# **BTD210- Lab 8**

We, Muskan and Priya, declare that the attached assignment is our own work in accordance with the Seneca Academic Policy. No part of this assignment has been copied manually or electronically from any other source (including web sites) **or distributed to other students.**

Please work in **groups** to complete this lab. This lab is worth 2% of the total course grade and will be evaluated through your written submission, as well as the lab demo. During the lab demo, group members are randomly selected to present the answers to each of the lab questions. Group members not present during the lab demo will lose the demo mark.

Please submit the following files through Blackboard. Only one person must submit for the team.

* Lab8.docx

1. Add this declaration on the top of your file.
2. Specify what each member has done towards the completion of this work:

|  |  |  |
| --- | --- | --- |
|  | Name | Task(s) |
| 1 | Muskan | Answers |
| 2 | Priya | Answers |
| 3 |  |  |

Download the Lab8\_StoreCo.SQL file. Open Microsoft SQL Server Management Studio. Use *File > Open* to open the above SQL file and *Execute* it. Right Click on Databases in th*e Object Explorer* and click *Refresh*. Confirm that the STORECO database is created.

For the following questions, include

1. The **SQL command**
2. The **output** in text format,
3. **How many rows** are affected.
4. List all employees born in May. Show the list in the following format.

Last name First name Date of Birth

--------------- --------------- -------------

Williamson John 05/21/64 ...

SELECT EMP\_LNAME, EMP\_FNAME, EMP\_DOB FROM EMPLOYEE WHERE MONTH(EMP\_DOB) = 5;

EMP\_LNAME EMP\_FNAME EMP\_DOB

--------------- --------------- -----------------------

Williamson John 1964-05-21 00:00:00.000

Smith Sherry 1966-05-25 00:00:00.000

Olenko Howard 1964-05-24 00:00:00.000

(3 row(s) affected)

1. List last name, first name, and initial of all employees who have their first name or their last name starting with the same letter as their initial in alphabetical order (of last name).

SELECT EMP\_LNAME, EMP\_FNAME, EMP\_INITIAL FROM EMPLOYEE WHERE LEFT(EMP\_FNAME, 1) = EMP\_INITIAL OR LEFT(EMP\_LNAME, 1) = EMP\_INITIAL ORDER BY EMP\_LNAME;

EMP\_LNAME EMP\_FNAME EMP\_INITIAL

--------------- --------------- -----------

Jones Rose R

Williamson John W

(2 row(s) affected)

1. List the store names by the first 3 letters of their name, followed by 3 ’\*’s, followed by the last 3 letters of their name, as shown in the following sample output.

Funny String

------------

Acc\*\*\*ion

...

SELECT ( LEFT(STORE\_NAME,3) + '\*\*\*' + RIGHT(STORE\_NAME,3) ) AS "FUNNY STRING" FROM STORE;

FUNNY STRING

------------

Acc\*\*\*ion

Dat\*\*\*ner

Tup\*\*\*rge

Att\*\*\*ley

Pri\*\*\*int

(5 row(s) affected)

1. Find the average year-to-date sales for all stores.

SELECT AVG(STORE\_YTD\_SALES) AS AVERAGE\_YTD\_SALES FROM STORE;

AVERAGE\_YTD\_SALES

---------------------------------------

1457378.600000

(1 row(s) affected)

1. Each store is going to spend a percentage of their sales on the education of their employees. The percentage is determined by store code times 1%. For example, Access junction will spend 1% while Attribute Alley will spend 4%. What is the total money contributed towards education among all stores?

SELECT SUM((STORE\_CODE\*STORE\_YTD\_SALES) /100) AS TOTAL\_MONEY FROM STORE;\

TOTAL\_MONEY

---------------------------------------

252365.450000

(1 row(s) affected)

1. Find the region code, the total and the average year-to-date sales for stores in each region.

SELECT REGION\_CODE, SUM(STORE\_YTD\_SALES) AS TOTAL, AVG(STORE\_YTD\_SALES) AS AVERAGE FROM STORE GROUP BY REGION\_CODE;

REGION\_CODE TOTAL AVERAGE

----------- --------------------------------------- ---------------------------------------

1 3916881 1958440.500000

2 3370012 1123337.333333

(2 row(s) affected)

1. List the store code and store name of the store with the highest year-to-date sales.

SELECT STORE\_CODE, STORE\_NAME FROM STORE WHERE STORE\_YTD\_SALES = ( SELECT MAX(STORE\_YTD\_SALES) FROM STORE);

STORE\_CODE STORE\_NAME

----------- --------------------

5 Primary Key Point

(1 row(s) affected)

1. Assuming that we don’t the year-to-date sales of two stores being the same, list the region code, store code and store name of the store with the highest year-to-date sales in each region.

SELECT REGION\_CODE, STORE\_CODE, STORE\_NAME FROM STORE WHERE STORE\_YTD\_SALES IN

(SELECT MAX(STORE\_YTD\_SALES) FROM STORE GROUP BY REGION\_CODE) ORDER BY 1;

REGION\_CODE STORE\_CODE STORE\_NAME

----------- ----------- --------------------

1 5 Primary Key Point

2 2 Database Corner

(2 row(s) affected)

1. How many employees work in each store? List as the following sample:

store\_code #employees

----------- -----------

1 3

...

SELECT STORE\_CODE, COUNT(EMP\_CODE) AS #EMPLOYEES FROM EMPLOYEE GROUP BY STORE\_CODE;

STORE\_CODE #EMPLOYEES

----------- -----------

1 3

2 3

3 6

4 6

5 3

(5 row(s) affected)

1. Make a list of all stores and regions, as in the following sample:

Code Description

----------- --------------------

1 Access Junction

1 East

2 Database Corner

...

SELECT STORE\_CODE AS CODE,STORE\_NAME AS DESCRIPTION FROM STORE

UNION

SELECT REGION\_CODE, REGION\_DESCRIPT FROM REGION;

CODE DESCRIPTION

----------- --------------------

1 Access Junction

1 East

2 Database Corner

2 West

3 Tuple Charge

4 Attribute Alley

5 Primary Key Point

(7 row(s) affected)

1. Find the intersection of employee codes, store codes and region codes.

SELECT EMP\_CODE FROM EMPLOYEE

INTERSECT

SELECT STORE\_CODE FROM STORE

INTERSECT

SELECT REGION\_CODE FROM REGION

EMP\_CODE

-----------

1

2

(2 row(s) affected)

1. List the (cross) product of STORE table and the REGION table. How many columns does the result have?

SELECT \* FROM STORE CROSS JOIN REGION;

STORE\_CODE STORE\_NAME STORE\_YTD\_SALES REGION\_CODE EMP\_CODE REGION\_CODE REGION\_DESCRIPT

----------- -------------------- --------------------------------------- ----------- ----------- ----------- -------------

1 Access Junction 1003456 2 8 1 East

2 Database Corner 1421987 2 12 1 East

3 Tuple Charge 986783 1 7 1 East

4 Attribute Alley 944569 2 3 1 East

5 Primary Key Point 2930098 1 15 1 East

1 Access Junction 1003456 2 8 2 West

2 Database Corner 1421987 2 12 2 West

3 Tuple Charge 986783 1 7 2 West

4 Attribute Alley 944569 2 3 2 West

5 Primary Key Point 2930098 1 15 2 West

(10 row(s) affected)

The resulting product table has **7 columns**.

1. List the employee codes that are not store codes.

SELECT EMP\_CODE FROM EMPLOYEE WHERE EMP\_CODE NOT IN (SELECT STORE\_CODE FROM EMPLOYEE);

EMP\_CODE

-----------

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

(16 row(s) affected)