs on

- Linux (RedHat, SUSE, Mandrake, Debian)

- Embedded Linux (MontaVista, LynuxWorks BlueCat)

- Unix (Solaris, HP-UX, AIX)

- BSD (Mac OS X, FreeBSD)

- Windows (Windows 2000, Windows NT)

- RTOS (QNX)

MySQL supported third party tools

Drivers:
- ODBC

- JDBC

- .NET

- C++

Languages:
- C

- C++

- C#

- Java

- Delphi

- Visual Basic

- Perl

- Python

- PHP

Development Tools:

- Microsoft Visual Studio

- Borland Delphi and JBuilder

- Eclipse

- NetBeans

Some of the widely used MySQL front ends (tools for managing MySQL)

The MySQL GUI Tools Bundle is a cross-platform open source suite of desktop applications, building and manipulating the data within MySQL databases Development on the GUI Tools bundle has stopped, The GUI Tools bundle has been replaced by MySQL Workbench with the beta releases of MySQL Workbench 5.2. Currently MySQL Workbench Team are working on Version 6.0. The first public beta, labeled version 6.0.2, was released on June 14, 2013. There are lot of third-party free and proprietary graphical administration applications available that integrate with MySQL and users to work with database. Here are some third-party tools for managing MySQL:

Tools	Description
phpMyAdmin	Third party, Free, Web based
HeidiSQL	Third party, Free, For Windows

Adminer	Third party, Free
DBEdit	Third party, Free
dbForge GUI Tools	Third party, Free
Navicat	Third party, Commercial
Maatkit	Third party, Command line, free
MySQL Sandbox	Third party, Command line, free
SQLBuddy	A free Web-based front end, developed in PHP.
SQLyog	Commercial, but a free 'community' edition is available.
Toad for MySQL	Third party, free from Quest Software

MySQL Data Types

In MySQL there are three main types : text, number, and Date/Time types.

Text types:

Data type	Description
CHAR(size)	Holds a fixed length string (can contain letters, numbers, and special characters). The fixed size is specified in parenthesis. Can store up to 255 characters
VARCHAR(size)	Holds a variable length string (can contain letters, numbers, and special characters). The maximum size is specified in parenthesis. Can store up to 255 characters. **Note:** If you put a greater value than 255 it will be converted to a TEXT type
TINYTEXT	Holds a string with a maximum length of 255 characters
TEXT	Holds a string with a maximum length of 65,535 characters
BLOB	For BLOBs (Binary Large OBjects). Holds up to 65,535 bytes of data

MEDIUMTEXT	Holds a string with a maximum length of 16,777,215 characters
MEDIUMBLOB	For BLOBs (Binary Large OBjects). Holds up to 16,777,215 bytes of data
LONGTEXT	Holds a string with a maximum length of 4,294,967,295 characters
LONGBLOB	For BLOBs (Binary Large OBjects). Holds up to 4,294,967,295 bytes of data
ENUM(x,y,z,etc.)	Let you enter a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted. **Note:** The values are sorted in the order you enter them. You enter the possible values in this format: ENUM('X','Y','Z')
SET	Similar to ENUM except that SET may contain up to 64 list items and can store more than one choice

Number types:

Data type	Description
TINYINT(size)	-128 to 127 normal. 0 to 255 UNSIGNED*. The maximum number of digits may be specified in parenthesis
SMALLINT(size)	-32768 to 32767 normal. 0 to 65535 UNSIGNED*. The maximum number of digits may be specified in parenthesis
MEDIUMINT(size)	-8388608 to 8388607 normal. 0 to 16777215 UNSIGNED*. The maximum number of digits may be specified in parenthesis
INT(size)	-2147483648 to 2147483647 normal. 0 to 4294967295 UNSIGNED*. The maximum number of digits may be specified in parenthesis
BIGINT(size)	-9223372036854775808 to 9223372036854775807 normal. 0 to 18446744073709551615 UNSIGNED*. The maximum number of digits may be specified in parenthesis
FLOAT(size,d)	A small number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter

DOUBLE(size,d)	A large number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter
DECIMAL(size,d)	A DOUBLE stored as a string , allowing for a fixed decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter

*The integer types have an extra option called UNSIGNED. Normally, the integer goes from an negative to positive value. Adding the UNSIGNED attribute will move that range up so it starts at zero instead of a negative number.

Date types:

Data type	Description
DATE()	A date. Format: YYYY-MM-DD **Note:** The supported range is from '1000-01-01' to '9999-12-31'
DATETIME()	*A date and time combination. Format: YYYY-MM-DD HH:MI:SS **Note:** The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'
TIMESTAMP()	*A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD HH:MI:SS **Note:** The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC
TIME()	A time. Format: HH:MI:SS **Note:** The supported range is from '-838:59:59' to '838:59:59'
YEAR()	A year in two-digit or four-digit format. **Note:** Values allowed in four-digit format: 1901 to 2155. Values allowed in two-digit format: 70 to 69, representing years from 1970 to 2069

Connecting to MySQL

Before you perform any operations, you need to connect to MySQL server and select a MySQL database.

Connecting to MySQL server from command line

Command :

```
mysql -h HostName -u UserName -p
```

Parameters :

Name	Description
-h	Keyword, followed by HOSTNAME.
HostName	MySQL server name
-u	Keyword, followed by USERNAME.
UserName	MySQL user name
-p	Asks you to enter password.

As soon as you enter this command, it asks you to provide password for the user you mentioned. Supplying the appropriate password would allow you to connect to MySQL server.

Connecting to MySQL database from command line

Command :

```
use DatabaseName;
```

Where DatabaseName is the database you want to select.

Disconnecting or closing MySQL server from command line

Command :

What is a database?

When an amount of data is stored in an organized way, which is called a database. In computers, a database is managed by a software called Database Management System.

What is a table?

A table is a set of data values. These values are organized using a vertical columns and horizontal rows. Columns are identified by their names.

Create a MySQL Table Using MySQLi

The CREATE TABLE statement is used to create a table in MySQL.

We will create a table named "MyGuests", with five columns: "id", "firstname", "lastname", "email" and "reg_date":

EXAMPLE 1

```
CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP
)
```

EXAMPLE 2

```
CREATE TABLE IF NOT EXISTS newauthor
(aut_id        varchar(8),
aut_name       varchar(50),
country        varchar(25),
home_city      varchar(25) NULL);
```

The SQL INSERT INTO Statement

The INSERT INTO statement is used to insert new records in a table.

SQL INSERT INTO Syntax

It is possible to write the INSERT INTO statement in two forms.

The first form does not specify the column names where the data will be inserted, only their values:

```
INSERT INTO table_name
VALUES (value1,value2,value3,...);
```

The second form specifies both the column names and the values to be inserted:

```
INSERT INTO table_name (column1,column2,column3,...)
VALUES (value1,value2,value3,...);
```

MySQL DESCRIBE statement

MySQL DESCRIBE statement is used to show the structure of the created table.

Syntax:

```
DESCRIBE [table_name];
```