21.04.2020

Computer Science 2nd Year
Subject - 'DBMS'

PL/SQL Transactions

A database transaction is an atomic unit of work that may consists of one or more related SQL statements. It is called atomic because the database modifications brought about by the SQL statement that constitute a transaction can collectively be either committed, i.e made permanent to the database or rolled back (undone) from the database.

A successfully executed SQL statement & a committed transaction are not same. Even if an SQL statement is executed successfully unless the transaction containing the statement is committed, it can be rolled back, and all changes made by the statement(s) can be undone.

Starting and Ending a transaction

A transaction has a beginning and an end. A transaction start when one of the following event take place.

⇒ The first SQL statement is performed after

Connecting to the database.

⇒ At each new SQL Statement Issued after a transaction is completed

* A transaction ends when one of the following event take place.

⇒ A COMMIT or a ROLLBACK Statement is Issued.

⇒ A DDL Statement, such as CREATE table Statement is Issued; because in that case a COMMIT is automatically performed.

⇒ A DCL Statement, such as a GRANT Statement is Issued; because in that case a COMMIT is automatically performed.

⇒ SQL*PLUS terminates abnormally a ROLLBACK is automatically performed.

⇒ A DML Statement fails; in that case a ROLLBACK is automatically performed for undoing that DML Statement.

Committing a transaction

* A transaction is made permanent by Issuing the SQL Command COMMIT. The general Syntax for the COMMIT Command is -

COMMIT;

Rolling Back transactions

changes made to the database without COMMIT could be undone using the ROLLBACK command is -

The general syntax for the ROLLBACK command is

```
ROLLBACK [TO SAVEPOINT < savepoint-name >];
```

when a transaction is aborted due to some unpredictions situations, like system failure, the entire transaction since a commit is automatically rolled back, If you are not using savepoint, then simply use the following statement- to rollback all the changes -

```
ROLLBACK;
```

Savepoints

Savepoints are sort of markers that help in splitting a long transaction into smaller unit by setting some checkpoints. By setting saving savepoints within a long transaction, you can roll Back to a checkpoint if required this is done by issuing the SAVEPOINT Command

The general syntax for the SAVEPOINT Command.

SAVEPOINT < Savepoint_name > ;

Sharetha Vaish 21/04/20..

Scanned with CamScanner