

## Exercice 2 MEA Modèle relationnel et SQL

---

L'agence LouerViteEtBien décide d'automatiser sa gestion. Cette gestion consiste à gérer un parc de plusieurs centaines d'appartements appartenant à des propriétaires et loués lors des différentes saisons.

Un appartement a un prix de base qui correspond à celui d'une semaine de location en basse saison et qui est spécifique à l'appartement. Les locations sont proposées aux clients « à la semaine ». Le client fait sa réservation avec le versement d'un acompte. L'agence établit un contrat de location où le montant d'une caution et les dates de séjour sont notamment précisés. Un client peut louer plusieurs appartements mais chaque contrat ne concerne qu'un seul appartement.

Une première étude, à partir de ces informations a conduit à établir le modèle entité association de l'annexe A.

Cette étude est incomplète en ce qui concerne la tarification des appartements. En fait, un appartement est d'un type donné que l'on a l'habitude d'identifier par un code et auquel on associe un libellé, par exemple: « 2 pièces », « studio + mezzanine », « studio + terrasse » etc.). Les prix des autres saisons sont obtenus par application d'un coefficient standard pour tous les appartements. Ce coefficient dépend du type de l'appartement et de la saison. La saison est découpée en semaines successives. On a donc défini un code saison auquel correspondent une date de début de saison et une date de fin de saison. **Pour simplifier, on suppose qu'un client ne peut pas réserver à cheval sur 2 saisons.**

Q1) Compléter le MEA en fonction de ces informations.

Q2) Dériver le modèle relationnel.

---

Q3) Ecrire les requêtes SQL suivantes:

- 1) Quels sont les numéro et tarif de base des appartements ne faisant pas l'objet de réservation ? Présentez le résultat trié du tarif le plus cher au tarif le moins cher.
- 2) On cherche pour chaque appartement, le nombre de contrats en cours et le montant global des acomptes reçus ?
- 3) Pour quel type d'appartement, l'agence a-t-elle le plus de contrats?

### Exercice 1

On considère la base de données suivante qui contient des informations sur la gestion d'un camping (placement de vacanciers et location de sports - tennis, vtt et autre - qu'ils peuvent exercer pendant leur séjour).

Cette base de données décrit les emplacements proposés par le camping. Chaque emplacement est identifié par un numéro, il a une certaine surface, il n'accepte qu'un nombre maximum de personnes, il est d'un certain type (tente, caravane, bungalow) et il servira à différents groupes de vacanciers tout au long de la saison. A chaque groupe accueilli au camping, on associe un numéro de client, on note le nom d'un des clients du groupe, la date de début de séjour, le nombre de personnes concernées et le numéro d'emplacement où ce groupe a été installé. Différents sports sont proposés aux vacanciers à l'unité de location. Chaque sport est affiché à un certain tarif pour cette unité de location. Il est aussi stocké le nombre d'unités de chaque sport loué par un groupe de vacanciers (sur l'ensemble de son séjour).

Emplacement(numEmpl, #numCl, surface, nbPersMax, type)

Client(numCl, nomClient, debutSejour, nbPers)

Location(#numCl, #nomSport, uniteLocation, tarifUnite, nbUnitesLouées)

1. Rappelez la définition d'une clé primaire et d'une clé étrangère.
2. Cette base de données est en réalité mal construite car elles présentent des anomalies de redondances d'informations. Lesquelles ?
3. Proposez une nouvelle version de cette BD sans anomalies.
4. Donnez les **expressions algébriques** correspondant aux requêtes suivantes **sur la version de BD corrigée à la question 3**.
  - a) Donnez le nom des sports non loués à ce jour
  - b) Donnez le nom des clients installés dans un bungalow ; indiquez le nombre de personnes du groupe de ce client.

## Exercice 2

L'image a été enregistrée dans /home/hafsaoui/Ima

Soit la base de données d'un cabinet d'architectes contenant les relations suivantes :

Batiment(idBat, adresse)

Etage(#idBat, numEtage, surfaceEtage)

Local(idLocal, #idBat, #numEtage, typeOccup, surfaceLocal)

Les trois tables satisfont les contraintes suivantes :

– Les attributs des clés primaires sont soulignés, les clés étrangères précédées d'un #.

