Table of Contents

Assignments will generally ask you to create 8-12 queries against specific tables. There are some rules that you need to follow so that I can grade your work. You will create your work in a script file and then run that script to produce an output file. I need to get files from you that are gradable and I will need to see both the SQL that you write and the result of running that code.

I assume that you will be developing and testing your code in interactive mode, possibly using a gui client. This discussion does not address that issue. What this talks about is how you produce the assignment for me to grade. You may find this technique useful for documenting other work that you do and need to save.

1. Running a script from the command line

Create a directory in a convenient place on your hard drive. Use this directory to store your script files and also to save your output files. I am using a Windows machine and have a directory named db_scripts at the root of my C: drive. I like having the scripts stored in this location since it makes the pathname short.

I assume you have created a database named a_testbed and that it contains the zoo table.

1.1. The script file

Create the following file in a text editor. There is an intentional error in the third select. Save this file in your script directory with the filename demo_01.sql.

Put your name as a comment at the top of the file.

```
/* Joe Student  */
use  a_testbed;

/*  TASK 01 */
select z_name, z_type
from zoo;

/*  TASK 02 */
select z_name, z_type
from zoo
where z_type ='giraffe';

/*  TASK 03 */
select z_name, z_type
from zoo
where z_typ ='giraffe ';

/*  TASK 04 */
select z_name, z_type
```

```
from zoo
where z_type ='bear';
```

1.2. Running the script to produce a spool file

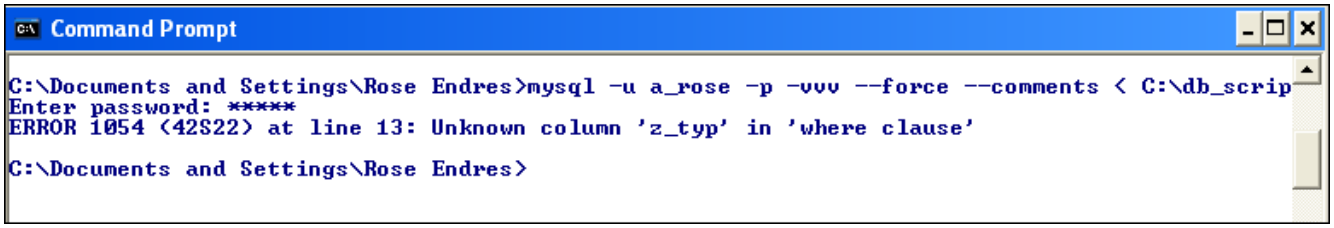
Now open a window with a command prompt and run the following command replacing the user name with your user name and adjusting the file paths as needed. This is a single line.

```
mysql -u a_rose -p -vvv --force --comments < C:\db_scripts\demo_01.sql >C:\
db_scripts\demo_01.lst
```

The command uses input redirection with the < symbol and output redirection with the > symbol. It is reading the sql commands from the sql file and it is writing the output to the lst file. It includes a number of options discussed in this document.

NOTE:

- The screen shot does not show the entire command- it is pretty long.
- This command is given from a command prompt; it is **not** given from a mysql prompt.
- You are prompted for your password
- The error message for task 03 shows up in the command window, not in the output file.
- You can edit your file(the slq file) can rerun this command. On a Windows system use the up arrow to get back to the command line to rerun it.
- Since this is a rather long command line with a bunch of option I keep a copy of that command in a text file and edit it for the filenames. I can then copy and paste it to the command prompt.
- If you do this as shown, you will get a new file in your directory demo_01.lst.



A student (Ameer Navidi Summer 2103) suggested this technique for getting the path to the file into the command line. After writing:

```
mysql -u yourusername -p -vvv --f --comments <
```

Mac terminology: Find your .SQL file in a Finder, and drag the file from finder into the terminal window. The terminal file will then write the file path to the file that was dragged into it (which is your .SQL file).

Then type the ">" symbol and then drag the .SQL file into the window again, so that Terminal would write the file path again. The final step is to change the .SQL in the second filepath to .LST so that the file created is a .LST file.

