Due Date: Tuesday March 18, 2014 11:00 PM
Points: 35 points max
Turn In: The script and spool files turned in via the assignment drop box

General Directions

This assignment uses the tables from the a_vets database.
Do not worry about formatting or column widths unless the task specifically says to do so. The sample data might not show the actual results. Use the column aliases shown in the sample display.
For question that refer to the current year, previous year, last year etc, do not use a literal number; derive the value from the system date.

Questions that ask for the total exam fees should display the total fees for an exam or the total fees for an animal or client depending on the task.
Some of the tasks may be easier if you use variables. If you use variables, include a **single** select statement that displays the values of all variables used in that task. Variable names must be meaningful. Do not assign data from a table into a variable- we have not discussed that technique. Each task must define all of the variables that it uses within that task sql. Do not define variables at the top of the script to be used later.

Several tasks ask you to filter for animal types in defined sets of values called **categories**; use these lists for those tasks. Some animal types are in more than one category.

Category	Animal type
Reptile	chelonian, crocodilian, lizard, snake
Rodent	hamster, porcupine, dormouse, capybara
Restricted	capybara, crocodilian, dormouse, hedgehog

Things to think about as you do the assignment.

- What is the difference between Count (…) and Count (distinct …)? If the attribute value you are counting can occur more than once, do you want to count each occurrence? How is this affected by joining several tables in the From clause?
- Do you need inner join or outer joins? (Hint- you will not need a full join.) Do not use an outer join if an inner join is sufficient; an outer join requires more resources. Do not use an outer join and then throw away the rows you added due to the outer join.
- How do outer joins affect which column you use for the aggregates?
- Case expressions are useful for some situation where you need to do different things with different values. If your only concern is if something is null or not, then coalesce takes less typing.
- Ignoring the definition of the categories, given in the books document, is not going to help your assignment score.

Tasks

Task 01: Consider all animals that have exam records. Display the animal id and type, the number of exams the animal has, and the total fees for those exams. Display the data in animal id order.

an_id	an_type	NumberExams	TotalFees
12035	bird	1	122.00
15001	chelonian	3	265.00
15002	chelonian	2	130.00
15401	lizard	1	105.00

Task 02: For each client we have in the clients table, display the client id and last name. If the client's animals have any exams last year display the average fee for the exams and the highest fee for

their exams. Display the string ' no exams' if there are no exams. This could be because the client has no animals, or because the animals have no exams. Order by the client id.

```
+--------+-------------+------------+------------+
| CL_ID  | LastName    | AverageFee | HighestFee |
+--------+-------------+------------+------------+
|   1234 | Simon       |      36.45 |     375.00 |
|   2345 | Bostone     |     125.75 |     275.00 |
|   3456 | Dawkinson   |  no exams  |  no exams  |
```

Task 03: Display the number of exams we had in the previous quarter and the number of animals we have which had at least one exam in the previous quarter. The term "previous quarter " means any date in the quarter before the current quarter.
If you run the query in Oct 2013, that is the third quarter of 2013. the query will return data for the second quarter of 2013. If you run the query in Feb 2013, the query will return data for the fourth quarter of 2012.

```
+-------------+---------------+
| NumberExams | NumberAnimals |
+-------------+---------------+
|          78 |            12 |
+-------------+---------------+
```

Task 04: Which reptile has the most exams? Display the animal id and name. Consider there might be ties for first place- in that case all tied animals should be returned.

```
+-------+---------+
| an_id | an_name |
+-------+---------+
| 87654 | Slim    |
+-------+---------+
```

Task 05: Display the id and last name of all clients with exam fees totaling at least $300.00 and no more than $900.00.

Task 06: We want to know how many animals we have in the animals table in each of the indicated animal types or **categories**. Display the result as a single output row. The animal types and categories are:

- Reptiles
- Rodents
- Restricted
- capybara
- hedgehog

There is also a column for all animals.
If an animal is in more than one category, then it counts in each of those categories..

```
+----------+---------+------------+----------+----------+-------------+
| Reptiles | Rodents | Restricted | Capybara | Hedgehog | All Animals |
+----------+---------+------------+----------+----------+-------------+
|       23 |     123 |        215 |        3 |      115 |         607 |
+----------+---------+------------+----------+----------+-------------+
```

Task 07: We want a display of the exam dates (year and month only) and the number of exams in each month and the total fees for each month.
There is one row for each month that we have in our exam headers table. The first column is the year and month with the month shown as the three letter abbreviation. The second column is the number of exams that month. The third column is the total fees for that month. The display is sorted by the year and month in calendar order. Some people find the sort difficult; it is not optional, but turn in what you can do even if it is not sorted properly.

Year-month	NumberExams	TotalFees
2013-Jan	21	4100.00
2013-Feb	15	2339.50
2013-Mar	31	309.00
2013-Apr	56	2850.00
2013-May	125	19817.21
2013-Jun	13	177.25
2013-Aug	10	870.50
2013-Oct	45	2450.00
2013-Nov	48	3330.00
2013-Dec	97	8770.50
2014-Jan	86	7592.00

Task 08: For each client that we have in the clients table, list the client id and last name, the animal id and name and the date of the most recent exam for that animal.
Display messages- see the sample display:
If the client has no animals, then display "No animals' in the column for the animal id; the animal name and exam date are left blank.
If the client has animals, but an animal has no exams, then display 'No exam' in the exam date columns.
Suggestion: work out the main logic of getting the most recent date first; this will have nulls for some data. When that works, then handle the special messages one at a time. How do you know if a client has no animals? How do you know if an animal has no exams?

Client ID	Client	AN ID	AN Name	Most Recent Exam
2678	Tripp	11111	Buttons	No exam
2678	Tripp	22222	Mittens	17/06/2013
2678	Tripp	22233	Soxs	23/07/2013
3030	Bender	12323	Boots	04/04/2014
3030	Bender	14589	Primrose	11/06/2013
3030	Bender	21002	Petunia	No exam
3489	Tobias	No animals		
3998	Milton	34356	Zinnia	29/03/2014