Mastère Spécialisé Big Data : gestion et analyse des données massives

TP n°4 de Bases de Données

Accès aux données avec Python et construction d'un schéma en étoile

Environnement informatique

En plus de l'environnement de SQLite et « DB Browser for SQLite » utilisés jusqu'à présent, nous allons accéder à une base de données SQLite au travers d'un langage de programmation, à savoir le langage Python à l'aide d'un notebook Jupyter.

Ouvrir un notebook Jupyter dans votre environnement :

- Sur votre poste personnel, par exemple en ayant installé au préalable par exemple `Anaconda`
- Sur les machines de TP de l'école, en tapant `jupyter notebook` dans une fenêtre terminal. Si besoin, mettre au préalable un mot de passe avec la commande `jupyter notebook password`.

En Python, vous aurez besoin d'importer les librairies `sqlite3` et `pandas` pour accéder à la base de données et créer un dataframe à partir du résultat d'une requête SELECT.

Rendu du TP :

- Un lien vers un fichier *zip* contenant le fichier de votre base de données + les scripts SQL que vous aurez écrits + le notebook Jupyter + les résultats obtenus éventuellement sous forme de copies d'écran.
- Donner le lien d'accès au fichier *zip* en utilisant le formulaire suivant : https://docs.google.com/forms/d/e/1FAIpQLSexbm6P20o8rvmh4R3hMQbbC-9jKOqtv4LOFBU644T3RI8Pgw/viewform?usp=pp_url

Interagir avec une base SQLite depuis le langage Python

Créer un nouveau Notebook Jupyter où vous importerez les bibliothèques `sqlite3` et `pandas`.

En vous inspirant de https://www.sqlitetutorial.net/sqlite-python/ et de https://andrewpuleo.com/sqlite-and-jupyter/ , réaliser les opérations suivantes (**dans le notebook**) :

1. Connectez-vous à la base « TP_python_sqlite.db » fournie sur le site pédagogique, en la recopiant au préalable dans votre répertoire.
2. Faire afficher par la requête « `select * from MAGASIN` » le contenu de la table « `MAGASIN` » qui est déjà créée dans la base (fournir une solution avec la fonction `fetchall` puis une autre avec la fonction `fetchone`).
3. Créer un dataframe à partir du résultat de la requête « `select * from MAGASIN` » et faire afficher le contenu du dataframe.

4. Créer une table « `employe` » avec 2 colonnes : « `nom` » de type TEXT et « `age` » de type INTEGER, de clé primaire « `nom` ». Bien créer la table depuis Python dans le Notebook et non pas avec `sqlite3`.
5. Insérer dans la table « `employe` » le n-uplet avec les valeurs « `toto` » et « `25` ».
6. Faire afficher le contenu de la table « `employe` » (requête « `select * from employe` ; »). Ouvrir une fenêtre `sqlite3` ou `DB Browser for SQLite` : la mise à jour est-elle visible ?
7. Rendre permanente la mise à jour effectuée.
8. Ecrire un programme qui demande à saisir un nom et un âge, et qui insère les informations saisies dans la table « `employe` » ; rendre permanente la mise à jour effectuée.
9. Ecrire un programme qui demande à saisir un état et qui exécute la requête retrouvant tous les magasins situés dans cet état (requête SELECT paramétrée sur la table `MAGASIN`).

Création d'un schéma en étoile pour stocker un cube de données

La base de données fournie contient les tables suivantes décrivant les ventes d'une chaine de supermarchés :

- MAGASIN : décrit les magasins de la chaine
- PRODUIT : décrit les produits qui sont vendus dans l'ensemble des magasins
- PROMOTION : décrit les promotions possibles dans les magasins
- VENTES : décrit pour chaque jour, chaque magasin, chaque produit et chaque type de promotion le chiffre d'affaire associé (CA) et le nombre d'exemplaires vendus

Concevoir et créer dans la base un schéma en étoile pour stocker un cube de données avec les dimensions suivantes :

- MAGASIN : NOM→VILLE→COMTE→ETAT
- TEMPS : JOUR→MOIS→ANNEE
- PRODUIT : LIBELLE→SOUS-CATEGORIE→CATEGORIE et LIBELLE→MARQUE
- PROMOTION : NOM_PROMOTION→TYPE_PUBLICITE

Avec les mesures CA et NB.

Ne pas oublier :
- De générer des identifiants entiers pour les valeurs de dimension (dans les tables de dimensions et dans la table de faits)
- De déclarer les clés primaires et étrangères du schéma en étoile

Ecrire en SQL la requête suivante sur le schéma en étoile : *chiffre d'affaire par marque de produit, comté du magasin et année, pour les magasins de l'état de Californie.*

Indication pour créer les identifiants entiers des valeurs d'une dimension : créer une table vide avec une clé primaire « autoincrement » et la remplir à partir des tables fournies.