

Nous considérons le schéma relationnel suivant modélisant les activités d'une agence de location de voitures. On suppose cette base créée dans le schéma « rapideloc » et que vous êtes administrateur.

Vendeurs (idvdr, nom, prenom)

Clients (idclt, nom, adresse, nbvoitureslouees)

Voitures (idvoit, dateconstruction, daterevision, kilometrage, #idmarque)

Marques (idmarque, nomMarque)

Locations (idloc, dateloc, duree, KMdepart, KMretour, etat, #idclt, #idvoit, #idvdr)

L'état de location correspond à « en cours », « rendue » ou « retard ». Les kilomètres retour ne peuvent être remplis qu'au retour.

I. Administration Base de Données

1. Créer les rôles suivants :

- RVendeur** : il a tous les droits sur les tables clients et locations mais ne peut que consulter le reste
- RClient** : il ne peut que consulter les tables voitures et locations
- RComptable** : accès en lecture seule aux tables locations, voiture et marques.
- RGaragiste** : accès total sur la table voitures. Il peut consulter toutes les autres tables.

2. Créer les vues suivantes et les droits qui vont avec :

- La liste des voitures (toutes informations) disponibles à la location, triée par marque. Cette vue devra être accessible aux clients et aux vendeurs.
- La liste des clients (id et nom) ayant loué une voiture et dont la durée est dépassée. Cette vue ne sera accessible qu'aux vendeurs.
- La liste des voitures à réviser, c'est-à-dire les voitures ayant parcouru plus de 50 000km et n'ayant pas été révisées dans les deux dernières années. Cette vue ne sera accessible qu'aux garagistes.
- La liste des voitures louées par les vendeurs dans l'année en cours, avec les durées de location. Cette vue ne sera accessible qu'au comptable pour l'établissement des primes des vendeurs.

II. PL/SQL

- Les identifiants des 5 tables étant des entiers, créer les séquences pour les 5 tables.
- Créer un trigger pour l'insertion d'une nouvelle location. Il devra valider les champs saisis et déclencher une erreur qui affichera un message si le champ n'est pas valide. En particulier, il devra capter l'inexistence du client, de la voiture ou du vendeur. La date devra être celle du jour. Le kilométrage sera automatiquement rempli comme égale au kilométrage de la voiture. La durée sera celle demandée et l'état sera positionné à « en cours ». Enfin, le champ nbvoitureslouees de la table Clients devra être mise à jour.
- Créer un trigger validant le retour d'une voiture louée. L'état passera à « rendu » et le kilométrage de retour sera validé. De plus, le kilométrage de la voiture devra être mise à jour.
- Définir une procédure **derniereLocation()** qui affiche, pour chaque client, la date, le numéro et la marque de la dernière voiture louée.
- Définir une procédure **verifLocation()** qui met à jour la table location en passant l'état à « retard » lorsque la durée de location est dépassée.