

Travaux dirigés

Thème : Données structurées et base de données (SQLite)



I) ORGANISATION DE LA BASE DE DONNEES RELATIONNELLE

Ouvrir la base de données `villes_en_france.db`

- 1) Quel est le nom des tables de cette base de données ?

VILLE et DEPARTEMENT

- 2) Quelle est la clé primaire de la table VILLE ?

Num_ville

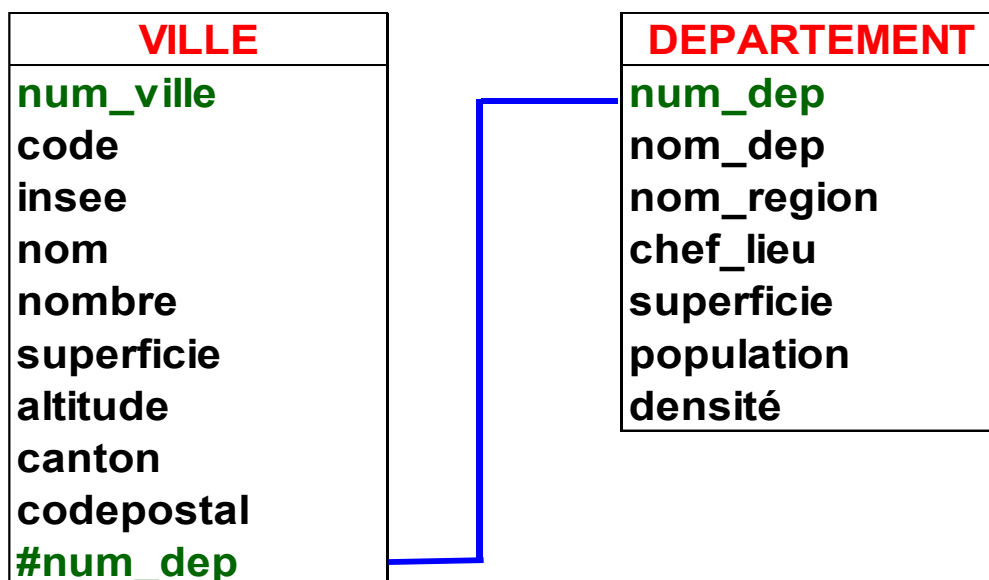
- 3) Quelle est la clé primaire de la table DEPARTEMENT ?

Num_dep

- 4) Quelle est la clé étrangère de la table VILLE ? Quel est son rôle ?

Num_dep est la clé étrangère de la table VILLE. La clé étrangère permet de mettre les 2 tables (VILLE et DEPARTEMENT) en relation.

- 5) Présenter le schéma relationnel de cette base de données



6) Combien y a-t-il d'enregistrements dans la table VILLE ?

36 229 villes

7) Combien y a-t-il d'enregistrements dans la table VILLE ?

101 départements

8) Pourquoi le champ [nom] ne peut pas être la clé primaire de la table VILLE ?

Plusieurs villes peuvent porter le même nom. Or, la clé primaire ne peut pas comporter 2 valeurs identiques d'où l'intérêt d'avoir comme clé primaire un numéro.

9) Que signifient les types de champ INTEGER et TEXT ?

DEPARTEMENT	
num_dep	INTEGER
nom_dep	TEXT
nom_region	TEXT
chef_lieu	TEXT
superficie	TEXT
population	INTEGER
densité	TEXT

**Les champs de type INTEGER contiennent des nombres entiers
Les champs de type TEXT contiennent des données alphanumériques**

II) INTERROGATION DE LA BASE DE DONNEES RELATIONNELLE

1) Comment interroger une base de données SQLite ?

En effectuant des requêtes en SQL

2) Que signifie la requête suivante ? Tester cette requête

```
SELECT *  
FROM VILLE  
WHERE num_dep=72  
ORDER BY NOM
```

Afficher les villes du département 72 par ordre alphabétique

3) Que signifie la requête suivante ? Tester cette requête

```
SELECT *  
FROM VILLE  
WHERE num_dep=72 OR num_dep=49 OR num_dep=44
```

Afficher les villes des départements 44, 49 et 72

4) Que signifie la requête suivante ? Tester cette requête

```
SELECT *  
FROM VILLE  
WHERE superficie>=35000
```

Afficher toutes les villes de plus de 35 000 km2

5) Que signifie la requête suivante ? Tester cette requête

```
SELECT nom_dep  
FROM DEPARTEMENT  
WHERE nom_region='Occitanie'
```

Afficher tous les départements de la région Occitanie

6) On vous fournit la requête SQL suivante :

```
SELECT D.nom_dep, SUM(V.superficie) AS superficie_totale  
FROM VILLE V  
JOIN DEPARTEMENT D ON V.num_dep = D.num_dep  
GROUP BY V.num_dep  
HAVING superficie_totale > 1000000
```

a) Cette requête contient 5 lignes. Sur quelle ligne voit-on la jointure entre les tables VILLE et DEPARTEMENT ?

