

M106 – Transaction Control Language (TCL) et ACID

F. Mauron

EPAI

07 mai 2024

- 1 Transactions
- 2 Principes ACID
- 3 Langage de Contrôle des Transactions (TCL)
- 4 Exemple

- 1 Transactions
- 2 Principes ACID
- 3 Langage de Contrôle des Transactions (TCL)
- 4 Exemple

Les transactions sont des séquences d'opérations effectuées sur une base de données qui sont traitées de manière logique et indivisible.

Elles sont essentielles pour maintenir l'intégrité des données, surtout lorsque plusieurs modifications sont apportées simultanément aux données.

Une transaction en SQL suit le principe ACID, qui garantit que les opérations sont effectuées de manière sûre et fiable.

- 1 Transactions
- 2 Principes ACID
- 3 Langage de Contrôle des Transactions (TCL)
- 4 Exemple

- **A**tomicité
- **C**ohérence
- **I**solation
- **D**urabilité

Atomicité

Une transaction doit être traitée comme une unité unique d'opération; elle doit être complètement exécutée ou pas du tout.

Exemple :

Lorsqu'un paiement est transféré d'un compte à un autre, le débit et le crédit doivent tous deux être exécutés pour que la transaction soit complète.

Cohérence

La transaction doit passer la base de données d'un état valide à un autre état valide, en respectant toutes les règles prédéfinies et les contraintes d'intégrité.

Exemple :

Si une transaction est exécutée pour inscrire un étudiant à un cours, elle doit également vérifier et maintenir le nombre maximum d'étudiants autorisés pour ce cours.

Isolation

Les modifications effectuées par une transaction doivent être isolées jusqu'à ce que la transaction soit complétée, pour éviter que ces modifications soient visibles par d'autres transactions parallèles.

Exemple :

Deux transactions qui mettent à jour le solde d'un même compte bancaire doivent être isolées pour empêcher l'une de voir les résultats intermédiaires de l'autre.

Durabilité

Une fois la transaction terminée et validée, ses effets sont permanents, même en cas de panne du système.

Exemple :

Après l'achat d'un article, les détails de la transaction restent stockés de façon permanente dans la base de données.

- 1 Transactions
- 2 Principes ACID
- 3 Langage de Contrôle des Transactions (TCL)
- 4 Exemple

Langage de Contrôle des Transactions (TCL)

Le Langage de Contrôle des Transactions (TCL) en SQL est essentiel pour gérer les transactions dans les systèmes de gestion de bases de données en permettant de garantir l'intégrité et la fiabilité des données.

BEGIN TRANSACTION ou simplement **BEGIN** permet de démarrer une transaction.

COMMIT valide toutes les modifications réalisées au cours de la transaction.

ROLLBACK annule toutes les modifications non validées si des erreurs surviennent pendant la transaction.

SAVEPOINT permet de créer un point de sauvegarde au sein d'une transaction, permettant un retour partiel.

ROLLBACK TO SAVEPOINT permet d'annuler les modifications jusqu'au dernier point de sauvegarde.

RELEASE ROLLBACK permet de supprimer un point de sauvegarde.

- 1 Transactions
- 2 Principes ACID
- 3 Langage de Contrôle des Transactions (TCL)
- 4 Exemple**

Exemple

```
BEGIN;  
-- Supposons que l'ID du produit est 101 et que l'ID de l'utilisateur est 501  
UPDATE produits SET quantite = quantite - 1 WHERE produitID = 101;  
UPDATE utilisateurs SET solde = solde - 100 WHERE userID = 501;  
-- Point de contrôle pour validation partielle  
SAVEPOINT AchatEffectue;  
-- Mise à jour d'une table d'ordre  
INSERT INTO commandes(userID, produitID, dateCommande)  
    VALUES (501, 101, CURRENT_DATE);  
-- Si tout est correct  
COMMIT;  
-- En cas d'erreur après le SAVEPOINT  
-- ROLLBACK TO SAVEPOINT AchatEffectue;
```