Lab 06 – Transactions and Security

Objectives:

The purpose of this lab is to introduce the student to both transactions and security. In the real-world, databases tasks often involve multiple steps and if any step in the middle fails, the procedure is a failure. This lap walks the student through a couple transactions and lets them learn how various steps have varying consequences that they need to be aware of.

By the end of this lab, the student will be able to:

- Describe the steps of a transaction, how a transaction begins and ends and walk through live scenarios of a variety of transactions
- Understand and act appropriately on what needs to be done in the case of transaction failure
- Grant and revoke permissions to and from other users and public users from the database

Submission:

Your submission will be a single WORD file with the query and result screenshot from Oracle SQL developer

Make sure every SQL statement terminates with a semicolon.

- You will use following data to complete the given tasks:
- ***SET TRANSACTION READ WRITE*** starts a new transaction.

- ***COMMIT*** commits the current transaction, making its changes permanent.
- ***SAVEPOINT <name>*** sets a pointer to a location that can be rolled back to.
- ***ROLLBACK*** rolls back the current transaction, canceling its changes.
- ***SET autocommit*** disables or enables the default ***autocommit*** mode for the current session.

Tasks:

It is very important that these tasks/questions be performed in the order presented here for maximum learning.

PART A - Transactions

1. List the 4 ways that we know that a transaction can be started

2. Using SQL, create an **empty** table, that is the same as the customers table, and name it ***newCustomers***.

3. Execute the following commands.

```
SET AUTCOMMIT OFF;
SET TRANSACTION READ WRITE;
```

4. Write an INSERT statement to populate the **newCustomers** table with the rows of the sample data. (Write a single INSERT statement to insert all the rows)

customerNumber	contactLastName	contactFirstName	phone	addressLine1	city	country
100	Patel	Ralph	2233355555	10 SenecaWay	Paris	France
101	Denis	Betty	3344455555	110 SenecaWay	Chicago	USA
102	Biri	Ben	44555445544	13000 SenecaWay	Toronto	Canada
103	Newman	Chad	66777332233	12 SenecaWay	Mexico city	Mexico
104	Ropeburn	Audrey	7788811212	15000 SenecaWay	Havana	Cuba

5. Create a query that shows all the inserted rows from the newCustomers table. How many rows are selected?

6. Execute the rollback command. Display all rows and columns from the newCustomers table. How many rows are selected?

7. Repeat Task 4. Make the insertion permanent to the table newCustomers. Display all rows and columns from the newCustomers table. How many rows are selected?

8. Write an update statement to update the value of column addressLine1 to ‘unknown’ for all the customers in the newCustomers table.

9. Make your changes permanent.

10. Execute the rollback command.

    a. Display all customers from the newCustomers table whose address is ‘unknown’. How many rows are still updated?

    b. Was the rollback command effective?

    c. What was the difference between the result of the rollback execution from Question 6 and the result of the rollback execution of this task?

11. Begin a new transaction and then create a statement to delete the customers from the newCustomers table

12. Create a VIEW, called **vwNewCusts**, that queries all the records in the newCustomers table sorted by last name and then by first name.

13. Perform a rollback to undo the deletion of the customers

    a. How many customers are now in the newCustomers table?

    b. Was the rollback effective and why?

14. Begin a new transaction and rerun the data insertion from Question 4 (copy the code down to Question 14 and run it)

15. Set a Savepoint, called ***insertion***, after inserting the data

16. Rerun the update statement from Question 8 and run a query to view the data (copy the code down and run it again)

17. Rollback the transaction to the Savepoint created in Question 15 above and run a query to view the data.
    What does the data look like (i.e. describe what happened?

18. Use the basic for of the rollback statement and again view the data. Describe what the results look like and what happened.

Part B - Permissions

19. Write a statement that denies all access to the newCustomers table for all public users

20. Write a statement that allows a classmate (use their database login) read only access to the newCustomers table.

21. Write a statement that allows the same classmate to modify (insert, update and delete) the data of the newCustomers table.

22. Write a statement the denies all access to the newCustomers table for the same classmate.

Part C – Clean up

23. Write statements to permanently remove the view and table created for this lab