

# Fiche de TP no 1

Considérons la base de données *Hotellerie* dont le schéma est le suivant :

```
hotel(numhotel, nom, ville, etoiles)
chambre(numchambre, numhotel, etage, type, prixnuitht)
client(numclient, nom, prenom)
occupation(numoccup, numclient, numchambre, numhotel, datearrivee, datedepart)
reservation(numresa, numclient, numchambre, numhotel, datearrivee, datedepart)
```

## 1 Compréhension du schéma

**Question 1.** Quel(le)s sont, dans le schéma ci-dessus,

- les tables ?
- les attributs ?
- les clés primaires ?
- les clés étrangères ?

**Question 2.** Dans le schéma ci-dessus, les types de colonnes ne sont pas spécifiés. Quel type de donnée (entier, réel (flottant), chaîne de caractères, date, ...) donneriez vous à chaque colonne (de chaque table) ?

**Question 3.** Quelles sont les tables de cette DB sur lesquels on peut faire un

- `PRODUIT`,
- `UNION`,
- `NATURAL_JOIN` ?

Pour quelle couple de table serait préférable utiliser `JOIN` à la place de `NATURAL_JOIN` ?

## 2 Prise en main de SQLite3

Sous Linux, placez vous dans une répertoire nommé `BD/TP1`.

Récupérez les fichiers pour construire et inhabiter la base `hotellerie.db` depuis l'adresse

<http://pageperso.lis-lab.fr/~luigi.santocanale/teaching/BD/code/BD-Hotel-SQLite.zip>

**Question 4.** Dans un terminal, lancez `sqlite3` de façon a créer la base `hotellerie.db` :

```
> sqlite3 hotellerie.db
```

**Question 5.** Faire `.help` : vous obtenez l'aide de `sqlite3`. En particulier, les commandes suivantes pourraient s'avérer utiles : `.read`, `.header`, `.mode`, `.exit`, ...

**Question 6.** Utilisez les fichiers téléchargés (avec la commande `.read`) pour construire les tables et y ajouter les données. Vous pouvez ouvrir les fichiers `schema.sql` et `insertion.sql` avec `geany` (ou votre éditeur favori) pour découvrir leur contenu.

**Question 7.** Utilisez la requête

```
SELECT * FROM Hotel ;
```

Pour afficher le contenu de la table `Hotel`. Attention à ne pas oublier le `;` à la fin.  
De façon similaire, affichez le contenu de chaque table.

**Question 8.** Sélectionnez un mode d’affichage des tables qui plus vous convient. Pour ce faire, demandez (via la commande `.help .mode`) quels sont les modes d’affichage disponibles.

**Question 9.** Sortez de `sqlite3`. Jetez un œil au fichier de sauvegarde. Quelle est sa taille ?

**Question 10.** Relancez `sqlite3 3` avec la base `hotellerie.db`. Vérifiez que tout a bien fonctionné : les tables sont toujours là ? les données aussi ?

**Question 11.** Sortez de `sqlite3`. Utilisez **DB Browser for SQLite3** pour afficher les tables de la base `hotellerie.db` ainsi que la structure de la base de données et de chaque table.

### 3 Interrogation des données

**Question 12.** Considérez les expressions de l’algèbre relationnelle ci dessous. Pour chaque expression :

- écrivez une requête SQL équivalente,
- exécutez, avec `sqlite3`, cette requête sur la base de données `hotellerie.bd`,
- proposez une phrase équivalente dans la langue française.

1.

```
PROJECT(Hotel, etoiles)
```

2.

```
PROJECT(Chambre, type)
```

3.

```
PROJECT(RESTRICT(Hotel, etoiles <= 2), nom, ville)
```

4.

```
PROJECT(  
  RESTRICT(  
    NATURAL_JOIN(Client, Occupation)  
    datearrive >= '2014-04-01')  
  nom, prenom)
```

5.

```
PROJECT(  
  RESTRICT(  
    NATURAL_JOIN(Client, NATURAL_JOIN(Occupation, Hotel))  
    ville = Lyon')  
  nom, prenom)
```

**Question 13.** Exécutez la requête suivante (depuis `sqlite3` ou depuis **DB Browser for SQLite3**) sur la base de donnée `hotellerie.db` :

```
SELECT * FROM
client NATURAL JOIN hotel NATURAL JOIN occupation
```

- Combien d'enregistrements sont trouvés ?
- Expliquez pourquoi.

Comment modifier cette requête pour trouver dans quelle ville chaque client a séjourné ?

**Question 14.** Écrivez en ensuite exécutez les requêtes SQL correspondant aux questions suivantes. Avant rédiger les requêtes SQL, listez, pour chaque question, les opérations de l'algèbre relationnelle qu'il faut utiliser.

**Conseil :** créez un fichier `tp1.sql` dans `geany` (ou un autre éditeur de texte) et écrivez les requêtes dans ce fichier ; puis copier-coller les dans le terminal où `sqlite3` est ouvert pour les tester (sous linux : copier = surligner, coller = cliquer bouton du milieu sur la souris). Pour exécuter les requêtes, vous pouvez utiliser `sqlite3` ou **DB Browser for SQLite3**.

1. Listez les noms d'hôtels et leur ville (12 enregistrements trouvés)
2. Listez les noms d'hôtels ayant 3 étoiles ou plus. (2 enregistrements trouvés)
3. Listez les hôtels sur lesquels portent une réservation (30/9 enregistrements trouvés, selon que l'on utilise `DISTINCT`)
4. Listez les clients ayant déjà occupé (ou en cours d'occupation d') une chambre.
5. Pour chaque hôtel, indiquer le nombre total de chambres. (il faudra utiliser les fonctions d'agrégats, `COUNT` avec `GROUP BY` en SQL).
6. Quels sont tous ceux qui ont été, ou sont encore, clients de l'hôtel des voyageurs à Nice ?
7. Combien de chambres triples possède l'Hôtel des ambassadeurs à Grenoble ?  
(1 enregistrements trouvés)
8. Pour chaque client, lister le nom et la ville des hôtels dans lesquels il a séjourné, ou séjourne encore, avec les dates d'arrivée et de départ correspondant à chaque séjour.
9. Listez les numéros de chambres disponibles dans l'hôtel de la gare, à Bordeaux, pour les nuits du 12 avril 2014 au 17 avril 2014.