

Exercices – Triggers

J. Darмонт (*jerome.darmont@univ-lyon2.fr*), 22/03/2000

Exercice 1

Soit une table quelconque TABL, dont la clé primaire CLENUM est numérique. Définir un trigger en insertion permettant d'implémenter une numérotation automatique de la clé. Le premier numéro doit être 1.

Exercice 2

Soit le schéma relationnel d'une agence bancaire régionale.

CLIENT (NUMCL, NOM, PRENOM, ADR, CP, VILLE, SALAIRE, *CONJOINT*)
DETENTEUR (*NUMCL*, *NUMCP*)
COMPTE (NUMCP, DATEOUVR, SOLDE)

Attributs soulignés : Clés primaires.

Attributs en italiques : Clés étrangères.

NUMCL et CONJOINT sont définis sur le même domaine.

Écrire un trigger en insertion permettant de contrôler les contraintes suivantes :

- le département dans lequel habite le client doit être 01, 07, 26, 38, 42, 69, 73, ou 74 ;
- le nom du conjoint doit être le même que celui du client.

Correction Exercice 1

```
create or replace trigger cleauto
before insert on tabl
for each row

declare

    n integer;
    newkey integer;
    preums exception;

begin

    -- Recherche s'il existe des tuples dans la table
    select count(*) into n from tabl;
    if n=0 then
        raise preums; -- Premiere insertion
    end if;

    -- Recherche la valeur de cle C la plus elevee
    -- et affecte C+1 a la nouvelle cle
    select max(clenum) into newkey from tabl;
    :new.clenum := newkey + 1;

exception

    -- Premier numero = 1
    when preums then :new.clenum := 1;

end;
.
```

Correction Exercice 2

```
CREATE TRIGGER INS_CLIENT
BEFORE INSERT ON CLIENT
FOR EACH ROW

DECLARE

    nom_conjoint CLIENT.NOM%TYPE ;
    compteur CLIENT.NUMCL%TYPE ;
    pb_dept EXCEPTION ;
    pb_conjoint1 EXCEPTION ;
    pb_conjoint2 EXCEPTION ;

BEGIN

    -- Contrainte sur le département

    IF TRUNC(:NEW.CP/1000) NOT IN (01, 07, 26, 38, 42, 69, 73, 74) THEN
        RAISE pb_dept ;
    END IF ;

    -- Contrainte sur le nom du conjoint (+ test d'existence du conjoint)

    IF NEW.CONJOINT IS NOT NULL THEN
        SELECT COUNT(*), NOM
        INTO compteur, nom_conjoint
        FROM CLIENT
        WHERE NUMCL = :NEW.CONJOINT
        GROUP BY NOM ;
        IF compteur = 0 THEN -- Pas de conjoint
            RAISE pb_conjoint1 ;
        END IF ;
        IF nom_conjoint <> :NEW.NOM THEN
            RAISE pb_conjoint2 ;
        END IF ;
    END IF ;

EXCEPTION

    WHEN pb_dept THEN RAISE_APPLICATION_ERROR (-20501,
        'Insertion impossible : le client n'habite pas en région Rhône-Alpes !') ;

    WHEN pb_conjoint1 THEN RAISE_APPLICATION_ERROR (-20502,
        'Insertion impossible : le conjoint du client n'existe pas !') ;

    WHEN pb_conjoint2 THEN RAISE_APPLICATION_ERROR (-20503,
        'Insertion impossible : le nom du conjoint est différent de celui du
        client !') ;

END ;
.
```