

TP2 - Protection des données sous SQL Server

Trigger

Khalid Gaber

On veut modéliser le fonctionnement de la caisse d'un magasin d'alimentation. Chaque produit a un code et un prix (on ne tiendra pas compte des marques : à un produit donné correspond un et un seul prix susceptible d'être mis à jour). On souhaite tenir un historique des ventes et une liste de clients privilégiés (les abonnés) à qui on fait une remise de 3% sur leurs achats et qui peuvent toucher cette remise dès qu'elle atteint 10€. Par ailleurs, on veut en profiter pour introduire une petite gestion des stocks. Pour simplifier, on suppose que le magasin a une seule caisse.

Pour cela, on considère les tables suivantes :

ABONNE(IdAbonne, nom, remise)

PRODUIT(IdProduit, nom, prix)

STOCK(codeproduit, quantité, limite)

VENTE(codeclient, codeproduit, quantité)

TICKET(somme)

REAPPROVISIONNEMENT(racheter, code)

La table ABONNE contient une liste des clients abonnés et, pour chacun, la remise à laquelle il a droit. La table STOCK contient l'état du stock pour chaque produit (attribut quantité) et la limite inférieure à partir de laquelle le magasin doit commander ce produit. La table TICKET contient une seule ligne qui est la somme partielle que le client qui est en train de passer à la caisse devra payer. Cette ligne est mise à jour au fur et à mesure que les achats du client passent en caisse. La table REAPPROVISIONNEMENT contient la liste des produits que le magasin doit racheter : l'attribut racheter contient le nom du produit et l'attribut code son code.

Ecrire en Transact SQL les actions suivantes :

1. Ecrire un trigger (déclencheur) qui met à jour la table TICKET.
2. Ecrire un bloc SQL qui affiche le contenu de la table TICKET et remet celui-ci à zéro.
3. Ecrire un trigger qui met à jour la table STOCK au fur et à mesure des achats d'un client.
4. Ecrire un trigger qui complète la table REAPPROVISIONNEMENT quand la quantité d'un produit en stock descend en dessous de la valeur limite.
5. Ecrire un trigger qui met à jour l'attribut remise de la table ABONNE.
6. Ecrire un trigger pour que dès que la remise d'un client abonné atteint 10,00€, celle-ci soit débitée automatiquement de la table ticket et sa valeur remise à 0.