

Module d'enseignement	Administration des Bases de Données	Code E.E	ECUEF412
Niveau	Licence Génie Logiciel et Système d'Information	Classe	LGLSI -2
TP N°4: Gestion des transactions			
Boîte à outils logiciels	Oracle XE 11g R2		
Elaboré par	Manel BOUTALEB		

Rappel:

- Une transaction est :
 - Une séquence d'instructions interrogeant ou modifiant la BD
 - L'exécution peut être validée ou totalement annulée : Garantir un état cohérent de la BD quel que soit le résultat de l'exécution de la transaction
- Accès concurrent
 - Il y a un accès concurrent lorsque plusieurs utilisateurs (transactions) accèdent en même temps à la même donnée dans une base de données.
- Gestion des accès concurrents (contrôle de concurrence)
 - S'assurer que l'exécution simultanée des transactions produit le même résultat que leur exécution séquentielle (l'une puis l'autre)
 - Garantir un état cohérent de la BD après un accès concurrent

Syntaxe :

Validation explicite d'une transaction	COMMIT ;
Annulation explicite d'une transaction	ROLLBACK ;
Création d'un point de sauvegarde	SAVEPOINT nomPointSauvegarde ;
Annulation vers un point de sauvegarde	ROLLBACK TO nomPointSauvegarde ;

Exercice1 : atomicité d'une transaction courante

1. Sous le super admin SYS, créer une table etudiants et y insérer quelques lignes, consulter le contenu de la table, modifier une ligne, supprimer une autre et enfin annuler les mises à jour venant d'être effectuées avec la commande ROLLBACK.
Voir à nouveau le contenu de la table et sa structure. Pourquoi la table est-elle vide ?
2. Insérer à nouveau quelques lignes dans Etudiants, les modifier et les détruire partiellement, puis valider ces mises à jour avec la commande COMMIT, puis déclencher un ROLLBACK.
Que s'est-il passé ?
Maintenant détruire les données de la table et valider.
3. Insérer à nouveau dans Etudiants quelques lignes et clore la transaction par un EXIT ou un QUIT.
Que s'est-il passé ?
4. Insérer à nouveau deux ou trois lignes dans la table Etudiants et fermer brutalement Sql Command Line, puis rentrer à nouveau dans votre compte. Les données saisies ont-elles été préservées ?
5. Insérer à nouveau quelques lignes dans la table Etudiants, puis adjoindre une nouvelle colonne à sa table (ou plus généralement émettre n'importe quelle commande de description des données) et essayer d'annuler les dernières insertions.
Pourquoi est-ce impossible ?

Exercice 2 : Commit, Rollback, Savepoint

Nous allons travailler sur la table members déjà crée dans le TP précédent.

1. Ouvrir la fenêtre Run sql Command Line
2. Se connecter sous le schéma où se trouve la table members (dans mon cas : USER_COMPTA)
3. Afficher toute la table members ainsi que le nombre des lignes
4. Ouvrir une autre fenêtre de run sql command line et se connecter avec le super admin sys
5. Afficher maintenant la table members
6. On retourne à la première fenêtre, insérer un nouveau tuple dans la table members
7. Réafficher la table members depuis le super admin SYS (dans la deuxième fenêtre). Que remarquez-vous ?
8. Faire le nécessaire pour valider la transaction (l'insertion) et voir si la modification est apparue depuis le super admin ou pas
9. Dans la première fenêtre, insérer de nouveau une autre ligne dans members
10. Afficher le nombre des lignes de la table
11. Faisons un ROLLBACK maintenant. Que remarquez-vous ?
12. Essayer cette transaction dans la première fenêtre :

SQL> delete from members;

SQL> select count(*) from members;

13. Maintenant dans la 2eme fenêtre, afficher le nombre des lignes de la table members. Que remarquez-vous ?
14. Annuler la transaction de suppression dans la première fenêtre
15. Insérer une nouvelle ligne dans la table members et se faire déconnecter d'une façon normale à l'aide de la commande exit.

Dans la deuxième fenêtre afficher le nombre des lignes de la table members. Que remarquez-vous ?

16. Dans la première fenêtre, supprimer toutes les lignes de la table members
17. Fermer la fenêtre d'une façon anormale (cliquer sur la croix de la fermeture). Maintenant, se reconnecter de nouveau avec l'utilisateur user_compta, et réafficher la table members. Que remarquez-vous ?
18. Pour l'utilisateur user_compta, changer le first_name des members_ID 3 et 2 en 'manel' et 'mohamed'.
19. Créer maintenant un savepoint qui porte le nom 'updates'. N'importe quelle transaction exécutée avant ce savepoint, elle va être protégée par ce savepoint updates. On ne peut pas faire un rollback
20. Insérer une nouvelle ligne et créer un savepoint INSERTIONS. Afficher la table
21. Faire un rollback vers insertions. Afficher la table. Que remarquez-vous ?
22. Faire un rollback vers updates. Que remarquez-vous ?
23. Valider la transaction de la mise à jour. Maintenant passons à la deuxième fenêtre du schéma sys. Afficher la table members

Exercice 3 : transactions concurrentes

1. Se connecter sous SYS
 2. Supprimer la table Etudiants (déjà créée lors de l'exercice 1), puis la recréer avec un NOM VARCHAR2(16) et ajouter quelques lignes. Ensuite, se connecter à son compte à partir d'une autre instance et consulter à travers cette nouvelle fenêtre le contenu du compte. Que voyez-vous ?
 3. Insérer une ligne dans la deuxième fenêtre. Afficher etudiants dans les deux tables. Que voit-on ?
 4. Créer dans l'une des deux fenêtres ouvertes une nouvelle table UE : (NomUE VARCHAR2(30) , Nbetudiants INTEGER) et y insérer par chacune des fenêtres quelques lignes. Que voit-on de la table Etudiants ?
 5. Détruire la table UE dans une des fenêtres. Que se passe-t-il ? Consulter le contenu de la table UE depuis l'autre fenêtre. Que contient-elle désormais ? Comment détruire UE ?
 6. Adjoindre une clé primaire à Etudiants. Insérer une même ligne dans la première fenêtre puis dans la deuxième. Que se passe-t-il ? Émettre un ROLLBACK dans la première fenêtre. Que devient le blocage ?
8. En conclusion, comment sont liés comptes, sessions et transactions ?