

Université Ibn khaldoun Tiaret  
Faculté des Mathématiques et Informatique  
Département Informatique  
Master Génie Logiciel



# La Journalisation des transactions de base de données

Compte rendu TP1 DBA & DM

## Les membres du groupe :

ZEGAI Houari

NADJEM Nour El Imane

MADENE Malika

## Enseignant :

Mr Ouared



## 1. Objective du TP:

Dans ce TP nous avons vu les mises à jours effectué à des tables d'une BDD en décrivant la notion des transactions, les exécuter et observer les différents résultats trouvés. Par la suite on va essayer de présenter le concept d'historique des modifications faites.

## 2. Présentation du problème :

D'abord dans la première partie de TP, nous avons exécuté un bloc de transactions, puis voir quelles sont les données validées après cet ensemble d'opération. Ensuite la deuxième partie est consacrée pour les Triggers et leur utilisation afin d'enregistrer l'historique des modifications d'une table.

## 3. Résultats et observations:

### ➤ *Partie 1 :*

Dans cette partie, et en premier lieu nous avons créés des points d'enregistrements 'SAVEPOINTS' puis exécuter d'autres instructions et après faire un 'ROLLBACK' à un des points d'enregistrements.

Deuxièmes dans le 1<sup>er</sup> bloc on remarque qu'il ne contient pas la commande 'COMMIT' cependant dans le 2eme on trouve cette instruction après finir toutes les transactions faites.

**Résultats :** Nous avons observés que toutes les modifications faites après le 'SAVEPOINT' et avant le 'ROLLBACK' ont été annulés. Le 2eme résultats qu'on peut conclure de cette partie est l'importance de la commande 'COMMIT', car dans la 1ere séquence et sans écrire cette instruction rien n'est fait.

### ➤ *Partie 2 :*

Comme nous avons déjà mentionné, cette partie est destiné pour comprendre comment utiliser des Triggers pour créer l'historique pour un ensemble des modifications faites dans une table.

Nous avons créé un Trigger qui archive les mises à jour effectué à une table 'Compte' dans une table 'Historique' en enregistrant quelque attributs (opération, date, ancienne valeur, nouvelle valeur ...etc.).

**Résultats :** Après chaque modification faite dans la table 'Compte', le Trigger se lance et les données sont sauvegardés dans la table 'Historique'.

#### 4. Interprétation des résultats :

- La commande 'COMMIT' est importante pour l'exécution de la transaction si on a changé la valeur par défaut (1 == True) de la commande AUTOCOMMIT.
- La commande 'ROLLBACK' d'un 'SAVEPOINT' précis annule toutes les opérations qui sont entre le 'ROLLBACK' et le SAVEPOINT appelé.
- On peut faire de nombreuses choses avec un trigger, parmi eux garder une trace des actions effectuées sur la base de données.

#### 5. Conclusion :

Les transactions sont une fonctionnalité absolument indispensable, permettant de sécuriser une application utilisant une base de données. Sans transactions, certaines opérations risqueraient d'être à moitié réalisées, elles permettent de regrouper des requêtes dans des blocs, et de faire en sorte que tout le bloc soit exécuté en une seule fois, cela afin de préserver l'intégrité des données de la base.