

3 -ème LMD Mathématique appliqués
Système d'information et BDD
Année : 2024-2025

TP Séries

Exercice : Gestion d'une Chaîne de Magasins

Scénario

La société CommercioLux, une chaîne de magasins spécialisée dans la vente de produits électroniques et électroménagers, souhaite moderniser son système de gestion afin d'améliorer ses opérations et d'assurer un meilleur suivi de ses activités. Le nouveau système devra prendre en charge différents aspects de l'entreprise, notamment la gestion des clients, des produits, des commandes, des employés, et des magasins.

CommercioLux possède plusieurs magasins répartis dans différentes villes. Chaque magasin est identifié par un Numéro de Magasin unique, et dispose d'un nom, d'une adresse complète (rue, ville, code postal) et d'un numéro de téléphone. Chaque magasin a un gérant qui supervise les opérations quotidiennes et un certain nombre d'employés.

Les employés de la société sont identifiés par un Numéro d'Employé et disposent de plusieurs informations personnelles, telles que leur nom, prénom, date de naissance, adresse, numéro de téléphone, et leur poste (vendeur, caissier, gestionnaire, etc.). Chaque employé est affecté à un seul magasin, mais un magasin peut avoir plusieurs employés.

Les produits vendus par CommercioLux incluent des téléviseurs, des ordinateurs, des réfrigérateurs, des smartphones, et d'autres articles électroniques. Chaque produit est identifié par un Code Produit et dispose d'un nom, d'une catégorie (informatique, électroménager, téléphonie, etc.), d'un prix unitaire, et d'une quantité en stock. Chaque magasin dispose de son propre stock de produits, qui peut varier en fonction des ventes et des réapprovisionnements.

Les clients de CommercioLux peuvent passer des commandes en ligne ou directement en magasin. Chaque client est identifié par un Numéro de Client et doit fournir son nom, prénom, adresse complète, numéro de téléphone, et adresse e-mail. Les clients peuvent passer plusieurs commandes, mais chaque commande est associée à un seul client.

Une commande est identifiée par un Numéro de Commande et comprend des informations telles que la date de commande, le montant total, et la liste des produits commandés avec leurs quantités respectives. Une commande peut inclure plusieurs produits, et un même produit peut apparaître dans plusieurs commandes. Chaque commande est traitée par un employé spécifique dans un magasin donné.

Enfin, CommercioLux organise régulièrement des promotions sur certains produits pour attirer plus de clients. Chaque promotion est identifiée par un Code de Promotion et inclut des informations telles que la réduction en pourcentage, la durée de validité, et les produits concernés.

Consignes

1. Modèle Conceptuel de Données (MCD) :

Identifiez les entités, leurs attributs, et les relations entre elles à partir du texte.

Construisez un MCD clair et détaillé pour représenter les données.

2. Modèle Logique de Données (MLD) :

Transformez le MCD en un MLD en définissant les tables, leurs attributs, et les clés primaires et étrangères.

3. Modèle Physique de Données (MPD) :

Concevez le MPD en spécifiant les types de données pour chaque colonne et en ajoutant des contraintes (unicité, intégrité référentielle, etc.).

4. Implémentation dans Access :

Créez la base de données dans Microsoft Access en utilisant deux méthodes :

a. Méthode manuelle : définissez les tables et relations directement dans Access.

b. Méthode SQL : utilisez les commandes SQL pour créer les tables et insérer des données.

SQL :

Brève Explication des Commandes SQL

1. CREATE

La commande CREATE est utilisée pour créer de nouveaux objets de base de données, comme des tables, des vues ou des bases de données.

```
CREATE TABLE nom_table (  
    colonne1 type_donnée contraintes,  
    colonne2 type_donnée contraintes,  
    ...  
);
```

Exemple :

```
CREATE TABLE Employe ( ID_Employe INT PRIMARY KEY, Nom VARCHAR(50), Departement  
VARCHAR(30), Salaire DECIMAL(10, 2));
```

2. INSERT

La commande INSERT permet d'ajouter de nouvelles lignes de données dans une table.

```
INSERT INTO nom_table (colonne1, colonne2, ...)  
VALUES (valeur1, valeur2, ...);
```

Exemple :

```
INSERT INTO Employe (ID_Employe, Nom, Departement, Salaire)
VALUES (1, 'Alice', 'RH', 55000.00);
```

3. UPDATE

La commande UPDATE permet de modifier les données existantes dans une table.

```
UPDATE nom_table
SET colonne1 = valeur1, colonne2 = valeur2, ...
WHERE condition;
```

Exemple :

```
UPDATE Employe
SET Salaire = 60000.00
WHERE ID_Employe = 1;
```

4. DELETE

La commande DELETE est utilisée pour supprimer des lignes dans une table.

```
DELETE FROM nom_table
WHERE condition;
```

Exemple :

```
DELETE FROM Employe
WHERE ID_Employe = 1;
```

Note : Si la clause WHERE est omise, toutes les lignes de la table seront supprimées :

```
DELETE FROM Employe;
```

5. SELECT

La commande SELECT est utilisée pour récupérer des données d'une table. Vous pouvez filtrer, trier ou agréger les données selon vos besoins.

```
SELECT colonne1, colonne2, ...
FROM nom_table
WHERE condition;
```

Exemple :

```
SELECT Nom, Salaire
FROM Employe
WHERE Departement = 'RH';
```

Pour récupérer toutes les colonnes, utilisez * :

```
SELECT * FROM Employe;
```

Points Clés

1. Contraintes : Les commandes comme CREATE incluent souvent des contraintes telles que PRIMARY KEY, FOREIGN KEY, NOT NULL, etc., pour assurer l'intégrité des données.
2. Conditions : Les commandes comme UPDATE, DELETE et SELECT utilisent la clause WHERE pour cibler des lignes spécifiques.
3. Intégrité des données : Soyez vigilant avec les commandes DELETE et UPDATE pour éviter des modifications ou suppressions accidentelles.