

Requêtes SQL sur une base de films

Lycée Jean Perrin

Erwan DEMERVILLE

Sommaire

| | | |
|----------|---|----------|
| 1 | Introduction | 2 |
| 2 | Partie 1 : Etablir un schéma relationnel | 2 |
| 3 | Partie 2 : Requêtes SQL | 2 |

1 Introduction

SQLite est une **bibliothèque** écrite en langage **C** qui propose un moteur de **base de données relationnelle** accessible par le langage **SQL**.

Le **SQL** est un langage **puissant** qui permet de nombreuses manipulations de **données**.

Il permet notamment d'effectuer des opérations de recherche, d'ajout, de modification ou de suppression de données dans des bases de données relationnelles.

Durant ce travail, *qui sera ramassé à la fin du cours*, vous travaillerez sur une **base de données de films**.

Le but est de tester vos connaissances du langage **SQL** à travers quelques requêtes basiques permettant d'extraire des données de cette base.

2 Partie 1 : Etablir un schéma relationnel

On dispose d'un fichier `movies.csv` (simplifié) contenant des **données sur de nombreux films**.

Ouvrez ce fichier sur **LibreOffice Calc** ou **Excel**.

| | A | B | C | D | E | F | G | |
|---|--|------|-------|-------------------|----------------------------|-------|-----------------|---------|
| 1 | titre | pays | duree | realisateur | realisateur_facebook_likes | annee | acteur1 | acteur2 |
| 2 | Avatar | USA | 178 | James Cameron | 0 | 2009 | CCH Pounder | Joel Di |
| 3 | Pirates of the Caribbean: At World's End | USA | 169 | Gore Verbinski | 563 | 2007 | Johnny Depp | Orland |
| 4 | Spectre | UK | 148 | Sam Mendes | 0 | 2015 | Christoph Waltz | Rory K |
| 5 | The Dark Knight Rises | USA | 164 | Christopher Nolan | 22000 | 2012 | Tom Hardy | Christi |
| 6 | John Carter | USA | 132 | Andrew Stanton | 475 | 2012 | Daryl Sabara | Saman |
| 7 | Spider-Man 3 | USA | 156 | Sam Raimi | 0 | 2007 | J.K. Simmons | James |
| 8 | Tangled | USA | 100 | Nathan Greno | 15 | 2010 | Brad Garrett | Donna |

Figure 1: Capture_csv

Question : On souhaite établir un **schéma relationnel** permettant de représenter ces données de manière plus intéressante.

Autrement dit, séparer les données en **plusieurs tables** reliées entre elles.

On souhaite avoir :

- Une table **acteurs** qui contiendra le **nom des acteurs** et leur **nombre de likes** sur Facebook.
- Une table **realisateurs** qui contiendra le **nom des réalisateurs** et leur **nombre de likes** sur Facebook.
- Une table **films** qui contiendra toutes les autres données sur les films, **réliée aux deux autres tables**.

Réalisez le schéma relationnel en indiquant bien les **clés primaires** et **clés étrangères**.

(Il se peut qu'un attribut puisse être à la fois une *clé primaire* et une *clé étrangère*...)

3 Partie 2 : Requêtes SQL

Une **base de données movies** est fournie dans le dossier.

Chargez-la sur **DB Browser for SQLite**.

Sur un document texte, écrivez les requêtes permettant d'obtenir :

- Le **titre** et le **realisateur** de tous les **films**.
- Même requête mais en mettant le réalisateur en **majuscules**.
- Toutes les données (de la table **films**) sur le film *Halloween*.
- Tous les **films** réalisés par *John Carpenter*.
- Le **nombre de films** réalisés par *Steven Spielberg*.
- Les **films** réalisés par *Tim Burton* dans lesquels *Johnny Depp* est l'**acteur 1**.
- Les **acteurs** ayant plus de *10000 likes*, triés par **ordre croissant** de likes.
- Le **nombre d'acteurs** ayant *plus de 500 likes* ET *moins de 1000 likes*.
- Le **nombre moyen de likes**, renommé *nb_moyen*, des **acteurs**.
- Les **realisateurs** groupés par *nombre de likes*, en ne gardant QUE les **groupes de plus de 1 élément**.

- La **somme** des *likes* des **réalisateurs**.
- Les **réalisateurs** dont le *nombre de likes est pair*. Vous pouvez avoir besoin d'utiliser le modulo %.
- Les **films** dont le réalisateur a *plus de 5000 likes*, par **ordre décroissant** de likes.

Quelques requêtes **bonus** (à faire s'il vous reste du temps) :

- Les **noms des acteurs** qui sont **acteur1** d'au moins 1 film. Chaque acteur ne doit apparaître qu'une seule fois.
- Les **réalisateurs** ayant **réalisé plus de 10 films**.
- **Recettes moyennes** des **films** par **réalisateur**, du la plus petite recette à la plus grande.
- Le **nombre de films** par année, du plus grand nombre au plus petit.