

1. Introduction aux Procédures Stockées

Définition simple

Une **procédure stockée** (ou *stored procedure*) est un ensemble d'instructions SQL enregistrées dans une base de données. Elle peut être appelée pour exécuter des opérations de manière automatisée.

Pourquoi utiliser des procédures stockées ?

- Pour éviter de répéter les mêmes requêtes.
- Pour regrouper la logique métier côté base de données.
- Pour améliorer la sécurité (moins d'accès direct aux données).

2. Exemple Commenté

Voici une procédure stockée simple qui ajoute un produit dans une table `produits`, à condition que le prix soit positif.

```
DELIMITER //
```

```
CREATE PROCEDURE ajouter_produit(IN nom VARCHAR(100), IN prix DECIMAL(10,2))
BEGIN
    IF prix > 0 THEN
        INSERT INTO produits(nom, prix)
        VALUES(nom, prix);
    ELSE
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Prix invalide';
    END IF;
END;
//
```

```
DELIMITER ;
```

Exécution :

```
CALL ajouter_produit('Stylo', 1.50);
```

3. Explication des Instructions

- `DELIMITER //` : Change le délimiteur de commande pour pouvoir écrire un bloc complet.
- `CREATE PROCEDURE` : Crée la procédure et définit les paramètres d'entrée.

- `BEGIN ... END` : Encadre le bloc d'instructions SQL.
- `IF ... THEN` : Permet de vérifier une condition.
- `SIGNAL` : Permet de générer une erreur personnalisée.
- `CALL` : Permet d'exécuter la procédure.

4. Exercice 1 : Ajouter un client si l'email est valide

Objectif : Créer une base de données et une procédure stockée qui ajoute un client à la table `clients` uniquement si l'adresse e-mail contient un @.

Étapes à suivre :

1. Créer la base et la table :

```
CREATE DATABASE tp_clients ;  
USE tp_clients ;
```

```
CREATE TABLE clients (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nom VARCHAR(100),  
    email VARCHAR(100)  
);
```

2. Créer une procédure `ajouter_client(nom_client, email_client)` qui :

- Vérifie si l'email contient @.
- Ajoute le client si c'est valide.
- Affiche une erreur sinon.

3. Tester avec :

```
CALL ajouter_client('Marie', 'marie@email.com');  
CALL ajouter_client('Jean', 'jean.email.com'); -- Devrait chouer
```

5. Exercice 2 : Mettre à jour le prix d'un produit

Objectif : Créer une procédure stockée qui permet de modifier le prix d'un produit, mais seulement si le prix est strictement positif.

Préparation

Créez d'abord la base et la table :

```
CREATE DATABASE tp_produits;  
USE tp_produits;
```

```
CREATE TABLE produits (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nom VARCHAR(100),  
    prix DECIMAL(10,2)  
);
```

```
-- Ajout d'un produit exemple  
INSERT INTO produits(nom, prix) VALUES ('Clavier', 20.00);
```

Travail demandé

Créer une procédure appelée `modifier_prix(id_produit, nouveau_prix)` qui :

- Vérifie que `nouveau_prix > 0`.
- Met à jour le prix du produit avec l'ID correspondant.
- Affiche une erreur si le prix est invalide.

Test de la procédure :

```
CALL modifier_prix(1, 25.00); -- OK  
CALL modifier_prix(1, -5.00); -- Erreur attendue
```

6. À retenir

- Une procédure stockée permet d'automatiser des traitements SQL.
- Elle accepte des paramètres d'entrée et peut inclure des conditions.
- Elle centralise la logique métier et sécurise les opérations.