

Spécialité NSI en terminale T.P.4.: Bases de données et requêtes SQL

Les bases de données et python :

Pour interagir avec une base de données au format SQLite3 depuis Python, il est nécessaire d'utiliser le module sqlite3. `import sqlite3`

Pour se connecter à la base de données (qui doit être dans le même répertoire que le script python), il faut utiliser l'instruction suivante :

```
connexion = sqlite3.connect('Ludotheque.db')#connexion à la base
```

A la fin du script, il faudra se déconnecter de la base à l'aide de l'instruction :

```
connexion.close()#deconnexion de la base
```

Les bases de données sont des systèmes transactionnels. Cela signifie qu'un programme qui interagit avec une base de données fait ses modifications en mémoire, jusqu'à ce que celles-ci soient validées. Lors de cette validation, si les modifications sont toujours possibles, elles sont toutes transférées dans la base de données sur le disque. Sinon, aucune n'est effectuée. C'est cette validation par paquet qui constitue une transaction.

En Python, les transactions sont validées en appelant la méthode **commit()** sur l'objet connexion.

Chaque commit termine la transaction précédente, puis en commence une autre. La première transaction est initiée lors de la connexion à la base de données.

Bien sûr, si aucune modification n'a été apportée à la base, la phase de validation n'est pas nécessaire.

Pour valider les modifications sur une base de données, il faut utiliser l'instruction :

```
connexion.commit()#Validation de la modification de la base
```

Pour exécuter une requête, il faut la passer en paramètre de la méthode **execute()** de l'objet connexion.

Il faut au préalable avoir activé la vérification des clés étrangères en exécutant la requête **PRAGMA foreign_keys = ON**.

```
connexion.execute('PRAGMA foreign_keys = ON')#Activation des clés étrangères
```

Création de la table Editeur dans la base de données Ludothèque :

```
#Création de la table editeur
connexion.execute('CREATE TABLE Editeur (idEditeur int primary key not null,nomEditeur text not null,nationaliteEditeur text)')
```

Valider celle-ci avec l'instruction :

```
connexion.commit()#Validation de la modification de la base
```

Si vous ouvrez le fichier créé avec SQLite on constate bien que la table a été créée.

Nom	Type	Schéma
Tables (1)		
Editeur		CREATE TABLE Editeur (idEditeur int primary key not null,nomEditeur text not null,nationaliteEditeur text)
idEditeur	int	"idEditeur" int NOT NULL
nomEditeur	text	"nomEditeur" text NOT NULL
nationaliteEditeur	text	"nationaliteEditeur" text
Index (0)		
Vues (0)		
Déclencheurs (0)		

Ajout d'un enregistrement : ATTENTION Pour exécuter à nouveau le script, il faut mettre en commentaire la création de la table car celle-ci a été créée lors de la dernière exécution.

```
#Création de la table editeur
```

```
#connexion.execute('CREATE TABLE Editeur (idEditeur int primary key not null,nomEditeur text not null,nationaliteEditeur text)')
```

```
#Enregistrement d'un éditeur PUISSANCE4 de nationalité Française
```

```
connexion.execute('INSERT INTO Editeur VALUES(1,"PUISSANCE4", "Française")')
```

Table : Editeur

idEditeur	nomEditeur	nationaliteEditeur
Filtre	Filtre	Filtre
1	1 PUISSANCE4	Française

On pourrait aussi rajouter un enregistrement grâce à des instructions input à mettre avant l'ouverture de la base de données.

```
# Créé par gbareux, le 24/10/2020 en Python 3.7
import sqlite3
nom_jeu = input("quel est le nom de l'éditeur ? ")
nationalite_editeur = input("quelle est sa nationalité ? ")
connexion = sqlite3.connect('Ludotheque.db')#connexion à la base
connexion.execute('PRAGMA foreign_keys = ON')#Activation des clés étrangères
#Création de la table editeur
#connexion.execute('CREATE TABLE Editeur (idEditeur int primary key not null,nomEditeur text)')
#Enregistrement d'un éditeur PUISSANCE4 de nationalité Française
#connexion.execute('INSERT INTO Editeur VALUES(1,"PUISSANCE4", "Française")')

#Ajout des valeurs saisies par l'utilisateur
connexion.execute('INSERT INTO Editeur VALUES (?, ?, ?)', (2, nom_jeu, nationalite_editeur))
```

A l'exécution :

Python input

quel est le nom de l'éditeur ?

OK Cancel

Python input

quelle est sa nationalité ?

OK Cancel

```
*** Console de processus distant Réinitialisée ***
quel est le nom de l'éditeur ? Days of Wonder
quelle est sa nationalité ? Française
^^^
```

idEditeur	nomEditeur	nationaliteEditeur
Filtre	Filtre	Filtre
1	1 PUISSANCE4	Française
2	2 Days of Wonder	Française

La méthode **execute()** présentée précédemment renvoie un objet **cursor** qui est itérable. On peut donc aisément parcourir les résultats d'une requête dans une boucle for, en itérant sur l'objet **cursor**.

```
#connexion.execute('INSERT INTO Editeur VALUES (?, ?, ?)', (2, nom_jeu, nationalite_editeur))
#affichage du contenu
curseur = connexion.execute("SELECT * FROM Editeur")
for tuple in curseur:
    print(tuple)
```

Pour récupérer les valeurs pour les différents champs de chaque tuple, il est préférable de les convertir en liste.

```
curseur = connexion.execute("SELECT * FROM Editeur")
#for tuple in curseur:
#    print(tuple)
for tuple in curseur:
    donnee = list(tuple)
    print("L'éditeur est :", donnee[1],"et sa nationalité est :",donnee[2])
```

