

1 Premiers pas avec SQLiteStudio

Nous allons employer durant tout ce TP le logiciel SQLiteStudio qui permet d'interroger une base de données SQL. Le système de base de données sous-jacent est "MySQL", fort répandu dans la gestion des bases de données web. Télécharger la base de données `movie.sqlite` sur la page web du cours, et ajouter-la dans SQLiteStudio. Vous pouvez alors voir la structure de la base de données.

Exercice 1 : Établir un diagramme lié à la base de données "movie" :

- Pour chaque table, donner la liste des attributs (avec leur type), et souligner la clé primaire s'il y en a une (en gras dans la correction).
- Relier les attributs des différentes tables qui permettront de faire des jointures.

2 Requêtes élémentaires

Vous pouvez sauvegarder les requêtes présentes dans l'éditeur dans un fichier `.sql`.

Dans la base de données "movie", cliquez sur l'onglet "SQL". C'est là que vous pourrez effectuer des requêtes SQL sur la base de données choisie. On pourra utiliser la touche **F9** de votre clavier pour exécuter ces requêtes.

Exercice 2 : Proposer des requêtes permettant :

1. De donner l'intégralité de la table personne ?
2. De donner tous les prénoms de la table personne ? Ajouter à la fin de votre requête l'instruction `ORDER BY prenom`. Que se passe-t-il ?
3. De donner les prénoms sans les doublons.
4. De déterminer le nom toutes les personnes dont le prénom est "John".
5. De dresser le produit cartésien (il suffit de mettre les tables dont on veut faire le produit entre virgules dans la clause `FROM`) de la table personne et de film (il n'y a pas nécessairement de lien entre la personne et le film). Observer le nombre de lignes retournées. Était-il prévisible ?
6. De déterminer le titre des films dont le genre est "Drame".

3 Jointures

Exercice 3 : Écrire des requêtes permettant de répondre aux questions suivantes :

1. Quels sont les noms et prénoms des réalisateurs ?
2. Dresser la liste des réalisations (nom, prénom du réalisateur, titre).
3. Quels sont les titres de films projetés après 2002 (inclus) ? On pourra comparer des dates à l'aide du tri lexicographique sur les chaînes de caractères. Par exemple : `"2005-01-01" > "2004-12-31"`.
4. Déterminer le titre des films réalisés par Lars von Trier.
5. Quels sont tous les acteurs qui sont aussi réalisateurs ?
6. Déterminer le titre des films projetés à l'UGC SQY.
7. Quels sont les titres des films dans lesquels Nicole Kidman a joué un rôle et qui ont été projetés à l'UGC SQY ?

Exercice 4 (LEFT JOIN) : À l'aide de `LEFT JOIN`, répondre aux questions suivantes :

1. Quelles sont les personnes n'ayant joué dans aucun film de la base de données ?
2. Quelles sont les personnes n'ayant réalisé aucun film de la base de données ?
3. En déduire une requête trouvant les personnes n'intervenant jamais dans la base de données.
4. Quels sont les films n'ayant aucun acteur dans la base de données ?
5. Quels sont les acteurs jouant dans un film n'existant pas dans la base de données ?

4 Fonctions d'agrégation

Un groupe est un sous-ensemble des lignes d'une table ayant la même valeur pour un même ensemble d'attribut donné. Par exemple, il peut s'avérer intéressant de regrouper les films en fonction du réalisateur pour comptabiliser le nombre de films réalisés par chaque réalisateur. Les modalités de regroupement sont déterminées par la clause GROUP BY suivie des attributs sur lesquels s'effectuent le regroupement. Les fonctions d'agrégation au programme sont les fonctions MIN, MAX, SUM, AVG (calcul de moyenne) et la fonction de comptage COUNT.

Exercice 5 :

1. Déterminer le nombre total de projections pour chaque cinéma (on renverra l'id du cinéma).
2. Modifier la requête précédente pour obtenir non plus l'id du cinéma mais son nom.
3. Dresser la liste des réalisateurs (nom, prénom) en précisant le nombre de films qu'ils ont dirigés. Le résultat doit être trié par ordre alphabétique des noms.
4. Modifier la requête précédente de façon à ajouter l'année de production du premier et du dernier film que chaque réalisateur a réalisé.
5. Dresser la liste des acteurs (nom, prénom) en précisant le nombre de films dans lesquels ils ont joué. Le résultat doit être trié par ordre alphabétique des noms.
6. Quels sont les acteurs ayant joué dans au moins deux films? On utilisera la clause HAVING permettant de sélectionner certains agrégats dans une requête utilisant une agrégation. Cette clause est à placer après la clause GROUP BY.
7. En supposant que les têtes d'affiche des films soient les acteurs recensés par la base de données, quel est le nombre de têtes d'affiche et le réalisateur de chacun de ces films? Pour simplifier on supposera qu'un réalisateur ne joue pas dans son propre film.
8. Quel est le nombre moyen de têtes d'affiches par film?
9. En supposant qu'un film coûte 10 000 000 euros plus 200 000 euros par tête d'affiche, donner, pour chacun des réalisateurs le prix moyen des films qu'il a réalisé.

