PROG2220: S.Q.L. (MySQL)
Assignment 1

Assignment Type: INDIVIDUAL

Due Date: Week 4 (start of class)

Tip: Review the output files provided (XXA01Task1.out & XXA01Task2.out). *Display the question headers in your output files.*

Topic: Retrieve data from a single table

Task 1. Textbook Exercises

Save your solution to **XXA01Task1.sql**. Redirect your output to **XXA01Task1.out**.

Note: You can compare your solution to the textbook exercise solutions under G:\mysql\ex_solutions.

Assumption: You must have **G:\mysql\db_setup\create_databases.sql**. You must have completed **Lab 1**.

Q1. Textbook Exercise 3-06 (page 111) [2 points]

Important: Append "LIMIT 10" to your solution.

Q2. Textbook Exercise 3-07 (page 111) [1 point]

Note: Replace the word "joins" to "concatenates". Relational "joins" will be covered in Chapter 4 (retrieve data from multiple tables).

Q3 Textbook Exercise 3-08 (page 111) [1 point]

Q4. Textbook Exercise 3-09 (page 111) [2 points]

Q5. Textbook Exercise 3-10 (page 112) [2 points]

Q6. Textbook Exercise 3-11 (page 112) [1 point]

Q7. Textbook Exercise 3-12 (page 112) [1 point]

Task 2. My Guitar Shop (MGS) Database

Save your solution to **XXA01Task2.sql**. Redirect your output to **XXA01Task2.out**.

Install the MGS database: You must have **G:\mysql\mgs_ex_starts\create_my_guitar_shop.sql**. You must have completed **Lab 2**.

Q1. MGS Exercise 3-1 [2 points]

Write a SELECT statement that returns four columns from the Products table: product_code, product_name, list_price, and discount_percent. Add an ORDER BY clause to this statement that sorts the result set by list price in descending sequence.

Q2. MGS Exercise 3-3 [5 points]

Write a SELECT statement that returns these columns from the Products table: product_name, list_price, and date_added. Return only the rows with a list price that is greater than 500 and less than 2000. Sort the result set in descending sequence by the date_added column.

Q3. MGS Exercise 3-5 [5 points]

Write a SELECT statement that returns these column names from the Order_Items table: item_id, item_price, discount_amount, quantity, price_total (i.e., multiply the item price by the quantity), discount_total (i.e., multiply the discount amount by the quantity), item_total (i.e., subtract the discount amount from the item price and then multiply by the quantity). Only return rows where the item_total is greater than 500. Sort the result set by item total in descending sequence.

Q4. MGS Exercise 3-6 [5 points]

Write a SELECT statement that returns these columns from the Orders table: order_id, order_date, ship_date. Return only the rows where the ship_date column contains a null value.

Q5. MGS Exercise 3-8 [3 points]

Write a SELECT statement without a FROM clause that creates a row with these columns:

- price: 100 (dollars)
- tax_rate: .07 (7 percent)
- tax_amount: price multiplied by the tax rate
- total: price plus the tax amount

To calculate the fourth column, add the expressions you used for the first and third columns.

Assignment Submissions

Reminder: All printouts must be stapled and submitted **in the correct sequence**!

1. Download and print the **PROG2220_CoverPage.pdf** (PROG2220 Assignment Cover Page and Standards Marking Sheet) posted in eConestoga. **All the sections** of the Cover page must be filled.
2. A printout of **A1Marking.pdf**
3. A printout of **XXA01Task1.sql**, where XX is your initials in upper case letters
4. A printout of **XXA01Task1.out**
5. A printout of **XXA01Task2.sql**
6. A printout of **XXA01Task2.out**