JOIN est sur la 3^{ème} ligne. JOIN permet la JOINTURE entre les 2 tables

b) Le résultat de la requête est le suivant :

	nom_dep	superficie_totale
1	Gironde	1014685

- Interpréter ce résultat

En France métropolitaine, un seul département dépasse 1 000 000 km2. C'est la Gironde (1 014 685 km2)

III) Etude d'un script écrit en Python

On vous fournit le script suivant :

```
import sqlite3

# Connexion à la base de données
conn = sqlite3.connect('villes_en_france.db')
cursor = conn.cursor()

# Requête 1
cursor.execute('''
    SELECT COUNT(*) FROM VILLE
''')
resultat_1 = cursor.fetchone()
print("Requête 1")
print(resultat_1[0], " villes en France")
print()

# Requête 2
cursor.execute('''
    SELECT AVG(nombre) FROM VILLE
''')
resultat_2 = cursor.fetchone()
print("Requête 2")
print(resultat_2[0]*1000, " habitants")
print()
resultat_2 = cursor.fetchall()

# Affichage des résultats sans parenthèses ni apostrophes
for resultat in resultat_2:
    print(resultat[0], ":", " ", resultat[1], " habitants au km2")

print()

# Requête 3
cursor.execute('''
    SELECT D.nom_dep, SUM(V.superficie) AS superficie_totale
    FROM VILLE V
    JOIN DEPARTEMENT D ON V.num_dep = D.num_dep
    GROUP BY V.num_dep
    HAVING superficie_totale > 900000
''')
resultat_3 = cursor.fetchall()
print("Requête 3")
print(resultat_3, "► Nombre de km2")
print()

# Requête 4
cursor.execute('''
    SELECT AVG(altitude) FROM VILLE
''')
resultat_4 = cursor.fetchone()
print("Requête 4")
print(resultat_4[0], "mètres")
print()
```

```

# Requête 5
cursor.execute('''
    SELECT nom FROM VILLE
    WHERE altitude = (SELECT MIN(altitude) FROM VILLE)
''')
resultat_5 = cursor.fetchone()
print("Requête 5")
print(resultat_5[0])
print()

# Requête 6
cursor.execute('''
    SELECT nom_region, SUM(population) AS population_totale
    FROM DEPARTEMENT
    GROUP BY nom_region
''')

resultat_population_par_region = cursor.fetchall()

# Affichage des résultats
print("Requête 6")
for resultat in resultat_population_par_region:
    population_formatee = '{:,}'.format(resultat[1]).replace(',', '.') # Utilisation
de la fonction format avec remplacement de la virgule par un point
    print(f"Région : {resultat[0]}, Population totale : {population_formatee}
habitants")

```

1) Indiquer l'utilité de chaque requête

Requête	Interprétation
Requête 1	Compter le nombre total de villes dans la base de données
Requête 2	Calculer la moyenne de la population des villes
Requête 3	Sélectionner les départements avec une superficie totale supérieure à 900 000 km²
Requête 4	Calculer la moyenne d'altitude pour toutes les villes
Requête 5	Ville de France métropolitaine avec l'altitude minimale
Requête 6	Nombre d'habitants par région

2) Ouvrir le fichier `Interroger_villes_enfrance_db01` et vérifier que les réponses de la question 1 sont cohérentes

3) Modifier ce script afin d'obtenir les résultats suivants :

Requête	Interprétation
Requête 1	Compter le nombre total de villes dans la Sarthe
Requête 2	Calculer la moyenne de la population des villes des Pays de la Loire
Requête 3	Sélectionner les départements des Pays de la Loire avec une superficie totale supérieure à 650 000 km ²
Requête 4	Indiquer l'altitude de la ville de BORDEAUX
Requête 5	Ville avec l'altitude minimale
Requête 6	Nombre d'habitants en Bretagne

4) Enregistrer le script. Nom du fichier : **Interroger villes_enfrance_db02**