- Le numéro d'étage est un entier dans l'intervalle  $[-3, 120]$ .
- La surface d'un étage et d'un local est un décimal avec deux chiffres après la virgule.
- Le type d'occupation est une chaîne de caractère de longueur variable avec une longueur maximale de 32 caractères.
- Tous les attributs sauf les surfaces doivent être renseignés au moment de l'insertion d'un n-uplet.
- Le schéma garantit l'intégrité référentielle entre toutes les tables.

1. Écrire la commande SQL pour la création de la table *Etage*.
2. Écrire une requête SQL, sous la forme d'une vue, qui compte le nombre de locaux actuellement répertoriés sur chaque étage.
3. Modifier la structure de la table *Etage* pour rajouter une colonne *nbLocaux* qui permettra de stocker le nombre de locaux à l'étage ; initialiser sa valeur à 0.
4. En utilisant la vue de la question 2, écrire une requête SQL pour mettre à jour automatiquement la colonne *nbLocaux* dans la table *Etage*.
5. Afin de ne pas mettre à jour la base manuellement à chaque fois que l'on ajoute un nouveau local à un étage, créer un déclencheur/trigger et sa fonction PL/PGSQL qui

Soit le schéma relationnel d'une Société Fictive d'Archéologie<sup>1</sup> qui gère des trésors archéologiques et des articles scientifiques sur ces trésors :

```
OBJET(numObjet, description, type, datation, etatConservation,  
      #numSite, #numMusee)  
VILLE(numVille, ancienNom, nomActuel)  
MUSEE(numMusee, #numVille, nomMusee)  
SITE(#numVille, numSite, descriptionSite, civilisation)  
ARTICLE(numA, titre, date)  
AUTEUR(numAuteur, nomAuteur, prenomAuteur)  
COOPERATION(#numAuteur, #numA)  
REFERENCE(#numA, #numObjet)
```

La relation OBJET décrit les objets proprement dits avec l'indication de leur type (par exemple *vase, frise*), de leur datation (par exemple *moyen-âge, premier siècle*), de leur état de conservation (par exemple *partiel ou total*), du site où ils ont été découverts et du musée où ils se trouvent actuellement. La relation VILLE comprend deux noms pour les villes ayant un nom historique comme par exemple *Byzance – Constantinople* (les deux noms sont identiques

---

<sup>1</sup> Extrait de *Bases de données : objet & relationnel*, G. Gardarin, Eyrolles, 1999

si le nom de la ville est resté le même) . La relation SITE indique la ville à laquelle se rattache le site identifié par un numéro. La civilisation d'un site est une grande catégorie comme *romaine*, ou *crétoise*. Les articles sont la plupart du temps co-écrits et peuvent concerner plusieurs objets archéologiques.

Retrouvez le MEA dont ont été dérivées ces relations.

#### Exercice 4

### Exercice 1 Algèbre relationnelle

---

Le schéma relationnel suivant représente la gestion simplifiée de magasins de vente de CD. Les clés primaires sont soulignées, les clés étrangères sont précédées d'un dièse.

*CD( NumCD, NomAlbum, Durée, Distributeur, #NumAuteurParole, #NumAuteurMusic)*

*Auteur(NumAuteur, Prénom, Nom, Nationalité )*

*Magasin(NumMagasin, NomMag, Adresse, Département, NomResponsable )*

*CDDansMagasin( #NumCD, #NumMagasin, NbCDDispo, PrixVente )*

Ecrivez les requêtes suivantes en ALGEBRE RELATIONNELLE

- 1) Donnez la liste des noms des auteurs étrangers
- 2) Donnez les numéros des albums qui n'ont pas de point de vente
- 3) Donnez la liste des personnes (description totale) étant à la fois auteurs de paroles et de musique.