

TP : Création et gestion d'une base de données pour un blog

Objectifs

- Modéliser et manipuler une base de données adaptée à un blog.
- Appliquer des requêtes SQL pour insérer, lire, modifier des données.
- Comprendre l'indexation, l'optimisation des requêtes et l'utilisation des triggers.
- Assurer la sécurité de la base de données.

Scénario

Vous êtes chargé de concevoir et de gérer une base de données pour un site de blog. La base de données doit gérer les utilisateurs, les articles, les commentaires, et les catégories des articles.

Étapes et défis

1. Modélisation de la base de données (20 min)

1. Utilisez MySQL Workbench pour modéliser les tables suivantes :
 - `users` : `id`, `name`, `email`, `password`, `created_at`.
 - `categories` : `id`, `name`.
 - `articles` : `id`, `title`, `content`, `user_id`, `category_id`, `created_at`.
 - `comments` : `id`, `article_id`, `user_id`, `content`, `created_at`.
2. Ajoutez les relations suivantes :
 - `users` et `articles` (1:N).
 - `categories` et `articles` (1:N).
 - `articles` et `comments` (1:N).
3. Exportez le script SQL pour créer les tables.

TP : Création et gestion d'une base de données pour un blog

2. Création et gestion de la base de données (10 min)

1. Créez une base de données nommée `blog`.
2. Importez le script généré pour créer les tables.

3. Manipulation des données (30 min)

1. Insérez plusieurs utilisateurs, catégories, articles et commentaires comme exemples.
2. Modifiez le contenu d'un article avec une nouvelle information.
3. Supprimez un commentaire spécifique.
4. Supprimez une table entière si elle n'est plus nécessaire.

4. Lecture et filtrage de données (40 min)

1. Écrivez des requêtes SQL pour afficher les articles avec le nom de l'auteur et la catégorie correspondante.
2. Filtrez les commentaires par article, triés par date de création.
3. Trouvez l'utilisateur ayant publié le plus d'articles.

5. Indexation des tables (15 min)

1. Ajoutez un index UNIQUE sur la colonne `email` de la table `users`.
2. Ajoutez un index composite sur les colonnes `user_id` et `created_at` de la table `articles`.
3. Expliquez les avantages des index pour accélérer les recherches.

TP : Création et gestion d'une base de données pour un blog

6. Optimisation des requêtes (20 min)

1. Analysez le plan d'exécution d'une requête complexe avec EXPLAIN.
2. Identifiez les problèmes de performance et proposez des solutions.

7. Triggers et contraintes (30 min)

1. Créez un trigger pour mettre à jour une colonne `updated_at` lors d'une modification d'article.
2. Ajoutez une contrainte de clé étrangère entre `comments.article_id` et `articles.id`.

8. Sécurité et privilèges (20 min)

1. Créez un utilisateur SQL avec des droits limités pour lire, insérer et mettre à jour les données.
2. Révoquez ses privilèges si besoin.

9. Fonction, vues et tables temporaires (30 min)

1. Créez une fonction utilisateur pour calculer le nombre de commentaires d'un article.
2. Créez une vue affichant les articles avec leur nombre de commentaires.
3. Utilisez une table temporaire pour les articles ayant plus de 10 commentaires.

10. Transactions et requêtes préparées (30 min)

1. Utilisez une transaction pour insérer un article et ses commentaires de manière atomique.
2. Créez une requête préparée pour insérer des commentaires.