



Objectif : le but de ce TD est d'assurer des contraintes via des déclencheurs

Aide-mémoire : Définition d'un déclencheur

```
CREATE [OR REPLACE] TRIGGER nom_declencheur
BEFORE|AFTER
INSERT|DELETE|UPDATE | INSERT [[OR] DELETE] [[OR] UPDATE]
ON nom_table
[FOR EACH ROW [WHEN <condition>]]
Bloc PL/SQL
```

Exercice 1 : Transformation, vérification et mise à jour automatique des données

Soit la table "Citoyen" qui présente les informations d'une personne et le numéro son conjoint. **Citoyen** (NumC, Nom, Prenom, NomJeuneF, numConjoint*)

CITOYEN	
P	NUMC INTEGER
	NOM VARCHAR2 (20 BYTE)
	PRENOM VARCHAR2 (20 BYTE)
	NOMJEUNEF VARCHAR2 (20 BYTE)
F *	NUMCONJOINT INTEGER
PK_CITOYEN (NUMC)	
FK_CONJOINT (NUMCONJOINT)	

- 1.Transformer, au moment de l'insertion en majuscule, la valeur du nom du citoyen quel que soit son format d'origine (utilisation de UPPER()).
- 2.Vérifier, au moment de l'insertion ou modification, que le nom des deux conjoints est le même.

Exercice 2 : Problème d'interblocage

Soient les deux triggers "T1" et "T2" sur la table "tab1" et la table "tab2" de la même base de données

- 1.Expliquer le problème que pose le déclenchement du Trigger T1.

```
CREATE TRIGGER "T1"
AFTER DELETE ON tab1
FOR EACH ROW
BEGIN
    UPDATE tab2
    SET attribut1='A'
END ;
```

```
CREATE TRIGGER "T2"
AFTER UPDATE OF attribut1 on tab2
FOR EACH ROW
BEGIN
    INSERT INTO tab1 Values (1,'A');
END ;
```

- 2- corriger les erreurs du déclencheur suivant

```
CREATE OR REPLACE TRIGGER "T2"
AFTER UPDATE OF attribut1 on tableT2
FOR EACH ROW
BEGIN
    :NEW.attribut1 := 'A';
END ;
```

Exercice 3 :

Soit la base de données pour la gestion d'évènements sportifs "DZZumbaDays". Cet Évènement est organisé le 15 octobre de chaque année dans un lieu différent, en deux sessions : session "matin" et "après-midi"

EVENEMENT (NumEven, LieuEven, adresse, capacité).

PARTICIPATION (NumPart, numPersonne*, NumEven*, Session)

PERSONNE (numPersonne, nom, âge, téléphone)

capacité : nombre de participants que peut accueillir le lieu où est organisé l'évènement : par exemple l'évènement "DZZumbaDays2023" sera organisé dans le stade "Frère Zerga Tlemcen". La capacité est de 100 participants ; deux sessions seront organisées : matin et après-midi donc le champ "session" peut prendre deux valeurs : matin , après-midi.

- Définir un déclencheur qui permet d'interdire une participation si l'évènement affiche complet (la capacité est atteinte).

Exercice 4 :

Soit le schéma relationnel (déjà vu dans la série 1 -Vues-) suivant :

CLIENT(NumCli, CINcli, Nomcli, Adrcli, Telcli, NumConjoint*, NbCptE)

OPERATION(NumOp, TypeOP, MtOp, NumCpt*, DateOp) **TypeOp='D'** ou **'R'**

COMPTE(NumCpt, SoldeCpt, TypeCpt, NumCli*) **TypeCpt='CC'** ou **'CE'**

Assurer l'ensemble des contraintes suivantes en définissant des déclencheurs

- 1- Mettre à jour le solde du compte client après chaque opération effectuée
- 2- Interdire aux clients de retirer un montant supérieur au solde du compte courant
- 3- Interdire à un client d'avoir plusieurs comptes courants
- 4- Mettre à jour automatiquement le champ NbCptE
- 5- Interdire à un client de faire plus de 2 opérations de retrait par jour sur le même compte.

Exercice 5 :

Soit le schéma relationnel suivant :

EMP(EmpNo, Enom, Eprenom, DirNo*, Sal, Prime, SertNo *)

SERVICE(SertNo, Dnom, ADR, NbPersonnel, Budget)

Projet (Numprojet , coûtP) \\ coûtP : coût global du projet

SERVICE_Projet(SertNo, num projet , pourcentage) ; // pourcentage désigne le pourcentage de coûtP.

- 1- Dans la table SERVICE la colonne NbPersonnel doit toujours contenir le nombre d'employés de chaque SERVICE, gérer automatiquement cette valeur.
- 2- Sur la table EMP, on veut imposer la contrainte suivante : après l'insertion ou modification dans la table EMP, la somme des salaires ne doit pas dépasser 1 000 000 DA.
- 3- Interdire à l'utilisateur 'Mohammed' de modifier le salaire de plus de 20%.
- 4- La somme des coûts des projets d'un service ne doit pas dépasser le budget de ce dernier.