Table of Contents

This document discussed some of the specific rules for assignment script and spool files.

1. Assignment Template

Rule 1: Use the template file that I have provided. This file is included on the download page and will work for almost every assignment. If you do not use the template file you are apt to repeatedly lose points on assignments. There is a 10 point penalty for not using the template and additional penalties for skipping parts of the template in your script.

These are the first few lines of the template file:

```
replace this line with a comment giving your name; I do *not* want your id,
assignment number, etc.- just your name as it appears in Insight.

\W    /* enable warnings! */

use   -- put the database name into this command and remove the comment. Keep
the ending semicolon;

/*  TASK 00 */
select user(), current_date(), version(), @@sql_mode\G

/*  TASK 01 */
replace this line with your SQL for task 01

/*  TASK 02 */
replace this line with your SQL for task 02

/*  TASK 03 */
```

- Start the script file with a comment with your full name. Do not include any other header comments. I know which class and assignment this is- you do not need to tell me; I do not need your student id, email, the current date, etc. Some instructors want a more complete documentation style- The documentation for this class is very simplistic. Note that is says to "replace this line" and you need to have your name as a comment.
  `/* your name */`
  how hard could that be?

- The next line is \W. leave this in the script file. If it does not run, then you are not using the correct command to create the spool file.

- Select the database used for the assignment.
  `use a_testbed;`
  how hard could that be? Note that the comment is gone!

- There is a Task 00. Leave this in the script without any change. Note that it is executed with the \G statement terminator. If it does not run, then you are not using the correct command to create the spool file.
  how hard could that be?

- Include the task number as a comment for each task in the assignment. These are in the template. Do not change the Task comments. I want this style and wording of the task comments.

It is essential that you look at the output file produced. Would you be able to grade this? Can you find the SQL used for a particular task followed by its result? When I grade assignments, I grade all assignments at the same time- first grading all of the task1 steps, then all of the task 2 steps etc. So I need to be able to find your sql and your results quickly.

Since this is a script/spool process, the files you turn in should have no mistyped commands. When you read your results file, if you see a mistyped SQL command, you should correct the script and rerun it. It takes very little time to run an assignment script to an output file. Typing errors which remain in a results file will cost you points. You can experiment with the queries one at a time until you get them correct and then add them to your script file.

If the SQL query is not included in the spool file, you will loose major points.

If the various steps are not correctly numbered, you will loose major points.

Sometime you will find a query that you cannot do. Include the comment for the Task number and a comment that you skipped that task.

```
/*  TASK 08 */
-- Omitted
```

2. Modifying the script file

These are some ideas about how to handle the mechanics of creating the scripts for the assignments.

You should create a folder/directory on your local system to save your work. The name of my directory is c:\db_scripts.

Download the template file into that directory and change the name of the file to A01_yourLastName.SQL . Obviously I do not mean to literally use the letters yourLastName, but I did not want to make 100 different copies of this document -one for each student. If you have a common last name, then use A01_yourLastName_yourFirstname.SQL.

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

```
/* Rose Endres */
```

Next there is a use command; you will need to edit this for each script to start with the proper database. Right now it says

```
use   -- put the database name into this command ;
```

For A01, you will edit it to

```
use a_testbed;
```

For A02, you will edit it to

```
use a_vets;
```

Then there is a pair of lines; the first is a comment and the second is a command that will display some information; leave this in the script.

```
/*  TASK 00 */
```

```
select user(), current_date(), version(), @@sql_mode\G
```

Next follows a set of comments for task numbers that corresponds to the tasks in the assignment. Leave these comments in the script and add your SQL after the appropriate task number. Do not change the comment style or wording.

```
/*  TASK 01 */
```

Save your file. Be certain that the file name extension is SQL. (I don't care about the case of the filename but people seem to notice the upper case letters.)

