MOUNT ROYAL UNIVERSITY
DEPARTMENT OF MATHEMATICS & COMPUTING
COMP 2521 – Winter 2019
Assignment 2
DUE: Monday, February 25, 2019 at 11:59 PM

Purpose:

To write complex SQL queries using group by / having clauses and complex join constructs.

Important Note:

There are some important concepts in this assignment. Be sure that you think about them. ***Don't just get code that works from somebody else***.

Guidelines:

- **Drop all existing tables** on your database before you start to avoid having to deal with too many tables and their relationships.
- Use the script file `classicmodels.sql` in the folder to **create the database** to be used in this assignment. Read over the comments to understand the tables, fields and their relationships. Now, draw student database conceptual schema diagram (ERD) so that you can follow the database.
- **Build all required queries** step-by step – start with a simpler version of the problem and then gradually build towards the final version.

Submission:

Build the SQL script file using Notepad with queries ready for submission. Your submission should consist of one text file: `<YourLoginName>.sql` submitted through the Blackboard assignment submission drop box. Emailed submissions will not be marked.

Your SQL statements should be neatly formatted according to the standards shown in class. All queries should have upper case keywords (SELECT, WHERE, IN, HAVING etc.) and lower case table and column names. Each clause should be on a separate line, and appropriate, consistent indenting must be used.

Comment out lines which explain the query such as the query number, description (if any) etc. so that I will be able to execute the script file while I mark. Comments on the SQL script file are represented by '-- 'placed at the beginning of the line to be ignored during execution. There MUST be a block at the top of the file that includes your name, the date, and the course and assignment number. Marks will be *deducted* if not done as instructed. Remember, no tabs are allowed inside script files.

Each of your answers can only have one SELECT statement. No fair getting procedural– no subqueries allowed.

You must use column and table aliases where appropriate. Results should display with a readable, correct and complete column heading.

Queries:

1. Retrieve the office code, city and phone number of offices that are located in the U.S. and in France. Write this query in two distinctly different ways. Clearly number them as (a) and (b).
2. Retrieve all the line details of order number 10203. Sort the result by orderLineNumber.
3. Retrieve all payments with check numbers that have 'O' (the 15th letter of the alphabet) as the second character and has a zero as the second last character.
4. Retrieve the employee number, the employee number of the person (s)he reports to for those employees whose last name starts with either 'B' whose first name ends with 'y'.
5. Retrieve the employee number, last name, first name and job title of all employees. Retrieve only those employees whose office code is 2 or higher. Consider only those employees that report to employee number 1143 or 1088.
6. Retrieve all product codes and smallest quantity ordered for each.
7. Retrieve all product codes and show the total quantity ordered for each with appropriate column alias. Sort the result by the total from highest to lowest.
8. Retrieve distinct countries from the customer table. Write this query in **two** ways as (a) and (b).
9. Retrieve the number of customers in each country.
10. For each manager, retrieve their employee number and display the number of employees (s)he is a manager of. For example, employee number 1056 (Mary Patterson) has four employees reporting to her. Your query result should also show the number of employees that do not have managers. (For example, Diane Murphy is the President of the company; therefore, does not report to anyone.) For this row in your result, the employee number will be NULL.
11. Retrieve the employee number, first and last name of all employees along with the employee number of their boss (reportsTo) with their boss's first and last name. Order the result by the reportsTo field. Also, explain why 'Diane Murphy' does not appear as an employee on the result of this query.
12. Retrieve all employee(s) who do not report to anyone.
13. Retrieve names of all Canadian customers. Also include the corresponding sales rep last and first name, order number, product names, quantity ordered and unit price of each item on each order. Write this query using both the traditional join and using the JOIN keyword. Number them as (a) and (b).
14. Retrieve names and phone numbers of those customers who have more than 10 orders on the database. Use a column alias in this query.
15. Using order details and product information, for each item ordered in all orders, retrieve the buy price, unit price, the difference between them, and the percentage of profit margin made. Display results in descending order of this percentage. You may round the values for a neater presentation.
16. Retrieve the country, status, and number of orders with that status for all orders that are either disputed or cancelled.