

Devoir n°2

Problème : base de données

La numérisation de larges corpus littéraires a permis la création de bases de données contenant des informations relatives au nombre d'occurrences des différents caractères alphanumériques dans diverses langues. Nous allons supposer disposer d'une base de données constituée de trois tables :

On dispose d'une base de données composée de trois tables :

- caractere(idCar, symbole, typeCaractere, codeHTML)
- corpus(idLivre, titre, auteur, annee, nombreCaracteres, langue)
- occurences(idCar, idLivre, nombreOccurences)

Table caractere

idCar	symbole	typeCaractere	codeHTML
65	'A'	'lettre'	'A'
48	'0'	'nombre'	'0'

Table corpus

idLivre	titre	auteur	annee	nombreCaracteres	langue
1	'Germinal'	'Zola'	1885	1152365	'Français'
2	'Les Misérables'	'Hugo'	1862	2245300	'Français'

Table occurences

idCar	idLivre	nombreOccurences
62	31	155
37	21	1550

Les requêtes demandées dans cette partie sont à écrire en langage SQL.

1. Combien d'attributs possède la table **corpus** ?
2. Quel type et domaine peut-on donner à l'attribut **auteur** ?
3. L'attribut **titre** de la table **corpus** est-il une clé primaire et pourquoi ?
4. L'attribut **idCar** de la table **occurence** est-il une clé primaire et pourquoi ?
5. L'attribut **idCar** de la table **caractere** est-il une clé primaire et pourquoi ?
6. Quelles sont les clés primaires à attribut unique de la table **caractere** ?
7. Comment désigne t-on l'attribut **idLivre** de la table **occurences** et pourquoi ?
8. La table **occurences** possède-t-elle une clé primaire ? Préciser et apporter éventuellement des solutions.

9. Dans cette base de données, combien y-a-t-il d'entités et d'associations ?
10. Écrire en langage SQL une requête permettant d'obtenir le nombre d'ouvrages du corpus.
11. Écrire en langage SQL une requête permettant d'obtenir le nom des ouvrages du corpus écrit par Zola, par ordre alphabétique.
12. Écrire en langage SQL une requête permettant d'obtenir le nombre d'ouvrage du corpus écrit au 20ième siècle.
13. Écrire en langage SQL une requête permettant d'obtenir la liste sans doublon des auteurs présents dans la table corpus.
14. Écrire en langage SQL une requête permettant d'obtenir les auteurs et le nombre de ses ouvrages du corpus.
15. Écrire en langage SQL une requête permettant d'obtenir les lettres et leur nombre d'occurences du corpus écrites par Hugo, classés par nombre d'occurences décroissant, avec un nombre d'occurences supérieur à 1000.
16. Écrire une requête SQL permettant d'obtenir la fréquence d'occurrences (donc comprise entre 0 et 1) de chaque caractère en 'Français'. Plus précisément, la requête devra renvoyer une table à deux attributs : une colonne contenant le symbole de chaque caractère, et une autre colonne contenant le rapport entre le nombre total d'occurrences du caractère et le nombre total de caractères des textes de tout le corpus.



1. Combien d'attributs possède la table **corpus** ? La table **corpus** possède 6 attributs (6 colonnes), qui sont **idLivre**, **titre**, **auteur**, **annee**, **nombreCaracteres**, **langue**.
2. Quel type et domaine peut-on donner à l'attribut **auteur** ? Cet attribut est de type chaîne de caractères, avec un domaine de chaîne de longueur variable ou de longueur limité à 100 par exemple.
3. L'attribut **titre** de la table **corpus** est-il une clé primaire et pourquoi ? Non, deux oeuvres distinctes peuvent porter le même titre.
4. L'attribut **idCar** de la table **occurrence** est-il une clé primaire et pourquoi ? Non, pas dans la table **occurrence**, puisque cette table est une table de liaison, et le même caractère peut concerner plusieurs livres (quasi sûrement).
5. L'attribut **idCar** de la table **caractere** est-il une clé primaire et pourquoi ? Oui, il permet d'identifier le caractère.
6. Quelles sont les clés primaires à attribut unique de la table **caractere** ? Les attributs uniques **idCar**, ou **symbole** ou encore **codeHTML** sont des clefs primaires.
7. Comment désigne t-on l'attribut **idLivre** de la table **occurrences** et pourquoi ? Il s'agit d'une clef secondaire, permettant la liaison à la table **corpus** dans laquelle il est une clef primaire.
8. La table **occurrence** possède-t-elle une clé primaire ? Préciser et apporter éventuellement des solutions. une clef à attribut unique, non. On peut par exemple ajouter un attribut **id** entier incrémentable.
Par contre le couple **idCar**, **idLivre** est une clef primaire.
9. Dans cette base de données, combien y-a-t-il d'entités et d'associations ? Nous avons 3 entités (tables) et deux associations. On peut bien sûr ajouter que la table **occurrences** est celle qui permet d'associer les deux tables **caractere** et **corpus**.
10. Écrire en langage SQL une requête permettant d'obtenir le nombre d'ouvrages du corpus.

```
SELECT COUNT(*) FROM corpus
```