3. Make copies of script file

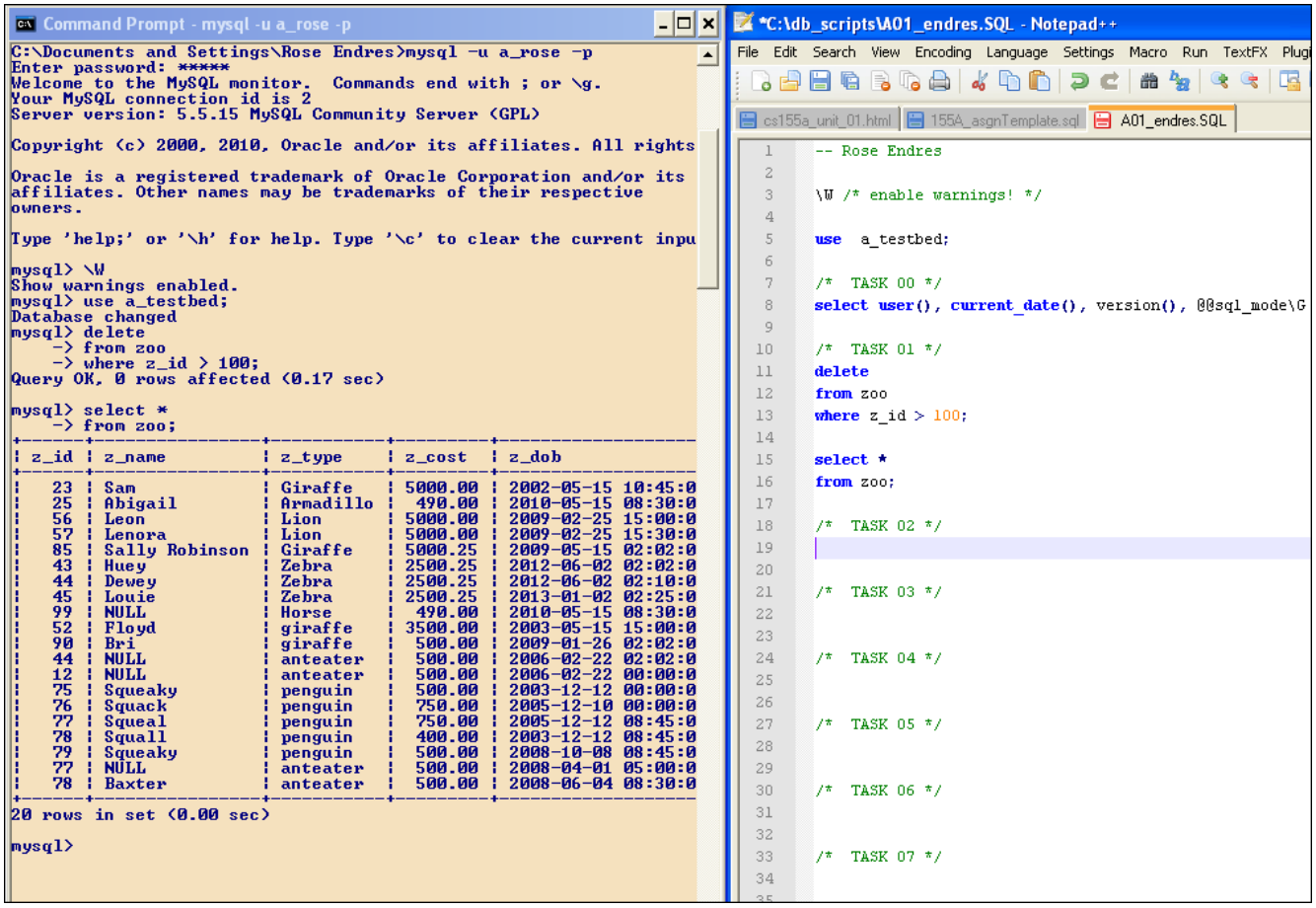
Now that you have modified the template with your name, you can go ahead and make 16 copies- one for each assignment. Change the file names to match the assignments.

```
A01_yourLastName.SQL
A02_yourLastName.SQL
A03_yourLastName.SQL
A04_yourLastName.SQL
```

4. Filling the script

You can open the script file in a text editor and fill in the SQL commands for each task and test them in the interactive mysql client. Be certain you are in the correct database- by running the use command.

What I usually do when I set up assignments or demo files is have the script file open in a text editor and a window open for the mysql command line client. I enter the sql query in the script file and then copy and paste it into the client window to run and test. The following shows the two windows I am working with.



If my sql works correctly, then I can save the text file and go on to the next task. If the sql query is incorrect then I can correct it in the text editor and copy it again. I am a poor typist and this way it is easier for me to correct errors than if I try to create the queries directly in the client window.

Save the script as you work. If you have troubles with one of the tasks, you can skip it temporarily and go on to the next. You can do some of the tasks, save the file and take a short break and then come back to work on other tasks. You need that script file to run the assignment, so it makes sense to me to build it up this way.

You should test the script to spool process occasionally as you build the scripts but the sql you execute in the mysql interactive client window should run the same way with the script-to-spool process. But be certain to test that early enough that if you have a problem you can fix it in time to turn in the assignment on time.

If you are using a GUI client that lets you build the script file in the client window that is also ok. But you will still need to run that script from the mysql command line. It is a good idea to test this a few times to be certain your gui client is compatible with the script-to-spool process.

I do get people who turn in a spooled file that is mostly empty and they do not get a chance to correct this. READ YOUR SPOOL FILE. I have to read it; it is only fair that you read it also.

The spooled file should contain

- your name as a comment
- the use command to switch to the correct database
- Task 00 as provided in the template
- the task number for each task as a comment as provided in the template
- the sql query(queries) needed at each step
- the output for each step

5. Testing your script-to-spool process

See the direction in 01-06 Section 3 for Running script. Read your LST file. If it has error messages or other errors, correct the script and resave it, then rerun the script-to-spool.

I keep a text file with the command lines I run the most often. That is a long command line and I do not want to rewrite it from scratch. So I have a text file (command_mysql.txt) with copies of the command line and I copy and paste them into the command window changing the names of the file from A01 to A02 etc.

6. Turning in the assignment

After you have written and tested your script and have created the spooled file and have read it for possible problems, then it is time to zip the two files. You can use the windows menu (send to compressed folder) or other file compression techniques that open with 7-Zip. The compressed file should use your name (such as A01_endres.zip)

7. File name problems I have seen in the past and do not want to see again

You turn in files with the name A01.txt and A01.1st-- you lose 10 points for having the wrong file extensions and 10 points for the files not including your name. (did you notice that the extension was a digit 1 and not the letter l?)

You turn in files with the name A01.sql and A01.lst-- -- you lose 10 points for the files not including your name.

There are two people in class with the same last name. You can check the participants listing in Insight and if you have the same last name as another student, please use the naming pattern A01_yourLastName_yourFirstName.SQL

The slq file and the lst fies are named correctly but the zip file has a name such as A01.zip.You lose 10 points.

Following directions is actually important. Use your creativity in writing queries- not in naming files.