**INFO 579 Week 8 Assignment**

Course:  INFO 579: SQL/NoSQL Databases for Data and Information Sciences

Module/Week: 8 - Week of October 9, 2023

**Topic: Beyond Simple Select Queries**

NOTE: The assignment document must provide the below information. Up to 10 points may be deducted due to the lack of the below information.

Student’s Full Name:

Course Title:

Term name and year: Example, Fall 2023

Submission Week: [example, Week 8 Assignment]

Instructor’s Name:

Date of Submission:

The above information must be provided at the upper left corner of the first page of the document.

Each answer(s) must be preceded by the question/ title of the topic/article of the assignment.

Acceptable File: Word (put the screenshots of the models) or PDF.

File Name Format: Name your file according to this convention: INFO579\_Week8\_Lastname.pdf. Submission must be made in a single document.

NOTE: There will be a 20% deduction of points for any late submission.

**Reading Requirements for this Assignment:**

Read the Data file titled, INFO579\_Week7\_Assignment\_Data.xlsx, uploaded to the Week 7 Assignment folder in D2L. You are using the same data that you already loaded into your database tables as part of previous week’s assignment. Also, read the slides uploaded to the Week 8 folder in D2L.

**NOTE-1:** All data display must be made based on SQL. No data display should be made using any graphical tool (e.g., MySQL workbench has such things).

**NOTE-2:** You cannot do any UPDATE statement to the data you already loaded in the tables. If you think, you have data integrity issue, you can empty the table(s) and reload. Also, you cannot create any additional table(s) to do this assignment.

**NOTE-3**: All SQL must specify join types (i.e., INNER JOIN, LEFT OUTER JOIN as appropriate):

**Example1**: This query with no explicit mention of JOIN type is NOT acceptable:

SELECT T1.TrainingName, T2.TrainerFirstName, T2.TrainingLastName

FROM Worker.Training AS T1, Worker.Trainer AS T2

ON T1.TrainingID = T2.TrainingID;

**Example2**: This the correct way of writing SQL with explicit mention of JOIN type:

SELECT T1.TrainingName, T2.TrainerFirstName, T2.TrainingLastName

FROM Worker.Training AS T1

INNER JOIN

Worker.Trainer AS T2

ON T1.TrainingID = T2.TrainingID;

**Assignment Instructions: Answer the following questions and provide screenshots, code, and results in a Word document or PDF file. You must put each question (question no. & description) before your answers. All screenshots must be easily readable. Always take the screenshot of what is needed instead taking the screenshot of the whole page. Grade points: 50 (5 points for each question).**

1. Write a query using UNION to display EmployeeID, FirstName, and LastName from the Employee table. The first part of the SQL (i.e., the select statement before the UNION) should use the filter, EmployeeID, 1, 2, 3. The second part of the SQL (i.e., the select statement after the UNION) should use the filter, EmployeeID, 3, 4, 5. Make your observations on the results of the UNION query.

2. Write a query using UNION ALL to display EmployeeID, FirstName, and LastName from the Employee table. The first part of the SQL (i.e., the select statement before the UNION ALL) should use the filter, EmployeeID, 1, 2, 3. The second part of the SQL (i.e., the select statement after the UNION ALL) should use the filter, EmployeeID, 3, 4, 5.

Make your observations on the results of the UNION ALL query. How is the query result different from the query results in Question 1?

3. Write a query to display the EmployeeID and total EquipmentCostAmount for each employee. Sort the results by EmployeeID. (Employee, EmployeeEquipment, Equipment tables).

4. Write a query to display the EmployeeID and total EquipmentCostAmount for each employee who incurs a total EquipmentCostAmount of more than $300.

Sort the results by EmployeeID. (Use Employee, EmployeeEquipment, Equipment tables). [Hints: use HAVING clause].

5. Write a query to display the EquipmentID, and the first 10 characters of EquipmentName [Hints: use SUBSTR/SUBSTRING]. Sort the results by EquipmentID.

6. Write a query to display the complete list of Training, and trainers (first and last name) available for each training. If the trainer's last name is not available, display a default value of 'TBD' by using the CASE Expression in the SQL. Sort the output by TrainingName and Trainers' first name.

7. Write a query to display the complete list of Training, and trainers (first and last name) available for each training. If the trainer's last name is not available, display a default value of 'TBD' by using the COALESCE function in the SQL. Sort the output by TrainingName and Trainers' first name.

8. Write a query using EXISTS operator to display the EmployeeID, FirstName, LastName, and HireDate of employees who work for the following departments: Accounting and Finance, IT Support, and Production. Sort the results by EmployeeID.

9. Write a query using EXCEPT operator to display the EmployeeID, FirstName, LastName, and HireDate of employees who do NOT work for the following departments: Accounting and Finance, IT Support, and Production. You must use these filters in your SQL. Sort the results by EmployeeID.

NOTE: In case your database does not support EXCEPT operator, you may consider NOT EXIST operator.

10. Write a query using the MINUS operator to display the distinct list of equipment NOT used by the current employees.

NOTE: In case your database does not support MINUS operator, you may consider NOT IN operator.

**End**