MySQL Database Primer I

This guide will demonstrate how to connect to a MySQL database as well as a number of basic SQL operations. SQL queries must be terminated with a semicolon (**;**) It is a common beginner mistake to omit the semicolon. If you do MySQL will not execute the command. It will wait for further commands or a semicolon.

Contents:

1. Requesting a MySQL Account

To request a MySQL account, please go to the link for MySQL account activation at http://uisacad5.uis.edu/mysql/. Supply the necessary information such as UIS NetID and password.

2. Changing Your Password

You can reset your MySQL account password at http://uisacad5.uis.edu/mysql/reset/

> **Note:**
> The MySQL database password does not accept special characters such the "@" or "/" sign. They are reserved characters.

3. Logging onto the MySQL Server

The uisacad5.uis.edu server hosts a MySQL database server. After logging into uisacad5 you can type the following and press enter:

mysql –u *NetID* –D *NetID* -p

> **-u NetID** tells MySQL to use your NetID as the username
> **-D NetID** tells MySQL to use the database which is named the same as your NetID
> **-p** tells MySQL to ask for your password

If successful, the SQL prompt appears:
mysql>
You can also connect by typing your password after the **–p** option, but this will expose your password if someone is watching your screen. It will also be displayed in your command history.

4. Creating a Table
Format:

CREATE TABLE <tableName> (
<columnname type, columnname type . . .>
);

Where: tablename is the name of the table
columnname is the name of the column
type is the type of data in that column such as: varchar(n) for strings, int for numbers, date for dates, etc.;

- varchar(n) - variable length string character, n number of charcters.
- date - date, by default they are specified as day-month-year as in '25-DEC-2006'
- int - integer or real value

Example:

```
CREATE TABLE student (
    netid varchar(8),
    name varchar(12),
    course varchar(6),
    grade int);
```

This command creates a table named student with four attributes defined as follows:
netid as a string with 8 characters string length
name as a string with 12 characters string length
course as a string with 6 characters string length
grade as an integer

5. Inserting a Row
Format:

INSERT INTO <tableName>
VALUES(<list of values for columnnames in order>);

Where:
tableName is the name of the table where a row is to be added.

Example:

```
INSERT INTO student VALUES('atest01', 'John Doe', 'CSC368',1);
INSERT INTO student VALUES('atest02', 'Jane Doe', 'CSC368',1);
```

6. Displaying the Contents of a Table
Format:

SELECT * FROM <tableName>;

Example:

```
SELECT * FROM student;
```

Result:

```
+-----------+-------------+-----------+--------+
| netid    | name       | course | grade |
+-----------+-------------+-----------+--------+
| atest01 | John Doe | CSC368 |     1 |
| atest02 | Jane Doe | CSC368 |     1 |
+-----------+-------------+-----------+--------+
```

7. Editing a Row

Format:

UPDATE <tableName>
SET <list of columnname = new value>
WHERE <list of columnname = value of an existing columnname>;

Where:

tableName is the name of the table to be edited
list of coulumnname - the column name or names
new value - the edited value or values
values of an existing columnname - the current value in a row

Example:

UPDATE student SET name = 'John Doe', grade =4 where netid='atest01';

8. Deleting Contents of a Table

Format:

DELETE FROM <tableName> **[WHERE** <condition>**]**;

Example 1:

DELETE FROM student WHERE netid= 'atest01';

Note:

Example 1 will permanently delete all records with a netid of atest01 in the STUDENT table.

Example 2:

DELETE FROM student;

Note:

Example 2 will remove all contents of the table.

9. Committing Data

Format:

COMMIT;

Note:

- to ensure that the data added or modified become part of the database
- changes will be visible to other database sessions
- many commands automatically result in commit such as creating or modifying a table

Format:

ROLLBACK;

Note:

- restore data to the last commit or modification of table

10. Altering a Table

Format:

ALTER TABLE <tableName>
ADD (<new column name type(number of characters), ...>);
ALTER TABLE <tableName>
MODIFY (<existing column name type(number of characters), ...)>;
ALTER TABLE <tableName>
DROP COLUMN <existing column name, ...>;

Example 1:

ALTER TABLE student ADD semester VARCHAR(8);

Note:

This will add a new column called semester with varchar type and 8 characters in the table student in the database.

Example 2:

ALTER TABLE student MODIFY semester VARCHAR(6);

Note:

This will modify the varchar2 length of an existing column called semester into 8 characters in the table student in the database.

Example 3:

ALTER TABLE student DROP COLUMN semester;

The column semester is dropped.

11. Deleting a Table

Format:

DROP TABLE <tableName>;

Example:

DROP TABLE student;

Note:

This will permanently remove the table from the database.

12. Displaying Contents of your Database

To see all tables you have created in your database:

Format:

SHOW TABLES;

To display the attributes of a table:

Format:

DESCRIBE <tableName>**;**

To display the number of rows in a table:

Format:

SELECT COUNT(*) FROM <tableName>**;**

13. Copying Tables

Format:

CREATE TABLE <newTableName> **LIKE** <tableName>**;**
INSERT <newTableName> **SELECT * FROM** <tableName>**;**

Example:
CREATE TABLE student_new LIKE student;
INSERT student_new SELECT * FROM student;

14. Loading Commands from a File
To run all commands located in a text file use the SOURCE command.
Format:
SOURCE /path/to/*file.sql*

Example:
SOURCE /home/kevin/commands.sql

15. Logging a Session
Format:
tee */path/to/file.log*

Example:
tee /home/kzepp2/mysql.log
Note:
The file will continue logging until you exit mysql.

16. Help
Format:
HELP [command]

Example:
HELP select

17. Quitting mysql Session
Format:
QUIT
or
EXIT