

# TP 8 : PL/SQL – Les Déclencheurs (Les Triggers)

Soit le schéma relationnel de la base de données commerciale "BDCOM":

---

**Client** (Codc, Nomc, Ville, CA, Cred, CredMax)

**Commande** (Numc, Datec, Mntc, #Codc) *-- '#' indique que l'attribut est une clé étrangère*

**Produit** (RefP, Libp, PU, Qtes, Seuil)

**Fournisseur** (Codf, Nomf, Villef)

**Lig\_cde** (#Numc, #RefP, Qtec)

**Frs\_Prod** (#Codf, #RefP, PUf, Qtef)

---

Avant de répondre aux exercices, créer un nouvel utilisateur nommé TP8A, lui attribuer les privilèges nécessaires, créer les tables et insérer les données. Ainsi, on vous demande de suivre les étapes suivantes :

- `Connect system/mot de passe de system`
- `Create user TP8A identified by tp8;`
- `Grant connect, resource, create view to TP8A;`
- `Connect TP8A/tp8`
- `Start c:\tp8\creationTP8.sql` *-- on suppose que le dossier tp8 est dans c:\*
- `Start c:\tp8\insertionTP8.sql`
- `Set serveroutput on`

## Exercice 1 :

Ecrire un trigger qui permet de mettre à jour la quantité en stock d'un produit après insertion ou mise à jour dans la table Frs\_Prod.

## Exercice 2 :

Ecrire un trigger qui est déclenché avant mise à jour ou suppression d'un produit afin de respecter les conditions suivantes :

- En cas de mise à jour la nouvelle valeur de Qtes ne doit pas être négative.
- En cas de suppression, il faut que la quantité en stock d'un produit soit égale à 0.

## Exercice 3 :

Ecrire un trigger qui permet de s'assurer que le crédit d'un client est égal à 0 avant de le supprimer. Si le crédit du client est égal à 0, il faut archiver ses données ainsi que celles de ses commandes dans les deux tables **Client\_OLD** et **Commande\_OLD** (à créer) qui ont les mêmes schémas que ceux des tables **Client** et **Commande**.

## Exercice 4 :

Ecrire un trigger qui est déclenché après mise à jour ou suppression ou insertion dans la table **Lig\_Cde** et qui permet de mettre à jour le montant de la commande, la quantité en stock du produit et le chiffre d'affaire du client.