INF1343, Winter 2012, Answers to Week 2 Exercises

*Those exercises will **not** be graded but similar questions and problems may appear on quizzes and the exam. You can work on those exercises together with other students.*

Unless indicated otherwise, answer each question using just **one** query. Your query should also be designed so that it would produce a table that contains **only the information needed to answer the question** and nothing else. For example, if you are asked to find out who owns the youngest pet, your query should produce a table with just one cell (not counting the header row) showing the name of the person who owns the youngest pet. It should **not** show the name of the pet or its age. Even more importantly, it should **not** show information about any other pets. On the other hand, if you are asked who are the *names* and *owners* of the *two* youngest pets, then your query should produce a table with *two* rows, one for the youngest pet and one for the second youngest pet, with both rows showing the name of the pet and the name the pet's owners.

Exercise 1

The **diveshop** database (available on the database server) is a database of a small company that sells diving vacation tours.

(a) Use the **vacation_order** table in this database to answer the following questions:

- What was the cost of the most expensive one-person vacation by for which the customer paid with cash? (One person vacation means only one person went on the vacation, rather than a group.)

  -
    ```
    select cost
    from vacation_order
    where
      no_of_people=1
      and payment_method="Cash"
    order by cost desc
    limit 1;
    ```

- What is the total cost of all vacations for which the customers paid with Visa?

  -
    ```
    select sum(cost)
    from vacation_order
    where payment_method="Visa";
    ```

- What was customer ID of the person whose vacation was the most expensive *per person.* (The per-person cost is the total cost of a vacation divided by the number of people who went on the vacation.)

  -
    ```
    select customer_id
    from vacation_order
    order by (cost / no_of_people) desc
    limit 1;
    ```

- What is the average cost of vacations per person? (Note: not per *order*, but per *person*. Hint: you will *not* need the AVG function for this.)

  -
    ```
    select sum(cost) / sum(no_of_people)
    from vacation_order;
    ```

(b) Use the **species** table in the database to answer the following questions:

- How many fishes have scientific names ("species_name") that end in "us" or "is"?

  -
    ```
    select count(*)
    from species
    where
      species_name like "%us"
      or species_name like "%is";
    ```

- What is the longest fish whose scientific name does *not* end in “us” or “is¨?

  -
    ```
    select species_name
    from species
    where
      not (species_name like "%us" or species_name like "%is")
    order by length_cm desc
    limit 1 ;
    ```

(c) Use the **site** table to answer the following question:

- From among the sites requires “Beginning” skill level and are at least 20 m deep, which one is the closest to the nearest town?

  -
    ```
    select site_name
    from site
    where
      skill_level="Beginning"
      and depth_m>=20
    order by distance_from_town_km
    limit 1;
    ```

Exercise 2

Design and create a database table that would store information about customers of an online store. In particular, the table needs to store customer’s name, postal address, email address, credit card number, as well as username and password. It should also record the last time the customer logged into to the store’s website and the last time the customer made a purchase. After you create the table enter data for 10 customers. Then write a query that would return the email addresses of customers who have a credit card on file *and* have logged into the website in 2011 but who have *not* made a purchase this year.