

# Audit d'une agence de voyages

Julien Vion

## 1 Compétences mises en œuvre

Ces compétences seront évaluées individuellement à la fin du cycle d'APP :

- Lire et interpréter un modèle relationnel,
- Algèbre relationnelle,
- Pratiquer le langage SQL.

## 2 Le problème

Une agence de voyages stocke l'ensemble de ses clients et des stations où ceux-ci sont susceptibles de séjourner dans un SGBD. À chaque fois qu'un client séjourne dans une station, l'information est stockée dans la base. Chaque station propose un certain nombre d'activités.

Pour la préparation d'un audit de l'agence, on nous demande d'extraire un certain nombre de statistiques de la base de données :

1. Lister toutes les stations connues.
2. Lister les clients européens.
3. Lister les stations situées en Afrique ou en Océanie.
4. Quelles sont les activités de la station « Venusa » ?
5. Lister tous les séjours du client Philéas Fogg (en indiquant le nom de la station).
6. Lister tous les séjours ayant eu lieu sur la station « Venusa » (en indiquant le nom du client).
7. Trier les stations en fonction du nombre d'activités proposées.
8. Quels sont les clients ayant séjourné à Venusa ?
9. Combien de stations sont situées en Afrique ?
10. Quel est le total des soldes des clients ?
11. Quel est le chiffre d'affaire de la société ?
12. Combien chaque station a-t-elle généré de CA ?

De plus, la direction de l'agence souhaite se prémunir d'éventuelles questions imprévues. Proposez d'autres statistiques intéressantes et non triviales (c'est-à-dire comportant au moins une jointure et/ou un agrégat).

## 3 Travail demandé

Le schéma relationnel (les tables) de la base de données est représenté sur la figure 1, et le code SQL correspondant pour générer les tables est donné sur le listing 1. Un fichier `.sql` permettant de générer les tables et d'y insérer des données est disponible sur *Moodle*.

Indiquer, pour chaque requête, sa formulation en algèbre relationnelle (si possible, c'est-à-dire si la requête ne comporte pas d'agrégat), puis la requête SQL. N'essayez pas d'indiquer les éventuels tris en algèbre relationnelle :

1. Utilisez la méta-commande `\i` de `psql` pour charger un fichier.

ce n'est pas possible. Pour les requêtes supplémentaires, énoncez-les d'abord en langage naturel.

## 4 Préparation

### 4.1 En groupe

Réécrivez le schéma relationnel sous forme de relations (inventez quelques données), identifiez les opérateurs d'algèbre relationnelle qui vous seront utiles. Les requêtes sont de difficulté croissante. Assurez-vous que chaque membre du groupe ait bien compris une requête avant de passer à la suivante. Commencez par les requêtes pouvant être écrites en algèbre relationnelle, c'est-à-dire les requêtes ne nécessitant pas l'utilisation d'agrégats. Pensez déjà à proposer vos propres requêtes ! Ainsi, vous n'aurez pas besoin à ce stade de connaître tous les mots-clés SQL. Ne passez au SQL que s'il vous reste du temps en fin de séance.

### 4.2 Individuellement

Prenez connaissance avant le début de la séance de TP des mots-clés SQL correspondant aux opérateurs d'algèbre relationnelle que vous avez utilisés lors de la séance de groupe. Prenez également connaissance du fonctionnement des fonctions d'agrégat.

## 5 Réalisation

Créez la base de données à partir du fichier disponible sur *Moodle*<sup>1</sup>. Écrivez les requêtes en SQL. Une fois fonctionnelles, recopiez les au propre sur papier. Identifiez les différences entre le SQL et l'algèbre relationnelle : sont-elles normales ou les requêtes élaborées en groupe étaient-elles fausses ?

## 6 Mise en commun du travail et présentation du travail réalisé

Travaillez par deux pour comparer les requêtes SQL de chacun, puis réalisez une présentation orale d'une durée de 8 minutes. Inutile de recopier chaque requête SQL. Indiquez la formule d'algèbre relationnelle pour chaque requête, identifiez les cas fondamentalement différents et indiquez une requête SQL pour chacun de ces cas.

