

## TP N°3

### Enoncé 1 : Gestion de Magasin

1. Table **produit** : Contient les informations sur les produits.

- `reference` (INT, Primary Key)
- `nom` (VARCHAR(100))
- `prix` (DECIMAL(10, 2))
- `quantiteStock` (INT)
- `categorie\_id` (INT, Foreign Key vers `categorie`)

2. Table **categorie** : Liste les catégories de produits.

- `id` (INT, Primary Key)
- `nom` (VARCHAR(50))

3. Table **commande** : Stocke les commandes passées par les clients.

- `id` (INT, Primary Key)
- `date\_commande` (DATE)
- `client\_nom` (VARCHAR(100))
- `client\_email` (VARCHAR(100))

4. Table **ligne\_commande** : Contient les détails des commandes.

- `commande\_id` (INT, Foreign Key vers `commande`)
- `produit\_id` (INT, Foreign Key vers `produit`)
- `quantite` (INT)

### Partie 1 : Création de Tables

1. Écrivez les instructions SQL pour créer les tables ci-dessus avec les clés primaires et étrangères appropriées.

### Partie 2 : Questions sur les Procédures

1. Créer une procédure : Ajouter un produit
- Créez une procédure appelée `ajouter\_produit` qui permet d'ajouter un nouveau produit à la table `produit`.

### Paramètres :

- `nom` (Nom du produit)
- `prix` (Prix du produit)
- `quantiteStock` (Quantité en stock)
- `categorie\_id` (Catégorie associée)

### 2. Créer une procédure : Mettre à jour le stock

Créez une procédure appelée `mettre\_a\_jour\_stock` qui prend en paramètre :

- `produit\_id` (Identifiant du produit)
- `quantite\_ajoutee` (Quantité à ajouter ou retirer).
- Cette procédure met à jour la quantité en stock pour le produit donné.

### 3. Créer une procédure : Afficher les commandes d'un client

Créez une procédure appelée `afficher\_commandes\_client` qui prend en paramètre :

- `client\_email` (Email du client).
- Elle affiche toutes les commandes passées par ce client (avec la date et les détails des commandes).

## Partie 3 : Questions sur les Fonctions

### 1. Créer une fonction : Calculer le total d'une commande

Créez une fonction appelée `calculer\_total\_commande` qui prend en paramètre :

- `commande\_id` (Identifiant de la commande).
- Elle retourne le montant total de la commande en multipliant la quantité commandée par le prix des produits.

## 2. Créer une fonction : Vérifier la disponibilité d'un produit

Créez une fonction appelée `verifier\_disponibilite` qui prend en paramètre :

- `produit\_id` (Identifiant du produit).
- Elle retourne "Disponible" si le stock est supérieur ou égal à 10, sinon "Rupture de stock".

## 3. Créer une fonction : Appliquer une réduction

- Créez une fonction appelée `appliquer\_reduction` qui prend en paramètre :

- `prix` (Prix initial).
- `pourcentage` (Pourcentage de réduction).
- Elle retourne le prix après application de la réduction.