



Epreuve d'informatique

2^{ème} semestre AU : 2023 – 2024

Date : Mai 2024 ; Durée : 2H ; Nombre de page : 4

L'utilisation des calculatrices n'est pas autorisée pour cette épreuve.

Implémentation. Dans ce sujet, nous adopterons les syntaxes des langages SQL & Python.

Problème 1: Analyse de Données sur la Biodiversité

Objectif : Développer un script Python pour analyser les données de biodiversité animale, en utilisant des fichiers, des dictionnaires et des listes.

Description : Vous avez un fichier texte contenant des données sur différentes espèces animales observées dans divers écosystèmes. Chaque ligne du fichier représente une observation avec le nom de l'espèce, le nombre d'individus observés, et l'écosystème où l'observation a eu lieu.

Exemple de contenu du fichier (observations.txt) :

Loup, 5, Forêt boréale
Salamandre, 12, Forêt tempérée
Ours, 3, Toundra
Loup, 8, Forêt boréale
Salamandre, 9, Forêt tempérée

Instructions de l'exercice

1. Lire le fichier : Écrire un script pour lire le fichier observations.txt et stocker les informations dans une liste de dictionnaires où chaque dictionnaire contient nom_espece, nombre_individus et ecosysteme.

Exemple :

```
[  
  {'nom_espece': 'Loup', 'nombre_individus': 5, 'ecosysteme': 'Forêt boréale'},  
  {'nom_espece': 'Salamandre', 'nombre_individus': 12, 'ecosysteme': 'Forêt tempérée'},  
  {'nom_espece': 'Ours', 'nombre_individus': 3, 'ecosysteme': 'Toundra'},  
  {'nom_espece': 'Loup', 'nombre_individus': 8, 'ecosysteme': 'Forêt boréale'},  
  {'nom_espece': 'Salamandre', 'nombre_individus': 9, 'ecosysteme': 'Forêt tempérée'}  
]
```

2. Analyser les données :

- Calculer le nombre total d'individus observés pour chaque espèce.
- Identifier l'écosystème avec le plus grand nombre d'espèces différentes.

Exemple :

```
{'Loup': 13, 'Salamandre': 21, 'Ours': 3}
```

L'écosystème avec le plus grand nombre d'espèces différentes est "Forêt tempérée".

3. Afficher les résultats : Afficher les résultats de l'analyse de manière lisible pour un utilisateur.
4. Écrire dans un fichier : Écrire les résultats dans un nouveau fichier appelé resultats.txt

Problème 2: Surveillance Volcanique : Analyse et Données

Description :

Dans ce problème, vous allez exploiter une base de données sur le thème des volcans pour effectuer des analyses en utilisant l'algèbre relationnelle, le SQL et Python avec SQLite. Les données concernent les volcans, leurs éruptions, et les observations scientifiques associées. Votre objectif est d'extraire des informations clés qui aideront à comprendre les activités volcaniques et à prévoir les éruptions futures.

Contexte

Les participants travaillent sur une base de données de volcanologie qui comprend trois tables principales :

- Volcans (volcans): contient des informations sur différents volcans.
- Éruptions (éruptions): enregistre les détails des éruptions pour chaque volcan.
- Observations (observations): recueille des données d'observations scientifiques faites avant, pendant, et après les éruptions.

Structure des Tables

1. Volcans

- volcan_id : identifiant unique du volcan
- nom : nom du volcan
- pays : pays où le volcan est situé
- type : type de volcan

2. Éruptions

- eruption_id : identifiant unique de l'éruption
- volcan_id : identifiant du volcan associé
- date_debut : date de début de l'éruption
- date_fin : date de fin de l'éruption
- indice_explosivite : indice d'explosivité volcanique (VEI)

3. Observations

- observation_id : identifiant unique de l'observation
- eruption_id : identifiant de l'éruption observée
- date_observation : date à laquelle l'observation a été faite
- description : description de l'observation

Partie 1 : Algèbre Relationnelle (10%)

1. Expression Relationnelle :

- Exprimez en algèbre relationnelle la requête pour obtenir les noms des volcans et les dates de toutes leurs éruptions. Assurez-vous de montrer clairement comment vous combinez les relations pour obtenir les résultats désirés.

Partie 2 : SQL (50%)

2. Requête SQL des Volcans par Pays :

- Écrivez une requête SQL pour trouver tous les volcans d'un pays spécifique et les détails de leurs éruptions.

3. Requête SQL sur l'Indice Explosivité :

- Créez une requête SQL qui liste les éruptions avec un indice d'explosivité supérieur à 4 et les observations correspondantes.

4. Jointure sur Trois Tables :

- Écrivez une requête SQL qui joint les trois tables pour obtenir une liste des volcans, avec pour chacun la dernière éruption et la dernière observation faite.

Partie 3 : Python avec SQLite (40%)

5. Script Python pour Requêtes :

- Utilisez Python et SQLite pour écrire un script qui connecte à la base de données et exécute la requête du point 2 (Requête des Volcans par Pays).

6. Fonction Python pour Éruptions par Pays :

- Créez une fonction Python qui prend en paramètre le nom d'un pays et retourne toutes les éruptions de ce pays avec les observations liées, en utilisant des jointures.