

Environnement

- MySQL Server
- MySQL Workbench

Directive

- Sauvegardez vos réponses texte et vos instructions SQL dans un fichier au `contraintes_ex3_2_3.sql`

Exercice 1 - Renommer une base de données

Discutez de la façon de renommer votre base de données `module02_exercices_1_2` en un autre nom.

- Peut-on renommer la base de données en utilisant l'instruction `ALTER DATABASE` ? Indiquez où vous avez trouvé cette réponse
- Quelles sont les modifications que vous pouvez apporter à une base de données avec la précédente commande ?
 - Listez les modifications possibles
 - Expliquez la différence entre `CHARACTER SET` et `COLLATION`
- En vous basant sur vos recherches, décrivez en vos mots la meilleure solution pour renommer une base de données.
- Testez votre solution en renommant votre base de données `module02_exercices_1_2` en `module03_nouveau`.
- Validez que votre structure est la même.
- Validez que vous avez bien toutes vos données.

Exercice 2 - Création de contraintes

- Partez du code SQL (`module03_ex2.sql`):
- À l'aide de la commande `ALTER TABLE NomTable ADD [CONSTRAINT [NomContrainte]] CHECK(Contrainte)`, Ajoutez la contrainte nommée `CHK_TemperatureSupMoins30` :
 - Qui valide que la colonne `MntTemperature` a une valeur supérieure à 60
 - Est-ce que cela fonctionne ?
 - Pourquoi ?

- Pour la suite de l'exercice, modifiez la validation pour que la température doive simplement être supérieure à -30
- À l'aide de la commande `ALTER TABLE`, modifiez la colonne `MntJour` pour qu'elle n'accepte que les valeurs du jour de la semaine (de 'Lun' à 'Dim'). Pour cela, modifiez le type de la colonne pour 'ENUM'
- À l'aide de la commande `ALTER TABLE`, ajoutez la contrainte `CHK_VitessePositive` qui valide que la vitesse est positive.
- Créez des requêtes d'insertion qui valident que vos contraintes fonctionnent : certaines doivent donc échouer, d'autres réussir.
- Analysez les messages d'erreur qui sont renvoyés.
- Modifiez la commande de la création de la table afin d'y intégrer directement les contraintes précédentes.

Exercice 3 - Création de contraintes

- Partez du code SQL (`module03_ex3.sql`):
- Récrire la définition de la table `Utilisateur` pour :
 - Ajouter une clé primaire (`UTL_ID`). Elle est de type `INT` et s'incrémente automatiquement;
 - Le courriel doit être unique;
 - Ajouter une contrainte de table (`CHK_AGE`) qui vérifie l'âge de l'utilisateur. Il doit être `>21`;
 - Faites des tests avec des ajouts.
- À l'aide de la commande `ALTER TABLE NomTable ADD` :
 - Ajouter une colonne (`UTL_SEXE`) dans la table `Utilisateur`. Ce champs est un `char(1)` n'est pas être `NULL` et par défaut = 'I';
 - Ajouter une contrainte (`CHK_GENRE`) pour valider que le champ n'accepte que les valeurs suivantes : M, F, X et I.
 - Détruire la contrainte (`CHK_AGE`).
 - Ajouter la contrainte (`CHK_AGE`) pour que l'on peut accepter les utilisateurs de 18 et plus.
 - Tester votre solution avec plusieurs ajouts.

Personnes ayant contribué à la rédaction de ce document :

- Jean-Pierre Duchesneau : version initiale, révisions
- Pierre-François Léon : révisions
- Ali Awdé : révisions