UNIVERSITY OF TECHNOLOGY, JAMAICA

Lab Manual

FACULTY: Engineering & Computing (FENC)

SCHOOL/DEPT: School of Computing & Information Technology

COURSE OF STUDY: Bachelor of Science in Computing

YEAR: Four (4)

MODULE TITLE: **Database Administration**

Introduction

This week you will be introduced to database **Procedures**, **views**, **Triggers,** and **Database Auditing**. We will be using Oracle 11G.

Students should store all SQL Queries in a txt file and store it on their G Drive.

Topic: Procedures, views, Trigger, Auditing

Objectives:

1. Creating Procedures and Views
2. Creating Triggers for auditing
3. Database Auditing

Please Note

Each Student should create a lab report folder on your G Drive. Week by week you will write a lab report of the activities you have done in the lab. At the end of the semester you will present the lab report to your lab tutor. Weekly activities account for 10% of your grade.

Lab Work should be done in groups of two. This allows students to try the various techniques on a live user. Students will observe the environment and makes notes in the Lab Report.

Creating Views for Security

Views can be used to in security by restricting results horizontally or vertically. With vertical restrictions we are are simply limiting the columns the user can interact with. When using horizontal restrictions we are simply limiting the rows the user can interact with by using a "where" clause.

Syntax:

CREATE VIEW <view_name>

AS

<SELECT STATEMENT>

Example:

CREATE VIEW view_emp

AS

SELECT empid FROM emp where salary > 5000000;

Exercise 1
(Please note that SQL statements and observations must be written for each exercise. Observation Questions are placed there to guide you and should be an addition to what you observe)

1. Ensure that StudentB (the user you created from the last lab) has no viewing privileges on the StudentA schema.

   Observation: What privileges did you revoke? How did you confirm that these privileges were removed?

   Note: You will need the SALES table in the StudentA schema

2. Use studentA to create a view "view_reorder_<yourid>" on your sales table demonstrating horizontal restriction. [**Products with a quantity under 5**]

   Observation: Can the user create the view? What privileges do you need?

3. Let's pretend the StudentB is an ordering officer. Allow your lab partner to log in as SutdentB and try to access the full sales table.

   Observation: Can you confirm if StudentB has any privileges on the table? Which tables did you query and what it is the purpose of that table?

4. Use **StudentA** to grant the appropriate privileges to **StudentB** to access the view.

   Observation: Which system table confirms that this was completed? Give a sample record form the table, explaining what each field and value means

5. Use StudentB to access to view.

6. Use StudentB to access the full sales table.

   Observation: What do you notice about accessing the information?

7. Use StudentA to create a view "vertical_view_<yourid>" that demonstrates vertical restrictions (Your Choice for the view).

   Observation: Which system table shows that the view was created? Which field in the system table lets you know that StudentA is the creator of the view?

8. Allow your lab partner to use StudentB to access the view.

   Observation: What did you observe on your attempt?

9. Use StudentA to grant StudentB the insert privilege on the view "vertical_view".

   Observation: What system table shows that StudentB now has insert privileges?

10. Use StudentB to insert a record into the sales table.
    Observation: Can the user insert into the sales table (not the view)? Explain why this is so.

11. Use StudentB to insert a record with all the fields of the sales table into "view_reorder_<your id number>".

    Observation: Can the user insert? Explain your Observation.

12. Use StudentB to insert a record with all the fields of the view into "vertical_view".

    Observation: Can the user insert? Explain your Observation.

13. Use StudentA and StudentB (Partner 1 – StudentA, Partner 2- StudentB) to update a column in the Sales Table.

   Observation: Can the users update? If one couldn't update what needed to be done?

14. Use StudentA and StudentB (Partner 1 – StudentA, Partner 2- StudentB) to update a column in "vertical_view".

   Observation: Can the users update? If one couldn't update what needed to be done?

Creating Stored Procedures for Security

Store Procedures can also be used for security. We can fire SQL with stored procedures to interact with tables that users don't have direct access to.

Example:

CREATE TABLE T2 (

a INTEGER,

b CHAR(10)

);

CREATE PROCEDURE addtuple1(i IN NUMBER) AS

BEGIN

   INSERT INTO T2 VALUES(i, 'xxx');

END addtuple1;

/

NOTE: the **"/"** is needed at the end.

Exercise 2
(Please note that SQL statements and observations must be written for each exercise. Observation Questions are placed there to guide you and should be an addition to what you observe)

1. Create a stored procedure to insert data into the sales table with StudentA.

   Observation: What privileges are needed? After you have corrected the problem, provide the system table that confirms the privilege has been granted and a sample record showing the information.

2. Try inserting with **StudentB** into the sales table

   Observation: Can they insert? If they can, revoke any privileges for insert and try again.

3. Use Student B to execute the stored procedure.

   Observation: Can the user execute the stored procedure? Which system table shows whether they have privileges or not to use the stored procedure?

4. Grant the appropriate privilege to student to execute the procedure with StudentA.

   Observation: Which tables confirm this?

5. Use StudentB to execute the procedure.

   Observation: Was the sales table updated?

6. Use StudentA a to create a stored procedure "**sp_time_window_<your_id>**" that only does inserts in the month of March. The stored procedure should accept values for the table but use the system date to determine if the records should be inserted.

7. Use StudentB to execute the procedure.

   Observation: What did you notice? Did the user have sufficient privileges ? How did you fix this?

8. Create a second procedure "**sp_time_window_2_<your_id>**" to insert only in the month of February

9. Try the procedure again with StudentB.

   Observation: Did the user have sufficient privileges? How did you fix this? What did you notice when the insert was attempted?