

# INTRODUCTION TO MYSQL

Presented by Ying Wang  
Uppsala University

# Content

- \* *What is MySQL*
- \* *Why MySQL*
- \* *LAMP*
- \* *How to Learn*
- \* *Basic Database Management*

# What is MySQL

- \* MySQL is a database management system

**database** - structured collection of data

simple shopping list, student information or....

to add, access, and process data stored in a computer database

- \* MySQL is a relational database management system

stores data in separate tables NOT “putting all the data in one big storeroom”

speed & flexibility

- \* MySQL software is Open Source

possible for anyone to use and modify

download and use for free

change source code to suit your needs

- \* The MySQL Database Server is very fast, reliable, and

easy to use (Try it!)

# Why MySQL

- \* Most popular open source database
- \* Fast, High reliability and Easy to use
- \* Open Source
- \* Cross platforms runs on more than **20** platforms including Linux, Windows, MAC OS/X, FreeBSD, OpenSolaris, SunOS
- \* multi-threaded and multi-user
- \* popular for web applications (LAMP)

# LAMP

- \* LAMP= Linux+Apache+MySQL+PHP / Perl / Python
- \* Often tied with popular PHP, MySQL acts as the database component of the LAMP (BAMP, MAMP, SAMP, and WAMP platforms (Linux/BSD/Mac/(Open)Solaris/Windows-Apache-MySQL-PHP/Perl/Python)
- \* Several high-traffic web sites (incl. Flickr, Facebook, Wikipedia, Google, Nokia and YouTube) use MySQL for data storage and user log in.

# How to Learn

- \* Get a copy of Reference Manual from <http://dev.mysql.com/doc/#refman>
- \* Beginners Tutorials: Ask Google with “MySQL tutorial”

# Connect to Server

- \* `shell> mysql -h host -u user -p`
- \* Enter password: `*****`
- \* Disconnect from the Server
- \* `mysql> QUIT`

# Manage Databases

- \* CREATE DATABASE databaseName;
- \* SHOW DATABASES;
- \* USE databaseName; (Select db to work with)
- \* DROP DATABASE databaseName;

# Manage Tables

- \* CREATE TABLE tableName(name1 type1, name2 type2, ...);
- \* SHOW TABLES;
- \* DESCRIBE tableName; (see the table characteristics)

# Manage Tables

- \* ALTER TABLE tableName options[, options...] (alter existing table structures)
- \* DROP TABLE [IF EXISTS] tableName [, tableName, tableName...]
- \* TRUNCATE TABLE tableName

# Use Index

- \* MySQL first check whether indexes exists
- \* If exists it will use index to select exact physical location without scanning the whole table.
- \* helps to speed the retrieval of data from MySQL database server
- \* `CREATE INDEX indexName ON tableName (columnName [(length)] [ASC | DESC],...)`
- \* `DROP INDEX indexName ON tableName`

# Query

```
* SELECT columnName1, columnName2...  
FROM tableName  
[WHERE conditions]  
[GROUP BY group  
[HAVING group_conditions]]  
[ORDER BY sort_columns];
```

# QUERY - WHERE Clause

\* select a particular rows which match its conditions

```
SELECT firstname, lastname, email  
FROM ECommerceStudnets  
WHERE lastname = "Lena"
```

# QUERY - DISTINCT

- \* get rid of the duplicated results from SELECT statement
- \* `SELECT DISTINCT lastname FROM ECommerceStudnets;`

# QUERY - GROUP BY

to find number of students in different programs you can use GROUP BY clause

```
SELECT COUNT( * ) AS `Numbers` ,  
`Program`  
FROM `ECommerceStudents`  
GROUP BY `Program`  
ORDER BY `Program`
```

The ORDER BY clause: sort the result set in ascending or descending order

# QUERY - HAVING Clause

\* Usually work with GROUP BY clause

```
SELECT COUNT( * ) AS `Numbers` ,  
`Program`  
FROM `ECommerceStudnets`  
GROUP BY `Program`  
HAVING count( * ) > 3
```

# QUERY - IN

\* select values which match any one of a list of values

```
SELECT email  
FROM ECommerceStudnets  
WHERE program IN (  
'msc', 'dvp3'  
)
```

# QUERY - BETWEEN

\* retrieve values within a specific range

\* **SELECT** column\_list

**FROM** tableName

**WHERE** columnName **BETWEEN** lowerRange **AND**  
upperRange

# QUERY - LIKE

- \* pattern matching
- \* % :any string of zero or more characters
- \* \_ :single character.
- \* How Many ECommerceStudnets are using Gmail ?

```
SELECT count (*)  
FROM ECommerceStudnets  
WHERE email LIKE '@gmail.com'
```

# UNION

- \* combine two or more result sets

- \* `SELECT statement1`

```
UNION [DISTINCT | ALL]  
SELECT statement2
```

# USEFUL FUNCTIONS

\* SUM

\* AVG

\* MAX and MIN

\* COUNT

# INSERT DATA

insert one or more rows to the table

```
INSERT INTO
```

```
`student`.`ECommerceStudents` (
```

```
`PNumber` ,
```

```
`lastName` ,
```

```
`firstName` ,
```

```
`email`
```

```
)
```

```
VALUES (
```

```
'8001020001', 'Xiaoming', 'Zhu',
```

```
'xiaoming@gmail.com'
```

```
);
```

# UPDATE DATA

\* update existing data

\* UPDATE ECommerceStudnets

SET email = 'lenali@gmail.com'

WHERE email = 'lena.li@gmail.com'

# DELETE DATA

\* remove one data row all many rows

\* **DELETE FROM** tableName

[**WHERE** conditions]

# Reference

- \* <http://dev.mysql.com/doc/refman/6.0/en>
- \* <http://www.mysqltutorial.org/>

THANK YOU