

## Contrôle continu N°1

### Exercice 1 :

On vous donne le schéma relationnel relatif à la gestion d'une bibliothèque :

*Ouvrage* (code-ouvrage, titre, nombre-exemplaires, #code-éditeur) ;

*Auteur* (code-auteur, nom-auteur, grade, spécialisation) ;

*Editeur* (code-éditeur, nom-éditeur, adresse) ;

*Ecrit* (#code-ouvrage, #code-auteur) ;

### Questions :

1. Donner le code MySQL qui permet de créer la table *Editeur*.

*create table Editeur (Code\_editeur char(5) primary key,  
nom\_editeur varchar(20) not null,  
adresse varchar(30) not null);*

2. Après création de la table *Editeur*, on suppose que vous avez exécuté la requête de création de la table *Ecrit*. Comment le SGBD répond à cette requête ? justifier votre réponse.

*La table Editeur ne peut être créée.*

*Car : la table Editeur contient des clés étrangères qui font référence aux tables Ouvrage et Auteur qui ne sont pas encore créées.*

3. Donner l'ordre de création du reste de la base de données.

*La table Ecrit ne peut être créée sans création des tables Ouvrage et Auteur, ses dernières sont indépendantes entre elles, donc :*

*Ordre1 : Ouvrage, Auteur, Ecrit. Ou bien*

*Ordre2 : Auteur, Ouvrage, Ecrit.*

4. Donner la réponse du SGBD en exécutant les commandes suivantes, on suppose que toutes les valeurs citées par les requêtes existent dans les tables interrogées :

*Requête 1 :*

*Delete from Ecrit where code-ouvrage='BDD105' ;*

*La requête s'exécute avec succès. La réponse du SGBD est Query OK. Et la ligne dont le code-ouvrage = 'BDD105' est supprimée.*

*Requête2 :*

UPDATE *Ecrit* set *Code-auteur*='AUT 01' where *code-ouvrage*='MATH50' ;

*La réponse dépend de la présence de la valeur du code-auteur qui est 'AUT01' dans la table de référence Auteur.*

*Si la valeur 'AUT01' est présente dans Auteur, la requête s'exécute avec succès.*

*Si la valeur 'AUT01' est absente dans la table Auteur, la requête ne s'exécute pas.*

*Requête3 :*

UPDATE *Auteur* set *Code-auteur*='AUT20' where *spécialisation*='Informatique' ;

*Si la valeur 'AUT01' est absente dans la table Auteur, la requête ne s'exécute pas.*

*Au niveau de la table mère « Auteur » la requête s'exécute avec succès.*

*Au niveau de la table fille « Ecrit », le résultat dépend de la clause ON UPDATE de la table Ecrit.*

5. Donner une interprétation de la table *Ecrit*. (Quelles données elle contient?)

*La table Ecrit contient les codes des ouvrages et leurs auteurs, un ouvrage peut être écrit par plusieurs auteurs et un auteur peut écrire plusieurs ouvrages.*

## **Exercice 2:**

Soit le modèle relationnel relatif à la gestion de patients dans un hôpital :

Médecin (Id-médecin, Nom, prénom)

Consultation (Id-consultation, date-consultation, #id-médecin, # id-patient)

Patient (Id-patient, Nom-patient, Prénom)

RDV (Numéro-RDV, #Id-patient, Date-prise, date-programmée-consultation)

Donner-lieu (#Numéro-RDV, # Id-Consultation)

**Questions :** Ecrire les requêtes SQL qui répondent aux questions suivantes :

1. Lister les noms et les prénoms des patients programmés pour une consultation le 30/06/2019.

*Select nom-patient, prénom-patient from Patient where id-patient in  
(select id-patient from RDV where date-programmée-consultation = '30/06/2019') ;*

2. Lister la date de prise de RDV et le numéro de RDV des consultations programmées le 05/07/2019.

*Select numero-rdv, date-prise from RDV where date-programmée-consultation='05/07/2019' ;*

3. Lister les médecins (Id médecin, nom et prénom) qui ont travaillé le 05/07/2019.

*Select \* from médecin where id-medecin in (select id-medecin from consultation where date-consultation = '05/07/2019') ;*

4. Donner le nom et le prénom du médecin qui a consulté le patient qui a pris le RDV numéro 12000.

*Select nom-médecin, prénom -médecin from médecin where id medecin=*

*(select id-médecin from consultation where id-patient =(select id-patient from RDV where numéro-RDV=12000)) ;*

**BON COURAGE**