

## Mission 1 : Retrait et restitution du véhicule loué

Documents à utiliser : 1 et 2

### Création d'un déclencheur (*trigger*)

Dans la base de données, on trouve deux types de formules :

- les formules de type "avec chauffeur" qui ont toutes la même durée de location : 24h ;
- les formules de type "sans chauffeur" qui permettent de louer un véhicule pour une durée exprimée en heures : 4 heures, 24 heures, 48 heures.

Un déclencheur *trgDateRetourLAC*, fourni dans le dossier documentaire, existe déjà. Il permet de mettre à jour la date et heure de fin de location (colonne *dateHreRetourPrevu* de la table *Location*) pour une location avec chauffeur. Il est exécuté lors de l'ajout d'une location avec chauffeur.

Un déclencheur *trgDateRetourLSC* doit être ajouté pour mettre à jour la colonne date et heure de fin de location de la table *Location* pour une location **sans chauffeur**.

**Question 1** : Écrire le déclencheur *trgDateRetourLSC*. Pour cela, vous utiliserez la partie de la base de données entourée dans le document 1.

### Statistiques sur les locations sans chauffeur

Le module "statistiques sur les locations" de l'application «*GestActivité*» doit être modifié pour obtenir des informations sur l'activité location sans chauffeur. De nouvelles statistiques vont permettre d'anticiper le renouvellement de la flotte automobile de **Localux**.

Pour les locations sans chauffeur, **Localux** veut obtenir les modèles les plus demandés.

Votre chef de projet vous demande de réaliser une requête qui permette d'obtenir le nombre de locations sans chauffeur pour chaque nom de modèle, avec les modèles les plus demandés en tête de classement.

**Question 2** : Écrire la requête répondant au besoin exprimé par votre chef de projet.

**Question 3** : Nombre de locations et montant Regle des locations de l'année 2009.

**Question 4** : Afficher le nom de chaque client avec le nombre de locations. Liste triée dans l'ordre décroissant du nombre de locations.

**Question 5** : Même question avec les clients qui n'ont encore aucune location.

**Question 6** : Donner la requête SQL permettant d'augmenter de 10% le prix du tarif du kilomètre supplémentaire dans la relation *Modele*.

**Question 7** : Donner la requête SQL pour supprimer le client Durand

**Question 8** : Créer en SQL une vue *LocationYear* qui pour chaque année, donne le chiffre d'affaires (cumul des montants réglés :

**Question 9** : Créer un utilisateur ayant pour nom votre login et pour mot de passe "mdp"

**Question 10** : Donner à votre utilisateur les droits SELECT et UPDATE sur votre BDD

**Question 11** : Retirer le droit UPDATE sur votre base de données à cet utilisateur:

**Question 12**: Supprimer votre utilisateur

## MCD MODELE CONCEPTUEL DE DONNEES

**Question 13** : Fournir le MCD du document 1

# Document 1 : Structure de la base de données LOCALUX

---

**Vehicule** (immatriculation, dateAchat, idModele)

*Clé primaire : immatriculation*

*Clé étrangère : idModele en référence à id de Modele*

**Modele** (id, nom, tarifKmSupplementaire, nbPlaces, vitesseMax, description)

*Clé primaire : id*

**EtreProposee** (idModele, idFormule, tarif)

*Clé primaire : idModele, idFormule*

*Clés étrangères : idModele en référence à id de Modele  
idFormule en référence à id de Formule*

**Formule** (id, libelle)

*Clé primaire : id*

**FormuleAvecChauffeur** (id, lieuDestination, tarif)

*Clé primaire : id*

*Clé étrangère : id en référence à id de Formule*

**FormuleSansChauffeur** (id, duree, nbKmlInclus)

*Clé primaire : id*

*Clé étrangère : id en référence à id de Formule*

**Location** (numLocation, dateLocation, montantRegle, dateHreDepartPrevu, dateHreDepartReel, dateHreRetourPrevu, dateHreRetourReel, idClient, immatriculation)

*Clé primaire : numLocation*

*Clés étrangères : idClient en référence à id de Client  
immatriculation en référence à immatriculation de Vehicule*

**LocationSansChauffeur** (numLocation, nbKmDepart, nbKmRetour, idFormule)

*Clé primaire : numLocation*

*Clés étrangères : numLocation en référence à numLocation de Location  
idFormule en référence à id de FormuleSansChauffeur*

**LocationAvecChauffeur** (numLocation, idFormule)

*Clé primaire : numLocation*

*Clés étrangères : numLocation en référence à numLocation de Location  
idFormule en référence à id de FormuleAvecChauffeur*

**Pays** (id, nom)

*Clé primaire : id*

**Client** (id, nom, prenom, mail, tel, adresse, cp, ville, motDePasse, idPays)

*Clé primaire : id*

*Clé étrangère : idPays en référence à id de Pays*

## Remarques :

- Toutes les colonnes contenant des dates (dont le nom commence par "date") sont de type horodatage ;
- La colonne duree dans la table FormuleSansChauffeur est exprimée en heures entières

## Document 2 : Déclencheur trgDateRetourLAC

---

```
CREATE TRIGGER trgDateRetourLAC BEFORE INSERT ON LocationAvecChauffeur
FOR EACH ROW
BEGIN
  DECLARE duree INTEGER ;
  DECLARE dateDepart DATETIME;
  DECLARE dateRetour DATETIME;
  /* la durée est de 24h pour une location avec chauffeur */
  SET duree = 24;
  /* récupération de la date de départ dans une variable */
  SET dateDepart = (SELECT dateHreDepartPrevu FROM Location
                    WHERE numLocation = NEW.numLocation) ;
  SET dateRetour = DATE_ADD(dateDepart, INTERVAL duree HOUR) ;

  UPDATE Location set dateHreRetourPrevu = dateRetour
                where numLocation = NEW.numLocation ;
END ;
```

**Remarque** : la fonction **DATE\_ADD**(date, **INTERVAL** *nb* **unitéDeTemps**) ajoute *nb* unités de temps à la date passée en paramètre (*nb* pouvant être une valeur ou le nom d'une variable)

Exemples de valeurs possibles pour **unitéDeTemps** : SECOND, HOUR, DAY, etc.