# DBMS/SQL

## Lesson 07 Transaction Control Language

Capgemini

# Lesson Objectives

To understand the following topics:

- Introduction to Transactions
- Statement execution  and Transaction control
- Commit Transactions
- Rollback transactions

# Defining Transaction

A “transaction” is a logical unit of work that contains one or more SQL statements.

- “Transaction” is an atomic unit.
- The effects of all the SQL statements in a transaction can be either:
  - all committed (applied to the database), or
  - all rolled back (undone from the database)
- A “transaction” begins with the first executable SQL statement.

# Defining Transaction

A “transaction” ends when any of the following occurs:

- A user issues a COMMIT or ROLLBACK statement without a SAVEPOINT clause.
- A user runs a DDL statement such as CREATE, DROP, RENAME, or ALTER.
- If the current transaction contains any DML statements, Oracle first commits the transaction, and then runs and commits the DDL statement as a new, single statement transaction.
- A user disconnects from Oracle. The current transaction is committed.
- A user process terminates abnormally. The current transaction is rolled back.

# Statement Execution and Transaction Control

A “SQL statement” that runs successfully is different from a committed transaction.

However, until the “transaction” that contains the “statement” is committed, the “transaction” can be rolled back. As a result, all the changes in the statement can be undone.

Hence we can say, “a statement, rather than a transaction, runs successfully”.

# Commit Transactions

Committing a transaction means making “permanent” all the changes performed by the SQL statements within the transaction.

- This can be done either explicitly or implicitly.

Syntax:

```
COMMIT [WORK];
```

# Commit Transactions

COMMIT types:

- Implicit: Database issues an implicit COMMIT before and after any data definition language (DDL) statement
- Explicit

Example of COMMIT command:

```
DELETE FROM student_master
WHERE student_name = 'Amit';
COMMIT ;
```

# Rollback Transactions

Rolling back a transaction means “undoing changes” to data that have been performed by SQL statements within an “uncommitted transaction”.

- Oracle uses “undo tablespaces” (or rollback segments) to store old values.
- Oracle also uses the “redo log” that contains a record of changes.

Oracle lets you roll back an entire “uncommitted transaction”.

- Alternatively, you can roll back the trailing portion of an “uncommitted transaction” to a marker called a “savepoint”.

# Summary

In this lesson, you have learnt:

- Transactions
  - Statement execution
  - Transaction control
  - Commit Transactions
  - Rollback transactions

# Review – Questions

Question 1 : ____ is a logical unit of work.

Question 2: A transaction is committed when the user issues a DDL statement.

- True/False

Question 3: A transaction is rolled back when ____.

- Option 1: rollback statement is issued
- Option 2: the user session is abruptly terminated
- Option 3: an error occurs in DML statement
- Option 4: none of the above

Question 4:In a transaction, DDL statement after DML statement  commits the changes done by DML.

- True/False

# Review – Questions

- **Question 1 : ____ is a logical unit of work.**
- **Question 2: A transaction is committed when the user issues a DDL statement.**
  - True/False
- **Question 3: A transaction is rolled back when ____.**
  - Option 1: rollback statement is issued
  - Option 2: the user session is abruptly terminated
  - Option 3: an error occurs in DML statement
  - Option 4: none of the above


Question 4:In a transaction, DDL statement after DML statement commits the changes done by DML.

- True/False