

Introduction à SQL – Atelier Pratique

Encadré par : ALI Nazim

Objectifs de l'atelier

Au cours de cet atelier, vous allez vous familiariser avec les bases de données relationnelles et apprendre à manipuler les données avec SQL, un langage fondamental pour tout développeur ou analyste. Nous utiliserons **MySQL**, une solution libre et populaire de gestion de base de données.

Mise en place de l'environnement

Ouvrez un terminal et installez MySQL avec la commande suivante(Cas de Linux) :

```
sudo apt-get install mysql-client mysql-server
```

Pendant l'installation, un mot de passe vous sera demandé pour l'utilisateur `root`. Conservez-le précieusement.

Une fois l'installation terminée, connectez-vous à MySQL :

```
mysql -u root -p
```

Créez ensuite une base de données nommée `movies` :

```
CREATE DATABASE movies;  
USE movies;
```

Importez le fichier `movies.sql` fourni, via la commande :

```
source <chemin/vers/movies.sql>
```

Vous pouvez désormais commencer à exécuter vos requêtes SQL.

Exercices pratiques

Chaque exercice correspond à une compétence SQL que vous devrez maîtriser. Les fichiers doivent être enregistrés sous le format `ex_0X/ex_0X.sql`.

Exercice 01 (1 point)

Fichier à rendre : `ex_01/ex_01.sql`

Écrire une requête qui affiche la liste de toutes les tables de la base de données.

Exercice 02 (1 point)

Fichier à rendre : `ex_02/ex_02.sql`

Écrire une requête qui affiche la description de la table **movies**.

Exercice 03 (1 point)

Fichier à rendre : `ex_03/ex_03.sql`

Écrire une requête qui affiche la date actuelle dans une colonne "Date" au format "AAAA-MM-JJ".

Exercice 04 (1 point)

Fichier à rendre : ex_04/ex_04.sql

Écrire une requête qui affiche les colonnes **title** et **summary** de tous les films triés par ordre alphabétique.

Exercice 05 (1 point)

Fichier à rendre : ex_05/ex_05.sql

Écrire une requête qui affiche le **nom** de tous les genres de la table **genres** en majuscules. La colonne sera nommée "NAME OF ALL THE GENRES".

Exercice 06 (1 point)

Fichier à rendre : ex_06/ex_06.sql

Écrire une requête qui affiche le **titre** des 42 derniers films de la table **movies**. La colonne sera nommée "Title of the last 42 movies". Les résultats doivent être triés par ID décroissant.

Exercice 07 (1 point)

Fichier à rendre : ex_07/ex_07.sql

Écrire une requête qui affiche le **nom** de l'abonnement le plus cher de la table **subscriptions**, ainsi que son prix. Les colonnes seront nommées respectivement : "Name of the most expensive subscription" et "Price".

Exercice 08 (1 point)

Fichier à rendre : `ex_08/ex_08.sql`

Écrire une requête qui affiche les titres des films dont le genre est "action" ou "romance".

Exercice 09 (1 point)

Fichier à rendre : `ex_09/ex_09.sql`

Écrire une requête qui affiche la durée du film le plus court en minutes. Les films avec une durée NULL ou 0 ne doivent pas être pris en compte. La colonne sera nommée "Duration of the shortest movie".

Exercice 10 (1 point)

Fichier à rendre : `ex_10/ex_10.sql`

Écrire une requête qui affiche l'ID des films dont le titre contient la chaîne "tard" (insensible à la casse). La colonne sera nommée "Identifier".

Exercice 11 (1 point)

Fichier à rendre : `ex_11/ex_11.sql`

Écrire une requête qui affiche le nombre de films dont le titre se termine par "tion" (insensible à la casse). La colonne sera nommée "Number of movies ending with 'tion'".

Exercice 12 (1 point)

Fichier à rendre : ex_12/ex_12.sql

Écrire une requête qui affiche le nombre total de films du genre "western" et dont les producteurs sont "tartan movies" ou "lionsgate uk". La colonne sera nommée "Number of 'western' movies".

Exercice 13 (1 point)

Fichier à rendre : ex_13/ex_13.sql

Écrire une requête qui affiche le numéro et le nom des salles ayant plus de 0 places et qui ne sont pas au premier étage (premier étage = étage 1). Les colonnes seront nommées "Room numbers" et "Room names".

Exercice 14 (1 point)

Fichier à rendre : ex_14/ex_14.sql

Écrire une requête qui affiche le nombre de films dont le titre commence par "eX" (sensible à la casse). La colonne sera nommée "Number of movies that starts with 'eX'".

Exercice 15 (1 point)

Fichier à rendre : ex_15/ex_15.sql

Écrire une requête qui affiche la durée moyenne d'un film arrondie à 2 décimales. La colonne sera nommée "Average duration".

Exercice 16 (1 point)

Fichier à rendre : `ex_16/ex_16.sql`

Écrire une requête qui affiche le mois de naissance en anglais du 42ème au 84ème membre (le 42ème et le 84ème inclus). La colonne sera nommée "month of birth".

Exercice 17 (1 point)

Fichier à rendre : `ex_17/ex_17.sql`

Écrire une requête qui affiche le titre du film le plus long. La colonne sera nommée "Title of the longest movie".

Exercice 18 (1 point)

Fichier à rendre : `ex_18/ex_18.sql`

Écrire une requête qui affiche le nom de famille suivi d'un tiret, suivi du prénom de chaque membre de la table **profiles**. La première lettre du nom et du prénom sera en majuscule. Les membres doivent être affichés du plus jeune au plus âgé. La colonne sera nommée "Full name".

Exercice 19 (1 point)

Fichier à rendre : `ex_19/ex_19.sql`

Écrire une requête qui affiche le titre des films dont l'ID est 21, 87, 263, 413 ou 633. La colonne sera nommée "Movie title".

Exercice 20 (1 point)

Fichier à rendre : ex_20/ex_20.sql

Écrire une requête qui affiche le nombre de films produits par année. L'année ne doit pas être 0. Les résultats doivent être triés par année de production décroissante. Les colonnes seront nommées "Number of movies" et "Year of production".

Bonne chance