

Algorithmique et SQL

Commandes de LDD et C/D à la base de données et LMD

I. Connexion et déconnexion à la base de données

A. Connexion à la base de données

Opération obligatoire qui doit intervenir avant l'exécution de la première commande SQL.

```
Exec Sql START DATABASE <identificateur> Fin Exec
```

Identificateur : représente une variable comportant les renseignements nécessaires permettant la connexion à la BDD.

Rem : La fonction booléenne **Erreur Sql()** permet de savoir si la connexion a réussi ou échouer.

B. Déconnexion de la base de données

Permet de libérer les ressources utilisées.

```
Exec Sql STOP DATABASE FinExec
```

Exemple

Connexion à une base de données Access situé localement sur le poste de travail.

```
Prog GestClient
Var
    ChaineConnexion : Chaine
Début
    ChaineConnexion := « C:\msoffice\exemples\Baseclient.mdb »
    Exec Sql
        Start Database ChaineConnexion
    Fin Exec
    Si Erreur Sql() alors
        Afficher « Connexion impossible »
    Sinon
        AppelTraitementAEffectuer()
        Exec Sql Stop database Fin Exec
    Fin si
Fin
```

II. Commande LDD (Langage de Définition de Données)

Toute commande LDD peut être utilisée sous réserve que l'utilisateur ait les droits permettant la gestion des objectifs référencés dans la commande.

A. Gestion des tables

1. Mise en place d'une table

```
Exec Sql
    CREATE TABLE table [(colonne_definition * [, colonne_definition])]
Fin Sql
```

2. Suppression d'une table

```
Exec Sql
    DROP TABLE table
```

Fin Sql

B. Gestion des index

Pour accroître la performance d'accès à une table, il est souvent nécessaire d'indexer une ou plusieurs colonnes de la table. De plus l'index est le dispositif de la base utilisé par le SGBDR pour contrôler l'unicité d'une colonne.

1. Mise en place d'un index

```
Exec Sql
    CREATE [UNIQUE] INDX index ON table (donnée [ASC/DESC], colonne [ASC/DESC])
Fin Exec
```

2. Suppression d'un index

```
Exec Sql
    DROP INDEX index
Fin Exec
```

Remarques :

- Il est conseillé d'indexer au moins toutes les colonnes (du primaire), des tables ainsi que → (clé étrangère)
- La clause unique permet d'indiquer l'absence de table de l'index. Dans la pratique cela permet la mise en place d'un index relatif à la clé primaire.
- La mise en place de certain index peut être assurée automatiquement par le SGBDR à partir de la clause **CONSTRAINT** de la commande CREATE.

C. Gestion des vues

Vue : table virtuelle utilisée essentiellement pour assurer la confidentialité d'une ou plusieurs tables.

```
Exec Sql
    CREATE VIEW vue AS requete
Fin Sql

Exec Sql
    DROP VIEW vue
Fin Sql
```

Exercice

Création d'une base de données salarié

Soit le modèle relationnel suivant :

SALAIRE (Scode, Snom, SdateEmbauche, SalaireBase, StauxCommission, Ccode#)

CATEGORIE (Ccode, clibellé)

Programme Salariés

Var

ChaineConnexion : CHAINE

Début

ChaineConnexion = « C:\Exercices\Salariés.mdb »

Exec sql

START DATABASE ChaineConnexion

Fin Exec

Si ErrorSql() Alors

```

        Afficher « Connexion impossible »
    Sinon
        MiseEnPlaceMpd()
        ExecSql STOP DATABASE Fin Exec
    FinSi
Fin

Procédure MiseEnPlaceMpd()
Début
    Exec Sql
        CREATE TABLE SALARIES
        (
            Scode                Number(5)
            Snom                  Varchar2(25)
            SdateEmbauche Varchar2(8)
            SsalaireBase          Number(8)
            StauxCommission       Number (8)
            Scoche                 Varchar2(4)
            Constraint Salarié_PK  Primary Key (Scode),
            Constraint Salarié_PK  Foreign Key (Ccode) referencesCategorie(Ccode)
        )
    Fin Exec
    AfficheMessage(1, « salaries »)

    Exec Sql
        CREATE TABLE Categorie
        (
            Ccode                Number(4)
            Clibelle              Varchar2(15)
            Constraint Categorie Primary Key(Ccode)
        )
    Fin Exec
    Affiche Message (1, « categorie »)
Fin

```

III. Commande LMD (Langage de Manipulation des Données)

Les commandes LMD (Langage de Manipulation des Données) autorisent la mise à jour des tables de la base de données sous réserve que l'utilisateur en ait le droit.

Elles permettent :

- l'ajout d'une ou plusieurs lignes dans une table
- la modification d'une ou plusieurs lignes d'une table
- la suppression d'une ou plusieurs lignes d'une table

Notion de variable Hôte

Plusieurs valeurs utilisables. Dans un programme, il peut être nécessaire d'utiliser une même commande LMD avec des valeurs distinctes. Les valeurs doivent être alors remplacées par des variables. Celles-ci sont appelées variables hôtes et doivent être déclarées dans l'algorithme. Une variable hôte est donc une variable permettant l'échange d'informations entre la BDD et le programme, permettant ainsi de paramétrer la commande LMD.

Variables utilisées = « variables hôtes d'entrées ». Précédées d'un préfixe (: en algo, @ avec SQL Server) , « **Update** », « **Insert values** », « **delete** ».

1. Ajout d'une ligne dans une table : INSERT

```

Exec Sql
    INSERT INTO table [(colonne, colonne,...)]
    VALUES (valeur, valeur,...) | requête
Fin Exec

```

« valeur » : constante ou variable hôte

« requête » : commande LID (Langage d'Interrogation des Données) dont la condition peut comporter des variables hôtes.

2. Modification d'une ligne dans une table : UPDATE

```
Exec Sql
  UPDATE table
    SET [ colonne = expression | requête ]
    [WHERE condition]
Fin Exec
```

3. Suppression d'une ligne dans un table : DELETE

```
Exec Sql
  DELETE FROM table
  [WHERE condition]
Fin Exec
```

Exercice

Soit le LMD suivant :

ETUDIANT (ecode, enom, eprenom, edatenaissance)
MATIERE (mcode, mlibellé, mcoefficient)
MOYENNE (ecode#, mcode#, moyenne)