

Introduction aux Bases de Données

Ulysse COUTAUD
coutaudu@gmail.com

Modifier des données

Insérer un enregistrement

Clause INSERT INTO

```
INSERT INTO table (nom_colonne_1, nom_colonne_2, ...)
VALUES ('valeur 1', 'valeur 2', ...)
```

BDD "fabrique" :

Ajouter un nouvel employé dans le service "Production"

Ajouter un service "Production Après-Midi" dont les horaires seront 13h-20h.

Ajouter les horaires 16h-5h pour le service "Astreinte"

Ajouter un nouvel employé "Abdoul" au service "Production Après-Midi".

Modifier des enregistrements

Clause UPDATE

```
UPDATE table SET
colonne_1 = 'valeur 1',
colonne_2 = 'valeur 2', ...
WHERE conditions;
```

BDD "fabrique" :

Augmenter le salaire des employes de "Astreinte" de 30%
L'employé Tom change de service et passe à l'après midi

Supprimer des enregistrements

BDD "atelier":

employees(prenom,salaire,adresse,service) horaires(service,debut,fin)

Clause DELETE FROM

DELETE FROM table WHERE conditions;

BDD "fabrique" :

Supprimer le service de production après-midi.

Modifier le schéma

Créer une table

CREATE TABLE

```
CREATE TABLE ma_table (  
  colonne1 TEXT,  
  colonne2 INTEGER  
);
```

BDD "fabrique" :

Ajouter une table "ordres_fabrications" avec 4 colonnes: id (le numéro d'OF), employe (l'employé qui va produire l'OF), reference_piece (la référence de pièce à produire)

Supprimer une table

DROP TABLE

```
DROP TABLE ma_table;
```

BDD "fabrique" :

Supprimer la table horaire.

Modifier une table

ALTER TABLE

ALTER TABLE ma_table mon_action ;

Avec mon_action:

- ADD ma_colonne INTEGER/TEXT/...
- DROP ma_colonne

BDD "fabrique" :

Ajouter la colonne quantite (le nombre de pièces à produire) à la table ordre_fabrication.

Les transactions

Le problème

```
UPDATE accounts SET balance = balance - 100.00
WHERE name = 'Alice';
```

```
UPDATE branches SET balance = balance - 100.00
WHERE name = (SELECT branch_name
               FROM accounts
               WHERE name = 'Alice');
```

```
UPDATE accounts SET balance = balance + 100.00
WHERE name = 'Bob';
```

```
UPDATE branches SET balance = balance + 100.00
WHERE name = (SELECT branch_name
               FROM accounts
               WHERE name = 'Bob');
```

La solution

Je valide une transaction avec COMMIT

```
BEGIN;  
UPDATE accounts SET balance = balance - 100.00  
    WHERE name = 'Alice';  
-- etc etc  
COMMIT;
```

J'annule une transaction avec COMMIT

```
BEGIN;  
UPDATE accounts SET balance = balance - 100.00  
    WHERE name = 'Alice';  
-- etc etc  
ROLLBACK;
```

La solution

Le SGBDR garantit des transactions ACID:

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