Exercice :

Télécharger la base de données [SpeBac.db](#) et l'ouvrir avec SQLite pour répondre aux questions suivantes :

1. Combien de tables possède la base de données ?

Réponse :

5

2. Combien de champs possède la table Eleve ?

Réponse :

4

3. Quelle est sa clé primaire ?

Réponse :

IdEleve

4. Combien de champs possède la table Professeur ?

Réponse :

3

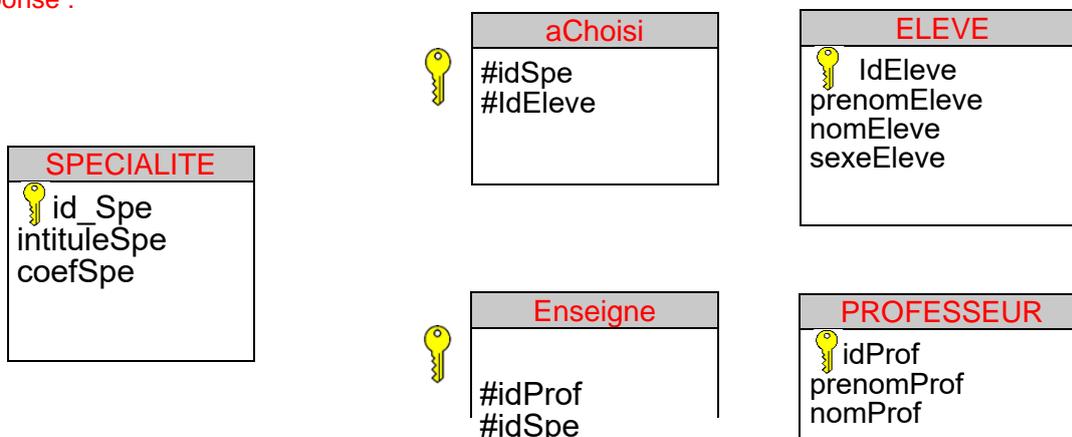
5. Quelle est sa clé primaire ?

Réponse :

idProf

6. En procédant de la même façon pour toutes les tables, représentez le schéma relationnel sous forme graphique, en faisant bien figurer les relations entre les clés primaires et clés étrangères.

Réponse :



En utilisant python et à l'aide des requêtes SQL sur la base SpeBac.db :

7. Afficher le nom, prenom de chaque élève suivi de son identifiant et de son sexe.

On attend :

```
*** Console de processus distant Réinitialisée ***
Maurice McVittie a pour identifiant idEleve : 1 et pour sexe : M
Anthony Cromly a pour identifiant idEleve : 2 et pour sexe : M
Maelys Cortese a pour identifiant idEleve : 3 et pour sexe : F
Marine Willoughy a pour identifiant idEleve : 4 et pour sexe : F
Didier Driscoll a pour identifiant idEleve : 5 et pour sexe : M
Lian Souza a pour identifiant idEleve : 6 et pour sexe : M
Yahn Basford a pour identifiant idEleve : 7 et pour sexe : M
Norbert Tuley a pour identifiant idEleve : 8 et pour sexe : M
Marine Berth a pour identifiant idEleve : 9 et pour sexe : F
Cecile De Blasio a pour identifiant idEleve : 10 et pour sexe : F
Aline Muller a pour identifiant idEleve : 11 et pour sexe : F
Jerome Dupont a pour identifiant idEleve : 12 et pour sexe : M
>>>
```

Réponse :

connexion.execute('Select nom, prenom,idEleve, sexe from eleve')

8. Afficher le nom, prénom de chaque prof suivi de son identifiant

On attend :

```
Steve Mizen a pour identifiant idProf : 1
Christophe Twining a pour identifiant idProf : 2
Milene Denyukhin a pour identifiant idProf : 3
Mathis Dable a pour identifiant idProf : 4
Charles Dupont a pour identifiant idProf : 5
Pierre Durand a pour identifiant idProf : 6
```

Réponse :

```
connexion.execute( 'Select nom, prenom,idEleve from Professeur')
```

9. Mise à jour de la table Eleve : un élève (3) change de spécialité il passe de la spécialité math à physique.

On attend :

```
Spe1=Math, Spé2=Physique,Spé3=NSI et Spé4=SI
idEleve : 3 a choisi la spé: 1
idEleve : 3 a choisi la spé: 4
idEleve : 3 a choisi la spé: 3
Après changement
idEleve : 3 a choisi la spé: 2
idEleve : 3 a choisi la spé: 4
idEleve : 3 a choisi la spé: 3
```

Réponse :

```
connexion.execute( update into Eleve set spé = spé2 where idEleve = 3')
```

10. Compter le nombre d'élèves ayant choisi la spécialité NSI.

On attend :

```
*****
Il y a 9 élèves qui ont choisi l'option NSI
```

Réponse :

```
connexion.execute( 'Select * count(*) from Eleve where spé = 'NSI'
```

11. Afficher Nom et prénom des filles ayant choisi la spécialité NSI, il faudra utiliser une jointure.

On attend :

```
*****
Cortese Maelys a choisi la spé NSI
Willoughey Marine a choisi la spé NSI
De Blasio Cecile a choisi la spé NSI
Muller Aline a choisi la spé NSI
```

Réponse : connexion.execute('Select nom, prenom from eleve where sexe = 'F' and spé ='nsi')

12. Afficher le nombre de filles ayant choisi la spécialité NSI.

On attend :

```
*****
Il y a 4 filles qui ont choisi la spé NSI
```

Réponse :

```
connexion.execute( 'Select nom, prenom from eleve where sexe = 'F' and spé ='nsi')
```