

TP5 : Les triggers

Partie 1 : Suite tp 4

1) Écrivez le déclencheur **TrigPassagerConducteur** sur la table transporter permettant de vérifier qu'à chaque nouveau transport, le passager déclaré n'est pas déjà enregistré en tant que conducteur le même jour.

2) Écrivez le déclencheur composé **TrigcapaciteVehicule** sur la table transporter permettant de contrôler, qu'à chaque nouveau transport, la capacité du véhicule n'est pas dépassée.

Vous éviterez le problème des tables mutantes en :

- déclarant dans la zone de définition commune un tableau recensant le nombre de personnes transportées par visite ;
- déclarant dans cette même zone un curseur qui va parcourir toutes les visites ;
- chargeant le tableau dans la section BEFORE STATEMENT ;
- examinant le tableau dans la section BEFORE EACH ROW et en le comparant avec les données à insérer.

Les messages à afficher pour tracer et rendre plus lisible ce déclencheur sont :

- dans la section BEFORE EACH ROW : "Enregistrement du transport de *nom*" puis éventuellement

" Premier trajet de la visite" ;

- dans la section AFTER EACH ROW : "Transport de *nom* bien enregistré" puis "Il ne reste plus que *x* place(s) disponible(s)" ;
- dans la section AFTER STATEMENT : "Nombre de trajet(s) traité(s) : *nombre*" ;

Les messages d'erreur à produire le cas échéant sont les suivants :

- "Capacité max atteinte *n* pour la visite *chantier* du *date*, pour le véhicule *v*" ;
- "BASE INCORRECTE : Capacité dépassée *n* pour la visite *chantier* du *date*, pour le véhicule *v*".

Partie II :

1) Construire la table services définie par le schéma suivant

Services(nserv, libellés).

- 2) Ajouter à emp le champ nserv sans indiquer de contrainte référentielle.
- 3) Construire un trigger SupprCasc qui réalise la suppression en cascade entre services et emp.
- 4) Vérifier si le trigger SupprCasc est fonctionnel ou non.
- 5) Construire un trigger MajCasc qui réalise la mise à jour en cascade entre services et emp.
- 6) Vérifier si le trigger MajCasc est fonctionnel ou non.
- 7) Ajouter à la table services le champ nbre_empls qui stocke le nombre d'employés de chaque service.
- 8) En utilisant une seule instruction update, mettre à jour le champ nbre_empls.
- 9) Ecrire un trigger qui à la suite d'un ajout d'un employé à un service, incrémente le nbre_empls de ce service. Vérifier la fonctionnalité de votre trigger.

- 10)** IDEM que précédente, mais pour une opération de suppression.
- 11)** IDEM que précédente, mais pour une modification de nserv dans emp.
- 12)** Construire une relation audit_emp contenant le champ de nemp et les champs :
- opération : nature de l'opération effectuée sur emp (insertion, suppression et modification),
 - utilisateur : le nom de l'utilisateur qui a effectué opération,
 - date_opération : date de l'opération.
- 13)** Construire les triggers nécessaires qui alimentent la table audit_emp et vérifier s'ils sont fonctionnels.