Windows terminology: Find your .SQL file in a File Explorer window, and drag the file from File Explorer into the Command Prompt window. The Command Prompt window will then write the file path to the file that was dragged into it (which is your .SQL file).

Then type the ">" symbol and then drag the .SQL file into the window again, so that Command Prompt would write the file path again. The final step is to change the .SQL in the second filepath to .LST so that the file created is a .LST file.

1.3. The spool file

This is the spool file that was produced.

```
/* Joe Student  */
use  a_testbed
--------------

Query OK, 0 rows affected (0.00 sec)

--------------
/*  TASK 01 */
select z_name, z_type
from zoo
--------------

+---------+-----------+
| z_name  | z_type    |
+---------+-----------+
| Sam     | Giraffe   |
| Abigail | Armadillo |
| Leon    | Lion      |
| Lenora  | Lion      |
| Sally   | Giraffe   |
+---------+-----------+
5 rows in set (0.00 sec)

--------------
/*  TASK 02 */
select z_name, z_type
from zoo
where z_type ='giraffe'
--------------

+--------+---------+
| z_name | z_type  |
+--------+---------+
| Sam    | Giraffe |
| Sally  | Giraffe |
+--------+---------+
2 rows in set (0.00 sec)

--------------
/*  TASK 03 */
select z_name, z_type
from zoo
where z_typ ='giraffe '
--------------

--------------
/*  TASK 04 */
select z_name, z_type
from zoo
where z_type ='bear'
--------------

Empty set (0.00 sec)

Bye
```

This file is called a spool file and this is one of the files that you submit for an assignment. Some things to note about the spool file.

- The comments such as your name and the task numbers appear in the spool file.
- The sql queries appear in the spool file
- The output of the query follows the query for each task.
- The file ends with Bye

- The output has a lot of extra blank lines and dashed lines etc. Don't worry about that

Note carefully the results of the queries for task 03 and task 04. Task 03 contained an error (I mistyped z_type.) There is no output and no response in this file. The error message went to the screen. But Task 04 did run- I did not have any bears in my table so there was no table style output- but I did get an "Empty set" message.

You always need to read you spool (LST) file. If there is no result table for a query- find out if that is because the query has an error and did not run- or if the query simply has no output to display.

2. The options on the command line

As long as you produce the correct type of output spool file you don't have to know what those option do, but this is why you need them.

2.1. -u

This is for your user name.

2.2. -p

This is to indicate that you will supply a password when asked.

2.3. -vvv

The first thing we need is to see the queries that are being run and the output formatted as rows and columns. This is done with the -vvv option.

If I cannot see your sql queries in the spool file, then you get a grade of 0 for the assignment.

2.4. -- force

This forces your script to continue to run even if you have an error in your script. Suppose you have 10 queries and query #3 has an error. If you do not include the force option your spool file would stop with query 2 and you would get no points for anything after query 2. You can use -f instead of --force if you want.

(You will still see an error message in the command window but the script will continue to run.)

2.5. -- comments

I need to see the comments to help me keep track of the various steps in the assignment and I need your name displayed. For this we include a --comments option on the command line.

If I do not see the task number comments and your name comments in the spool file, then you will lose points on each query.

3. Working with the assignment script files

I have provided a template file for the assignments which you are required to use. The easiest way to handle this is to create a folder to store your script files. Copy the template into the folder. Change the name of the file to A01_yourLastName.SQL . Obviously I do not mean to literally use the letters yourLastName. If you have a common last name, then use A01_yourLastName_yourFirstname.SQL.

Open the file in a text editor and REPLACE the line that starts "replace this line" with a comment giving your name. For example, if I were a student, I would use the following line.

-- Rose Endres
Save the file. Be certain that the file name extension is SQL.

Now that you have modified the template with your name, you can go ahead and make 16 copies- one for each assignment. Change the file names to match the assignments.

```
A01_yourLastName.SQL
A02_yourLastName.SQL
A03_yourLastName.SQL
```