Voici le barème d'évaluation :

- Chaque formule d'algèbre relationnelle : 1 point
- Chaque cas de passage de l'algèbre au SQL identifié et correctement expliqué : 2 points
- Chaque cas redondant :  $-\frac{1}{2}$  point
- Qualité de la présentation : 2 points

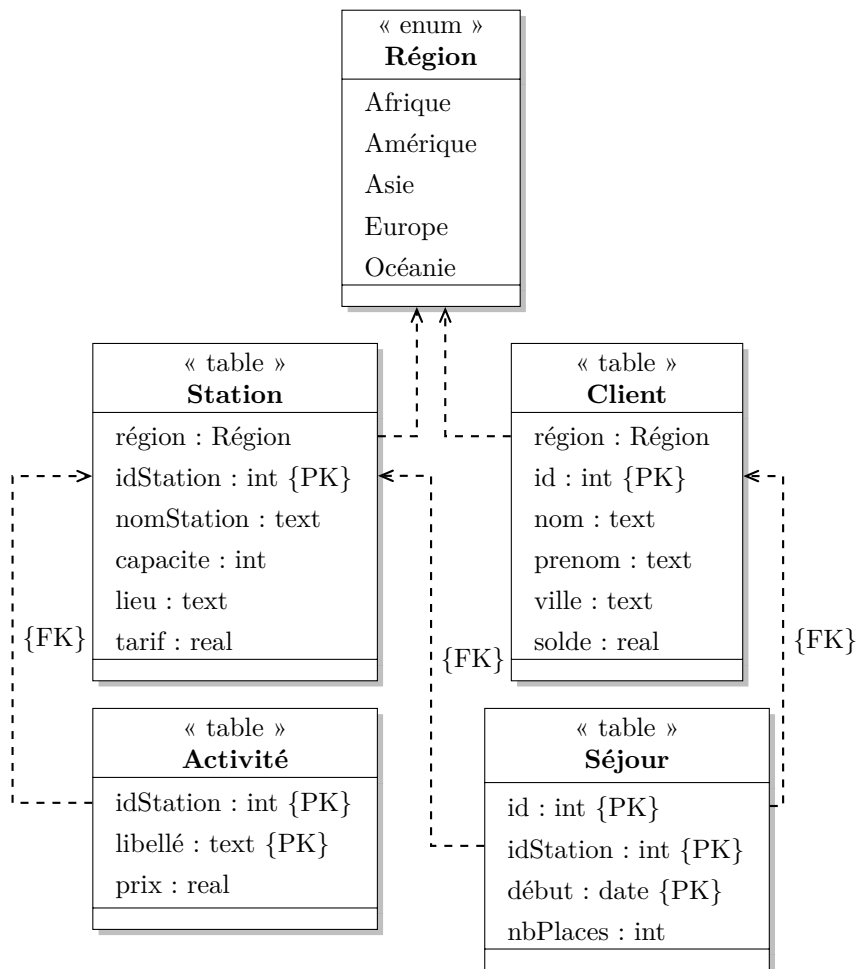


FIGURE 1 – Modèle relationnel de la base de données

```
CREATE TYPE Region AS ENUM ('europe', 'asie',  
    'océanie', 'Amérique', 'afrique');
```

```
CREATE TABLE Station (  
    idStation int PRIMARY KEY,  
    nomStation text,  
    capacite int,  
    lieu text,  
    region Region,  
    tarif real);
```

```
CREATE TABLE Client (  
    id int PRIMARY KEY,  
    nom text, prenom text,  
    ville text,  
    region Region,  
    solde real);
```

```
CREATE TABLE Activite (  
    idStation int REFERENCES Station ,  
    libelle text,  
    PRIMARY KEY (idStation , libelle),  
    prix real CHECK (prix >= 0));
```

```
CREATE TABLE Sejour (  
    id int REFERENCES Client ,  
    idStation int REFERENCES Station ,  
    debut date,  
    PRIMARY KEY (id , idStation , debut),  
    nbPlaces int);
```

Listing 1 – Commandes SQL de création des tables