11. Écrire en langage SQL une requête permettant d'obtenir le nom des ouvrages du corpus écrit par Zola, par ordre alphabétique.

```
SELECT titre FROM corpus  
WHERE auteur = 'Zola'  
ORDER BY titre
```

12. Écrire en langage SQL une requête permettant d'obtenir le nombre d'ouvrage du corpus écrit au 20ième siècle.

```
SELECT COUNT(*) FROM corpus
WHERE annee >= 1901 AND annee <= 2000
```

13. Écrire en langage SQL une requête permettant d'obtenir la liste sans doublon des auteurs présents dans la table corpus.

```
SELECT DISTINCT auteur FROM corpus
```

14. Écrire en langage SQL une requête permettant d'obtenir les auteurs et le nombre de ses ouvrages du corpus.

```
SELECT auteur, COUNT(*) FROM corpus
GROUP BY auteur
```

15. Écrire en langage SQL une requête permettant d'obtenir les lettres et leur nombre d'occurrences du corpus écrites par Hugo, classés par nombre d'occurrences décroissant, avec un nombre d'occurrences supérieur à 1000.

```
SELECT symbole, SUM(nombreOccurrences) as nombre_occurrences
FROM caractere as ca
JOIN occurrence as o ON ca.idCar = o.idCar
JOIN corpus as co ON co.idLivre = o.idLivre
WHERE typeCaractere = 'lettre' AND auteur = 'Hugo'
GROUP BY idCar
ORDER BY nombre_occurrences DESC
HAVING nombre_occurrences >= 1000
```

16. Écrire une requête SQL permettant d'obtenir la fréquence d'occurrences (donc comprise entre 0 et 1) de chaque caractère en 'Français'. Plus précisément, la requête devra renvoyer une table à deux attributs : une colonne contenant le symbole de chaque caractère, et une autre colonne contenant le rapport entre le nombre total d'occurrences du caractère et le nombre total de caractères des textes de tout le corpus.

```
SELECT symbole, SUM(nombreOccurrences)/ nbTotal
FROM caractere as ca,
(SELECT SUM(nombreCaracteres) AS nbTotal FROM corpus
WHERE langue = 'Français')
JOIN occurrence as o ON o.idLivre = ca.idCar
JOIN corpus ON o.idLivre = corpus.IdLivre
WHERE langue = 'Français'
GROUP BY ca.idCar
```

- Q25 L'attribut `pi_id`, entier auto-incrémenté, identifie de manière unique le pilote, pas son nom (deux pilotes peuvent avoir le même nom). Il peut donc être choisi comme clef primaire (à attribut unique).

Q26 Une estimation, donc à priori sans utiliser SQL !

Depuis 1950, soit 75 années, une vingtaine de courses par an, entre 40 et 80 tours, et une vingtaine de pilotes soit

$$75 \times 20 \times 80 \times 20 = 2.240.000$$

tours comme estimation du nombre de lignes de la table [Tour](#).

Q27 La requête SQL :

```
SELECT gp_date, ci-nom from Circuit as c
JOIN GrandPrix as g ON g.ci_id = c.ci_id
WHERE ci_pays = 'France'
ORDER BY gp_date ASC
```

Q28 La requête SQL :

```
SELECT ci-nom, pi_nom, sum(to_temps) from Circuit as c
JOIN GrandPrix as g ON g.ci_id = c.ci_id
JOIN Participation as p ON p.gp_id = g.gp_id
JOIN Pilote as pi ON pi.pi_id = p.pi_id
JOIN Tour as t ON t.pa_id = p.pa_id
WHERE p.pa_cla = 1 AND YEARS(gp_date) = 2021
GROUP BY t.pa_id
```

Q29 Renvoie la liste des circuits, pilotes, dates, records du tour, classés alphabétiquement selon le nom du circuit.