

TP13 INTRODUCTION AUX BASES DE DONNÉES

As usual, l'invitation GitHub pour votre dossier de travail vous a été envoyée par mail.

Partie I

Installation de SQLiteBrowser

Si vous êtes sur le réseau du Lycée, un lien direct vers le lecteur de base de données devrait se trouver sur le bureau.

Si vous travaillez sur votre propre ordinateur, vous pouvez télécharger SQLiteBrowser à l'adresse <http://sqlitebrowser.org> et installer la version adéquate pour votre ordinateur.

Pour chacune des sections suivantes, vous pourrez trouver la « bonne » requête en jouant avec SQLiteBrowser pour la recopier à l'endroit adéquat (sous forme de chaîne de caractères) dans le fichier TP13_intro_BDD.py afin que les tests automatiques puissent fonctionner¹.

Partie II

Jeux de triangles

La base `triangles.sqlite` est constituée d'une seule table dont le schéma relationnel est le suivant :

```
TRIANGLE (idt:int, ab:int, bc:int, ac:int)
```

Chaque ligne représente un triangle défini par la taille de ses trois côtés (AB, AC et BC).

1. Lancer (successivement !) les requêtes suivantes en réfléchissant à leur signification/objectif :

```
SELECT COUNT(*) FROM triangles ;  
SELECT * FROM triangles WHERE ab+ac+bc=100 ;  
SELECT ab*ac*bc FROM triangles WHERE ab+ac+bc>=100 ;
```

NB : par la suite, quand on dit « tous les triangles », on utilisera l'étoile pour récupérer toutes les colonnes disponibles dans l'ordre par défaut de la table.

2. Déterminer, à l'aide de requêtes SQL, la plus petite valeur des produits $AB \times AC \times BC$ pour les triangles ABC de périmètre supérieur ou égal à 100 ;
3. les longueurs AB, AC et BC correspondants au(x) triangle(s) pour le(s)quel(s) le minimum précédent est atteint ;

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

4. tous les triangles rectangles en A ;
5. le nombre de tels triangles ;
6. le maximum des périmètres des triangles rectangles en A ;

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

1. Si vous êtes sûr de votre coup, vous pouvez aussi directement l'écrire dans le fichier python...

7. tous les triangles équilatéraux ;
8. tous les triangles tels que $\frac{AB + AC + BC}{3} = 42$.

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé.
Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

Partie III

Communes, départements et régions




1. Ouvrir la BDD `communes_departements_regions.sqlite` et écrire *sur papier* les schémas relationnels des tables qu'elle contient.
2. Lancer la requête

```
SELECT C.nom, D.nom FROM communes AS C JOIN departements AS D ON C.dep = D.id ;
```

3. En s'inspirant du modèle précédent, donner la liste de toutes les communes avec pour chacune son département, sa région et sa population.
4. Trier la liste précédente par ordre décroissant de population.

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé.
Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

-  5. Déterminer le rang de la ville de Strasbourg dans ce classement.
-  6. Faire de même² avec le Bas-Rhin pour la population des départements.
-  7. Et l'Alsace par rapport aux régions tant qu'à faire³ !

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé.
Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

8. Donner la liste des communes (nom et population) dont le nom commence par **Pa** et se finissant par **is**. On pourra pour cela utiliser le mot-clé **LIKE** (et Google pour savoir comment l'utiliser).
9. Déterminer les communes (nom et population) qui ont strictement moins d'habitants que de lettres dans leur nom⁴.

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé.
Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

2. Celle-ci est un peu plus dure.
3. Mais celle-ci n'est pas beaucoup plus compliquée que la précédente.
4. Là encore, Google est votre ami à condition de lui poser la question gentiment.

Partie IV

Prénoms parisiens

La base `prenoms_paris.sqlite` contient les prénoms enregistrés à l'état civil de Paris depuis 2004 jusqu'en 2013.

1. Donner le schéma relationnel des différentes tables de cette base.
2. Pour chacune des requêtes SQL suivantes, donner la traduction « en français » et vérifier la vraisemblance du résultat depuis SQLite Manager.

```
SELECT DISTINCT prenom FROM prenoms ;  
SELECT SUM(nombre) FROM prenoms ;  
SELECT COUNT(DISTINCT prenom) FROM prenoms ;  
SELECT annee,SUM(nombre) FROM prenoms GROUP BY annee ;
```

3. Combien de fois votre prénom a-t-il été donné à Paris pendant les années concernées ?
On donnera le résultat trié par années croissantes.
4. Même question avec le prénom des professeurs présents.

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

5. Donner, pour chaque année, le nombre de prénoms différents qui ont été enregistrés.
6. Quels prénoms ont été donnés exactement 100 fois sur la période complète ?
Sur au moins une année ?

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

7. Quel est le prénom qui a été le plus donné sur l'ensemble de la période (on donnera aussi le nombre de fois qu'il a été donné) ?
Pour chaque année (pour ce dernier, donner le prénom, l'année et le nombre) ?

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

Partie V

Notes de colles

La base de données `notes_colles.sqlite` contient trois tables décrivant les colles virtuelles données par des agrégés de la promotion 1930 à des agrégés de la promotion 1950.

1. Écrire le schéma relationnel des différentes tables. Comprendre le lien entre les différentes tables.
2. Déterminer la liste des professeurs.
3. Déterminer celle des élèves.
4. Déterminer le nombre de « 20 » qui ont été attribués, ainsi que le nombre de notes majorant 6.



STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

5. Déterminer les notes de Jacques-Louis Lions (triées selon les semaines croissantes).
6. Refaire la même chose avec cette fois le nom des colleurs associés.
7. Déterminer les quadruplets (élève, prof, note, semaine) pour toutes les colles où la note était supérieure ou égale à 19.
8. Déterminer la moyenne des notes de colle de Jacques-Louis Lions.


STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

-  9. Parmi tous les élèves, déterminer le nom de ceux ayant eu au moins 10 notes strictement sous la moyenne.
-  10. Parmi tous les élèves, afficher ceux ayant eu au moins 6 notes strictement supérieures à 18.

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.

11. Déterminer la liste des couples (élève, moyenne).
12. Déterminer le nom du colleur qui tend à donner les meilleurs notes en moyenne.
13. Déterminer pour chaque colleur la moyenne des notes données et la variance (l'écart-type serait plus parlant, mais la fonction `SQRT` n'existe pas en SQLite). Quel est le colleur qui donne le plus souvent les mêmes notes ?
-  14. Écrire (sans tricher) une requête permettant de calculer la moyenne des moyennes des élèves.

STOP GitHub

Allez sur Github Desktop pour faire un commit. Choisissez vous-même (avec pertinence) le résumé. Pensez aussi à appuyer sur le bouton «Push origin» en haut à droite pour mettre à jour sur le web.