CSCI 3308 Software Methods & Tools [Fall 2015]
Instructor: Boese
Lab 5

SQL with MySQL Server

Objectives

- Able to install MySQL server on a UNIX/Linux host
- Able to write SQL commands to affect/access the database
- Please pair-program today!

Note: if there is anything wrong with your computer, pair-program today with someone else!

PRE-LAB work:

1. Be sure to have a working copy of the current semester's VM installed.
2. Run the following command from the terminal to install mysql
   `sudo apt-get update && sudo apt-get install cu-cs-csci-3308`
   When prompted, type Y
   When prompted for a password, type something in.
   Make sure there are no errors displayed when you do these. If so, try importing a new VM or contact help@cs.colorado.edu

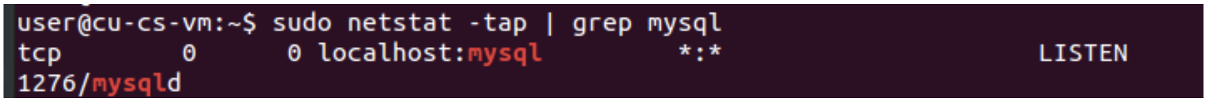
Set a password when prompted (use 3308 so in case you forget, the whole class can remind you).

Steps

1. Make sure the server is working

```
sudo netstat -tap | grep mysql
```

You should see:



```
user@cu-cs-vm:~$ sudo netstat -tap | grep mysql
tcp        0      0 localhost:mysql         *:*                     LISTEN
1276/mysqld
```

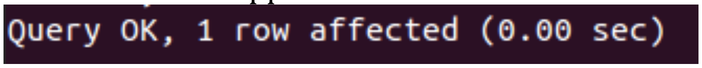
2. Access MySql via the CLI (where –u is for user, and –p is password)

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

3. Now inside the MySQL CLI, run the following:

```
CREATE DATABASE Lab;
```

You should see happiness:



```
Query OK, 1 row affected (0.00 sec)
```

4. Now tell it to use that database:

```
USE Lab
```

5. Create a tables to play with. To do so, it is easiest to add them to a file then load them from the SQL CLI.
   Using vim or emacs, type the following SQL code to a file named 'db.sql'

   <u>Store</u>

```
CREATE TABLE IF NOT EXISTS `store` (
`Id` int(1) NOT NULL auto_increment,
`Name` varchar(40) NOT NULL,
`Qty` int(1) NOT NULL,
`Price` float NOT NULL,
PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=7 ;

INSERT INTO `store` (`Id`, `Name`, `Qty`, `Price`) VALUES
        (1, 'apple', 10, 1),
        (2, 'pear', 5, 2),
        (3, 'banana', 10, 1.5),
        (6, 'lemon', 100, 0.1),
        (5, 'orange', 50, 0.2);
```

   <u>Courses</u>

```
CREATE TABLE IF NOT EXISTS `Course` (
`Id` int(1) NOT NULL auto_increment,
`Name` varchar(4) NOT NULL,
`deptId` int(1) NOT NULL,
PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=8 ;

INSERT INTO `Course` (`Id`, `Name`, `deptId`) VALUES
(1, '111', 1),
(2, '112', 1),
(3, '250', 1),
(4, '231', 1),
(5, '111', 2),
(6, '250', 3),
(7, '111', 4);

CREATE TABLE IF NOT EXISTS `Dept` (
`Id` int(1) NOT NULL auto_increment,
`Name` varchar(3) NOT NULL,
PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

INSERT INTO `Dept` (`Id`, `Name`)
VALUES (1, 'CSC'), (2, 'MTH'), (3, 'EGR'), (4, 'CHM');
```

```
CREATE TABLE IF NOT EXISTS `Enrollment` (
`Id` int(1) NOT NULL auto_increment,
`count` int(1) NOT NULL,
PRIMARY KEY (`Id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=8 ;

INSERT INTO `Enrollment` (`Id`, `count`) VALUES
(1, 40),
(2, 15),
(3, 10),
(4, 12),
(5, 60),
(6, 14),
(7, 200);
```

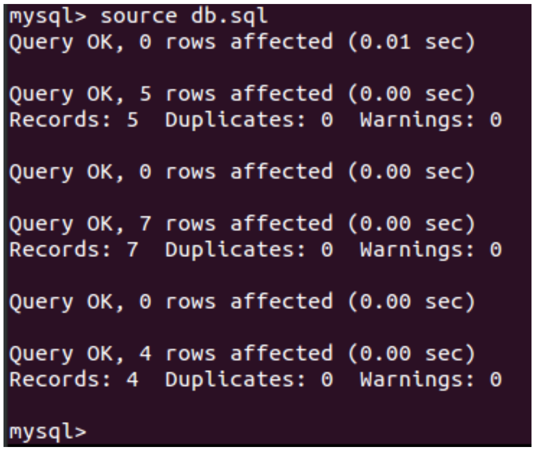
6. Now if you try to list the tables in the database, it will say an Empty set.

```
show tables;
```

7. Create the tables and add the content from the data file:

```
source db.sql
```

You should see more happiness:



```
mysql> source db.sql
Query OK, 0 rows affected (0.01 sec)

Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

Query OK, 0 rows affected (0.00 sec)

Query OK, 7 rows affected (0.00 sec)
Records: 7  Duplicates: 0  Warnings: 0

Query OK, 0 rows affected (0.00 sec)

Query OK, 4 rows affected (0.00 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql>
```

Questions – write out the query to do the following (test inside your VM).
Create another text file with all your queries in it, use the file extension .sql

List all the items sorted alphabetically. Then list only the first 3. Then the last 3.

1. List only the items that are more than $1 per unit price
2. List all the items with their extended price (quantity * price)
3. List the total cost of all the items in the store
4. How many different items do we have in the store?
5. List all the CS classes.
6. What is the total enrollment over all the classes?
7. How many different classes are taught?
8. How many different departments are there?
9. List all the classes in the database, with the department name and the class name on the same line, e.g. CSC 111, CSC 112, ..., EGR 250, ... CHM 111
10. List the name of the CS classes so that they are output as "CSC111", "CSC112", etc... (in other words, concatenate department with class number.)
11. List all the information in the database, where each class appears on 1 line, along with its department, and its enrollment.
12. Use a tool to create an E-R Diagram of these tables.

To get credit for this lab exercise, show the TA and sign the lab's completion log.

More about databases:

- *NoSQL (check out MongoDB, it's popular lately)*
- *Other relational databases: MySQL, SqlServer, postgres, Oracle…*
- *Database triggers and stored procedures*
- *phpAdmin great tool for managing DB via GUI interface*