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

Answer:

1. The user has established a new connection to the server and has a blank sheet ready to go, starts a new transaction

2. The user uses the BEGIN statement in Oracle SQL, this will start a new transaction

3. The user executes a COMMIT statement, the current transaction is commited, and a new transaction starts.

4. The user executes ANY DDL statement. This automatically triggers an auto-commit of the current transaction and starts a new transaction.

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

Answer:

Graphical user interface, text, application

Description automatically generated

1. Execute the following commands.

SET AUTCOMMIT OFF;

SET TRANSACTION READ WRITE;

Graphical user interface, text, application, email

Description automatically generated

1. 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)

Table

Description automatically generated with medium confidence

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

One row is selected

Graphical user interface, text, application, Word

Description automatically generated

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

Graphical user interface, application

Description automatically generated

1. 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?

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

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

Graphical user interface, text, application, email

Description automatically generated

1. Make your changes permanent.

Graphical user interface, application, Word

Description automatically generated

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

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, text, application

Description automatically generated

b. Was the rollback command effective?

No, the rollback command was not 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?

The main difference between the result of the rollback execution from Q-6 and the result of the rollback it only affects the current transaction, while the commit statement that we used before it already started a new transaction.

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

Graphical user interface, text, application, email

Description automatically generated

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

Graphical user interface, text, application, email

Description automatically generated

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

Graphical user interface, application, Word

Description automatically generated

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

No table is updated

Graphical user interface, text, application

Description automatically generated

1. Was the rollback effective and why?

No, the rollback was not effective because the transaction was already committed.

1. Set a Savepoint, called insertion, after inserting the data

Graphical user interface, text, email

Description automatically generated

A picture containing text

Description automatically generated

1. Set a Savepoint, called insertion, after inserting the data

Graphical user interface, text, application, email

Description automatically generated

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

Graphical user interface, text, application, email

Description automatically generated

1. 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?

Answer: The data looks similarly like the data we inserted initially.

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated with low confidence

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

Here the new transaction is run by the rollback statement

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

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

Graphical user interface, text, application

Description automatically generated

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

Graphical user interface, text, application, email

Description automatically generated

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

Graphical user interface, text, application, email

Description automatically generated

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

Graphical user interface, text, application, email

Description automatically generated

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

Graphical user interface, text, application, email

Description automatically generated