Due Date: August 25, 2015 11:00 PM [late date Sep 01. This is the only assignment with a one week late date. and Assignment 02 is also due Sep 01, 2015.]

Points: 35 point max

These tasks focus on how to write SQL statements, how to use Oracle and the SQL*Plus client, and how to write a script and produce a spool file for an assignment. The primary purpose of this assignment is to check that you can use the software and create the script and spool file as required for the assignments. I will give you the SQL for some of the tasks in this assignment.

Preliminary Tasks. These steps are NOT to be included in the script

These are tasks you need to do before you run the script. Generally these preliminary tasks will include steps that you do only once- such as creating tables and inserting the original rows of data that I provide.

- Run the SQL script to create the table zoo_2015; use the file from the download page
- Run the SQL script to populate the table zoo_2015; use the file from the download page. This will insert 10 rows.

General Directions

Use the zoo_2015 table.

Use the template provided for assignments for your queries. Use the assignment script template posted on the download page as the basis for this script. Create a script file with the queries described in the various tasks listed below. After you have written and tested your queries and it follows the assignment guidelines, run the script to produce the spool file. Zip the script file and the spool file together into one compressed file and upload that file to the Insight page for this assignment. Be certain to use the file names as described in the Assignment Rules document. The script filename must follow the pattern A01_yourLastName.SQL (substitute your last name in the file name) and the spool file will have the same name with an LST extension A01_yourLastName.LST .

Read your spool file before you submit the assignment. It should contain

- the set commands as provided in the template
- your name as a comment
- Task 00 as provided in the template
- the task number for each task as a comment
- the sql query(queries) needed at each step
- the output for each step

The file names for the files to be turned in are important. I download all of the files into a local folder for grading. The download programs finds only file names that start with the proper letters (here A01). I also need your name in the file name. The case of the filename is not significant.

Remember to include the task number as a comment at each step. These comments are in the template.

Read your spool file before you turn it in. Does it show the task numbers, the SQL, and the result for each query? If not, correct the script file and rerun the spool before you turn it in.

For this assignment only, I will grade files as they are posted in Insight so you should check Insight in the early evening the day after you have posted this assignment. If there are any significant errors, you will have a chance to resubmit this by August 25 for full credit and by Sep 01, 2015 11:00 pm as a late assignment (for this assignment only).

Review the general Assignment Rules to see that your work is meeting the assignment guidelines.

For most assignments you will not have as long a set of directions but you may need all of these to get started.

Tasks

At the top of the script file, you should several lines of set command as provided in the template. You have a comment with your name.
You will also have Task 00 which displays the version of SQL Server you are using and some other pieces of data.

Task 01: Copy and run the following two SQL statements. The first will remove any rows from the zoo_2015 table where the z_id value is greater than 100 and the second will show the rows in the table. At this point you should have the original rows that I supplied in the script. The rows I provided in the inserts script have a z_id value less than 100 and will not be deleted.

```
delete
from zoo_2015
where z_id > 100;

select *
from zoo_2015;
```

Task 02: (3 insert statements) Add an additional 3 rows to the table. For the animal id, use an animal id value **that is greater than 100**. For the rest of the data, use any data values you want. Post the sql for these 3 inserts to the Insight forum named A01_inserts.

Task 03: (6 insert statements) Copy and run at least 6 good inserts from the Insight forum named A01_inserts. The more rows you have in your table, the better for experimenting with it. You cannot complete this step until some other people have already done their inserts.

Task 04: Copy and run the following SQL statement. It will display the rows in the table.

```
select z_id, z_name, z_type, z_cost, z_dob, z_acquired
from zoo_2015;
```

Task 05: Write and run the SQL statement so that it shows only three columns: the type of animal in the first column, the animal's name in the second column and the cost in the third column. Sort the rows by the animal type with a secondary sort on the name. (I am sure you can figure out how to do this.)

Task 06: We want to see the id, name and date of birth of all of the zebras in the table. Display the id of animal in the first column, the animal's name in the second column and the date of birth in the third column. Sort by the date of birth with the youngest animals displayed first.

For the last tasks, I am asking you to display some data **about** your tables. The other queries asked you to show data that is **in** your table. It is OK if you do not fully understand these queries.

Task 07: Give the following command which shows the names of your tables.

```
select tname
from tab;
```

Task 08: Give the following command which shows and some information about the table columns.

```
desc zoo_2015
```

Note: If you are using the CCSF Oracle system, I would expect you to have only a few tables at this time: zoo_2015 and possibly the vet series of tables. If you have tables from a previous semester of this class, you should overwrite them with this semester's data; the data in the tables may change with each semester.
You may see tables with names such as
BIN$ivzYov4fS5qPJdqGPpoeAQ==$0
You can ignore those names.