

## TP n° 14 : Interrogation d'une BDD

### Exercice 1 – Requêtes dans une base de donnée

#### Préambule : les bases de données sous Python

Dans ce TP, on manipule une base de données *via* Python. Cela nécessite l'import de la librairie `sqlite3`.

Il faut alors créer un objet de connection vers une base de données, par une instruction :

```
connection = sqlite3.connect('nom_de_la_base.db')
```

S'il n'existe pas de base de ce nom, il sera créé une base vide. Si le fichier n'est pas dans le répertoire de travail, il faut indiquer son chemin.

L'étape suivante est de définir un « curseur » vers cet objet de connection : c'est ce curseur qui va permettre de récupérer le résultat d'une requête :

```
cur = connection.cursor()
```

Toutes les requêtes SQL envoyés à la base (que ce soit pour la création, la modification, ou le questionnement de la base de données) le sont alors par l'instruction suivante :

```
cur.execute("""Instruction en syntaxe SQL""")
```

Ainsi, *via* `cur.execute`, on peut envoyer les requêtes une à une, sous forme d'une chaîne de caractères (les « raw string » sont plus prudents, et permettent notamment de faire facilement des retours à la ligne, ce qui est appréciable dans les requêtes un peu longues).

À la fin du programme, on ferme la connection à la base avec :

```
connection.commit()
connection.close()
```

À l'issue de chaque requête, `cur` est un objet itérable, les termes de l'itération étant des tuples de valeurs issues de la selection. Pour afficher le résultat, il convient donc d'afficher ces tuples les uns après les autres, à l'aide d'une boucle sur cet objet itérable.

La base de donnée étudiée ici est donnée par le fichier `ma_banque.db`, à récupérer sur ma page web ou sur une clé. Les tables de cette base ont été créées par les instructions SQL suivantes :

```
CREATE TABLE IF NOT EXISTS client (
    idclient int(5) NOT NULL,
    nom varchar(128) NOT NULL,
    prenom varchar(128) NOT NULL,
    ville varchar(128) NOT NULL,
    PRIMARY KEY (idclient)
);

CREATE TABLE IF NOT EXISTS compte (
    idcompte int(5) NOT NULL,
    idproprietaire int(5) NOT NULL REFERENCES client,
    type varchar(128) NOT NULL,
```

```

PRIMARY KEY (idcompte)
);

CREATE TABLE IF NOT EXISTS operation (
  idop int(5) NOT NULL,
  idcompte int(5) NOT NULL REFERENCES compte,
  montant decimal(10,2) NOT NULL,
  informations text NOT NULL,
  PRIMARY KEY (idop)
);

```

On fera un affichage des réponses à l'écran, ainsi qu'une sauvegarde ligne par ligne dans un fichier `reponse_mabanque.txt`, les réponses aux différentes questions étant mises les unes derrière les autres, séparées par une ligne blanche et une ligne indiquant le numéro de la question. La manipulation des fichiers est expliquée dans le chapitre 2 de votre cours.

1. Donner le nom et le prénom de tous les clients.
2. Donner le nom et le prénom des clients habitant à Paris.
3. Donner les identifiants des comptes de type Livret A.
4. Donner les identifiants des opérations de débit sur le compte d'identifiant égal à 1.
5. Donner, sans doublon, les identifiants des propriétaires de livret A, classés par ordre croissant.
6. Donner l'identifiant des clients n'ayant pas de livret A.
7. Donner l'identifiant de compte et le type de compte des clients habitant à Paris.
8. Donner la liste des comptes et les types de compte de Dumbledore.
9. Donner le nombre de clients par ville, classé par ordre alphabétique de villes
10. Donner la ville ayant le plus de clients
11. Trouver le nombre d'opérations effectuées sur chaque compte
12. Trouver le nombre maximum d'opérations effectuées sur un compte.
13. Trouver le ou les numéros de compte réalisant le maximum de la question précédente.
14. Afficher, type par type, la moyenne des soldes des comptes (tous clients confondus) de chaque type (en supposant qu'initialement, les comptes sont tous vides).
15. Afficher, classé par nom et prénom, le nom, le prénom, le type de compte, et le solde, pour tous les comptes.
16. Même question, en se limitant aux clients dont le nom commence par K,L,M ou N.
17. Afficher le nom et le prénom des personnes ayant débité au moins un chèque sur leur compte courant, classé par nom.
18. Nom, prénom et ville de tous les clients ayant réalisé un nombre maximal d'opérations au guichet.
19. Moyenne par ville des fortunes totales des clients (somme sur tous leurs comptes), classé par valeur croissante.