**TP 1 : Création et Gestion de Tables avec MySQL**

**Objectifs :**

* Concevoir et gérer des tables en utilisant le langage de définition de données (LDD).
* Appliquer des contraintes pour garantir l'intégrité des données.
* Modifier et manipuler la structure des tables.

**Exercice 1 : Création de tables**

1. Créez une base de données appelée GestionEmployes et sélectionnez-la.
2. Créez une table Employes avec les colonnes suivantes :
   * IdEmploye : Identifiant unique (clé primaire).
   * Nom : Nom de l'employé (obligatoire).
   * Prenom : Prénom de l'employé (obligatoire).
   * Poste : Poste occupé par l'employé.
   * Salaire : Salaire mensuel de l'employé, obligatoire.
3. Créez une table Departements avec les colonnes suivantes :
   * IdDepartement : Identifiant unique (clé primaire).
   * NomDepartement : Nom du département, unique.
   * Budget : Budget du département.
   * Ajoutez une contrainte pour que le budget soit supérieur à 10 000.

**Exercice 2 : Gestion des contraintes**

1. Ajoutez une clé étrangère dans la table Employes pour lier la colonne IdDepartement à la table Departements.
2. Ajoutez une contrainte CHECK à la table Employes pour que le salaire soit supérieur à 3 000.

**Exercice 3 : Modification des tables**

1. Ajoutez une nouvelle colonne DateEmbauche (DATE) à la table Employes.
2. Modifiez la colonne Salaire pour augmenter sa précision à DECIMAL(12, 2).
3. Supprimez la colonne Poste de la table Employes.

**Exercice 4 : Suppression et renommage des tables**

1. Renommez la table Departements en Services.
2. Supprimez la table Employes de la base de données.

**TP 2 : Gestion des Données et Contraintes dans une Base de Données MySQL**

**Objectifs :**

* Maîtriser la gestion des tables et des contraintes dans MySQL.
* Pratiquer les opérations de modification et de suppression sur des tables.
* Renforcer la compréhension des types de données et des contraintes.

**Scénario :**

Vous êtes chargé de gérer une base de données pour une bibliothèque. Les tables doivent stocker des informations sur les livres, les auteurs et les utilisateurs qui empruntent les ouvrages.

**Exercice 1 : Création des tables**

1. Créez une base de données appelée Bibliotheque et utilisez-la.
2. Créez une table Livres avec les colonnes suivantes :
   * IdLivre : Identifiant unique du livre (clé primaire).
   * Titre : Titre du livre (obligatoire).
   * AuteurId : Référence vers l'auteur du livre (clé étrangère).
   * AnneePublication : Année de publication (obligatoire).
   * Quantite : Nombre de copies disponibles, avec une valeur par défaut de 1.
3. Créez une table Auteurs avec les colonnes suivantes :
   * IdAuteur : Identifiant unique de l'auteur (clé primaire).
   * Nom : Nom de l'auteur (obligatoire).
   * Pays : Pays d'origine de l'auteur.
4. Créez une table Emprunts avec les colonnes suivantes :
   * IdEmprunt : Identifiant unique de l'emprunt (clé primaire).
   * UtilisateurId : Référence vers l'utilisateur (clé étrangère).
   * LivreId : Référence vers le livre emprunté (clé étrangère).
   * DateEmprunt : Date de l'emprunt, obligatoire.
   * DateRetour : Date prévue de retour, obligatoire.

**Exercice 2 : Gestion des contraintes**

1. Ajoutez une contrainte à la table Livres pour que l’année de publication soit supérieure à 1800.
2. Ajoutez une contrainte NOT NULL sur la colonne Pays de la table Auteurs.
3. Configurez une contrainte de clé étrangère sur la table Emprunts pour que les emprunts soient supprimés automatiquement lorsque l'utilisateur ou le livre correspondant est supprimé.

**Exercice 3 : Modification des tables**

1. Ajoutez une colonne Categorie (VARCHAR) à la table Livres pour indiquer la catégorie du livre.
2. Modifiez la table Emprunts pour ajouter une colonne Statut (VARCHAR), avec une valeur par défaut de En cours.
3. Supprimez la colonne Pays de la table Auteurs.

**Exercice 4 : Nettoyage et suppression**

1. Videz les données de la table Emprunts sans supprimer sa structure.
2. Supprimez la table Auteurs en s'assurant qu'aucune contrainte n'empêche cette action.

**Étude de cas 1 : Gestion d'une entreprise de location de voitures**

Une entreprise de location de voitures souhaite digitaliser la gestion de ses opérations. Voici les grandes lignes des besoins :

1. **Gestion des véhicules :**
   * L'entreprise possède plusieurs types de véhicules (voitures, utilitaires, motos).
   * Chaque véhicule a des informations sur son état (disponible, en location, en maintenance).
2. **Gestion des clients :**
   * Les clients peuvent louer plusieurs véhicules, mais un véhicule ne peut être loué qu’à un seul client à la fois.
   * Les informations des clients doivent être enregistrées pour chaque location (identité, contact, etc.).
3. **Gestion des contrats de location :**
   * Chaque contrat de location contient des informations comme la durée de location, le coût, et les pénalités en cas de retard.
   * Un client peut prolonger une location existante ou ajouter de nouveaux véhicules à son contrat.
4. **Suivi des paiements :**
   * Les clients doivent régler leur contrat en plusieurs paiements ou en une seule fois.
   * L’entreprise veut garder une trace des montants payés et des éventuels montants dus.
5. **Suivi de l’entretien des véhicules :**
   * Chaque véhicule doit passer des entretiens réguliers.
   * L’historique des maintenances, les réparations effectuées et les coûts associés doivent être tracés.

**Consignes pour les stagiaires :**

1. Analysez l'étude de cas et identifiez les entités principales (tables) nécessaires.
2. Déterminez les attributs (champs) de chaque entité et leur type de données.
3. Identifiez les relations entre les entités (ex. : relations un-à-plusieurs, plusieurs-à-plusieurs).
4. Définissez les contraintes à appliquer (clé primaire, clé étrangère, unique, not null, etc.).

**Tâches à réaliser :**

**Étape 1 : Création de la structure**

1. Concevez la base de données en identifiant toutes les entités nécessaires.
2. Définissez les champs pour chaque table et choisissez les types de données appropriés.
3. Créez un diagramme entité-association (optionnel).

**Étape 2 : Implémentation en MySQL**

1. Créez la base de données et les tables correspondantes.
2. Ajoutez des contraintes pour assurer l’intégrité des données (clés primaires, clés étrangères, contraintes de validité).