5 Opérations ensemblistes

Les opérations ensemblistes de réunion, d'intersection, de différence et de division se traitent différemment en MySQL. En effet, il n'y a pas d'instruction explicite dédiée (sauf pour la réunion à l'aide de l'opérateur UNION). C'est pourquoi on utilisera les opérateurs booléens IN et EXISTS permettant de vérifier respectivement si un attribut est dans une table et si une table est vide ou non. Attention le résultat de la sous-requête argument de l'opérateur IN doit contenir autant d'attributs que ceux recherchés.

Exercice 6 :

1. A l'aide de l'opérateur IN ou EXISTS, déterminer tous les réalisateurs.
2. Déterminer les réalisateurs ayant réalisé des drames ou des épouvantes en utilisant l'opérateur UNION.
3. Déterminer tous les acteurs n'ayant jamais joué dans un film dramatique.

6 Recherche de motifs

On peut effectuer des recherches de motifs sur chaînes de caractères à l'aide de l'instruction LIKE.

Sa syntaxe est de la forme :

```
SELECT <...> FROM <...> WHERE <attribut/chaîne> LIKE <motif>
```

En théorie : un motif est une description syntaxique d'un ensemble de chaînes de caractères. En pratique, un motif est une forme (nous considérerons des chaînes de caractères en général) du type "l'expression contient un e" etc. Pour les exprimer, on utilise souvent un "caractère joker" : % qui représente n'importe quelle chaîne de caractère possible. Par exemple les motifs suivant signifient :

- 'ja%son' : toutes les chaînes de caractères commençant par "ja", contenant une chaîne de caractères ensuite puis terminant par "son". Cela recouvre les chaînes "janson", "jason", "jamuson" etc
- '%a%' : toutes les chaînes contenant un 'a'.

Par défaut, les motifs ne tiennent pas compte de la casse typographique.

Exercice 7 :

1. Quelles sont les personnes (nom, prénom) dont le prénom contient le caractère 's', sans tenir compte de la casse (capitale ou minuscule) ?
2. Quels sont les prénoms des personnes qui terminent par le caractère 's', sans tenir compte de la casse ?
3. Quels sont les prénoms des personnes qui ne contiennent pas le caractère 'e', sans tenir compte de la casse ?
4. Quels sont les noms des personnes qui contiennent les caractères 'a' et 'l' dans cet ordre (sans tenir compte de la casse) ?
5. Même question lorsque l'ordre des caractères n'a plus d'importance.
6. Quels sont les noms des personnes qui contiennent les sous-chaînes 'an' ou 'on' ?
7. Quels sont les titres de films qui contiennent au moins trois caractères 'e' ?
8. Quels sont les titres des films qui contiennent le caractère 'b' ainsi que le caractère 'e' (sans tenir compte de la casse), mais qui ne finissent pas par 'e' ?

7 SQL Murder Mystery

Exercice 8 : Un crime a eu lieu, et le détective a besoin de votre aide. Le détective vous avait donné un dossier, mais vous l'avez perdu. Vous vous souvenez simplement que le crime était un meurtre qui a eu lieu le 15 janvier 2018 à SQL City.

Vous avez accès à une gigantesque base de données accessible sur la page web du cours, qui vous sera utile pour résoudre ce mystère.

1. Récupérer le rapport de la police concernant ce meurtre.
2. Utiliser votre savoir-faire en SQL pour trouver le meurtrier.

Vous pouvez tester si vous avez résolu l'affaire avec les commandes suivantes :

```
1 INSERT INTO solution VALUES (1,"Prénom Nom du meurtrier") ;
2 SELECT value FROM solution ;
```

8 Un RPG en SQL

Exercice 9 : Vous avez survécu à un crash d'avion, et vous vous retrouvez coincé sur une île : SQL Island.

1. Aller sur le site : <https://sql-island.informatik.uni-kl.de/>.
2. Trouver un moyen de vous échapper de l'île.