

Due Date: February 17, 2016 11:00 PM

Points: 35 point max

## General Directions

This assignment uses the tables associated with the vets database.

- 1) Use the fewest tables possible for the queries. (This is a general assignment rule; it is listed here only for emphasis.)
- 2) Any joins must do the join in the From clause; use the condition join syntax. If you do the join in the Where clause, you will not get any credit for that task. (This is a general assignment rule; it is listed here only for emphasis.)

If you have not read the document on the vets table that came with the zip file for vets, please do so now.

### Using the fewest tables possible for the queries.

For example: if I asked you to display the animal's date of birth and the client id for each cat, you should write the following. You do not need the client table since the client id is in the animals table.

```
select an_dob, cl_id
from vt_animals
where an_type in ('cat');
```

If I asked you to display the animal's date of birth and the client state for each cat, you should write the following. Now you do need the client table because you need to display the client's state value.

```
select an_dob, cl_state
from vt_animals
join vt_clients on vt_animals.cl_id = vt_clients.cl_id
where an_type in ('cat');
```

You need to write queries that follow the wording of the task descriptions. For example if I ask you to display the last names of clients who live in New Mexico, you can filter on the column: cl\_state and the literal value 'NM'. You may not "read the table" find the client id values for those clients and then filter on the cl\_id. That is not the way to write a query. You can look at the table to see if your result makes sense but your query needs to work correctly if it is run against the table when there are thousands of rows of data that change frequently.

Using the state abbreviation for New Mexico is not a problem. You can look at the clients table and see that the states are all two letter abbreviations and you can look up the two letter abbreviation for states.

## Tasks

- Task 01:** We want to see the following data for each cat and dog born in 2009 or earlier. Client id, animal id, animal type and animal name.
- Task 02:** Display the exam id, exam date and service id and the fee charged for each exam done on a rodent. Order the rows by the exam id and service id.
- Task 03:** Display the staff id and name and exam date for each exam, but exclude exams done on a reptile or a bird. Sort by the exam date. The staff name is displayed in a single column as shown here.

```
+-----+-----+-----+
| stf_id | staff           | ex_date           |
+-----+-----+-----+
|      29 | Sandy Helfen    | 2015-10-10 09:15:00 |
|      29 | Sandy Helfen    | 2015-11-03 14:30:00 |
|      15 | Eliza Dolittle  | 2015-11-10 10:30:00 |
```

- Task 04:** The vet is concerned about exams where the exam fee charged is less than \$25.00 or more than \$200.00. Show the client id, exam id and exam date, the service id and the exam fee for these exams.. Order by the exam date and the service id.
- Task 05:** Display all services which include the word 'Feline' but not 'Dental' in their description. Display all of the columns in the services table for the matches. Sort by the service type and by the service id within the type.

The following tasks ask you to think more carefully about testing. They are more realistic tasks. The tasks ask you to display only the client id and name. While you are testing the queries you probably should display more columns to help you test your query. You should also check that data in the tables to see which clients you should be returning. But when you run the final script to spool, be certain that these tasks select only the client id and last name.

For the next two tasks, consider the following data (which might not be in our tables)

```
Client 5689 has one dog and no other animals
Client 408763 has three dogs and no other animals
Client 4534 has three birds and no other animals
Client 411 has three dogs, a hedgehog, and a lizard
Client 7212 has no animals
```

For that data:

Task 06 would return clients 5689, 408763, and 411 because each of these has a dog (possibly more than one dog). Task 07 would return clients 4534 and 411 because they have an animal which is not a dog. It is not relevant that client 411 also has a dog. The task would return client 411 only once even if he has two animals that are not dogs.

When you write the next two queries, the only filter you are allowed to use is a filter that refers to 'dog'.

- Task 06:** We want to see the id and last name of each client who has a dog. Display the client information only once even if the client has more than one dog. Order by the client id.
- Task 07:** We want to see the id and last name of each client who has an animal that is not a dog. Suppress duplicates. Order by the client id.
- Task 08:** We want to see the id and last name of each client who has a rodent. Display the client information only once even if the client has more than one rodent. Order by the client id.
- Task 09:** We want to see the id and last name of each client who has an animal that is not a rodent. Suppress duplicates. Order by the client id.

The End