

TD3 - Les triggers

Exécuter le script `td3-tables.sql` pour créer la table `Article(id int, nom varchar, prix numeric)` et y insérer les données, et pour créer la table `Facture(id# int, quantite int, prix_total numeric)`. (Les prix sont exprimés en euros.)

Exercice 1 :

Donner des explications et des commentaires pour chaque question.

- Exécuter le script `td3-trigger.sql` pour créer le trigger `trig_four` et sa fonction `fourniture()`.
- Lister le contenu de la table `Facture` et y insérer les valeurs suivantes pour (`id`, `quantité`) : (101, 4), (102, 1), (103, 3), (104, 8).
- Lister le contenu de la table `Facture` et y insérer les valeurs suivantes pour (`id`, `quantité`, `prix_total`) : (105, 10, 0), (106, 12, 7.54), (107, 6, 2.28), (108, 5, 0).
- Modifier la quantité de l'article 109 dans la table `Facture` pour la passer à 9 unités.
- Insérer dans la table `Facture` les informations suivantes : (110, 1, 0).
- Pour tous les articles dont le prix total est supérieur à 12.50 euros, passer leur quantité à 0 dans la table `Facture`. Lister le contenu de la table `Facture`.
- Arrondir le prix unitaire de l'article 102 dans la table `Article` vers l'entier le plus proche.
- Comparer le prix unitaire de l'article 102 dans la table `Article` avec le prix total de l'article 102 contenu dans la table `Facture`.
- Supprimer de la table `Article` tous les articles dont la quantité est 0 dans la table `Facture`.
- Rajouter la contrainte `NOT NULL` à l'attribut `prix_total` de la table `Facture`. Désactiver le trigger `trig_four` défini sur la table `Facture`. Insérer dans la table `Facture` les valeurs suivantes pour (`id`, `quantité`) : (109, 20).
- Réactiver le trigger `trig_four` défini sur la table `Facture`, et insérer les valeurs (109, 20).

Exercice 2 :

- Exécuter à nouveau le script `td3-tables.sql`. Créer 2 tables `Article_sc` et `Facture_sc` ayant les mêmes attributs et les mêmes domaines de définition que les tables `Article` et `Facture` respectivement, mais sans les contraintes. Utiliser la commande :

```
create table nom_table2 as select * from nom_table1;
```
- Vérifier la structure et le contenu de ces nouvelles tables.
- Ecrire un trigger `trig_pk_Article_sc` pour simuler la clé primaire sur la table `Article_sc`.
Rappel : la clé primaire vérifie à l'insertion et la mise à jour d'un tuple que la valeur proposée pour les attributs de la clé primaire n'existent pas déjà dans la table. S'ils existent, il faut refuser de faire la commande.
- Tester son déclenchement par les commandes suivantes :
 - insérer un nouvel article dont l'identifiant vaut null, le nom est trieur et le prix est 9.99 euros.
 - mettre à jour l'article numéro 105 en mettant son identifiant à null.
 - insérer un article dont l'identifiant est 105, le nom est agrafeuse et le prix est 8.29 euros.
 - mettre à jour l'article dont l'identifiant est 107 avec l'identifiant 105.
 - insérer un article dont l'identifiant est 100, le nom est ardoise et le prix est 4.89 euros.
 - mettre à jour le nom de cet article en tablette LED.
 - détruire cet article dont l'identifiant est 100.