

SUJETS DE BAC: BASES DE DONNÉES

SUJET METROPOLE 1

Dans cet exercice, on pourra utiliser les mots clés suivants du langage SQL :

`SELECT, FROM, WHERE, JOIN, ON, INSERT INTO, VALUES, MIN, MAX, OR, AND.`

Les fonctions d'agrégation `MIN(propriete)` et `MAX(propriete)` renvoient, respectivement, la plus petite et la plus grande valeur de l'attribut `propriete` pour les enregistrements sélectionnés.

Des acteurs ayant joué dans différentes pièces de théâtre sont recensés dans une base de données `Theatre` dont le schéma relationnel est donné ci-dessous :

- `Piece (idPiece, titre, langue)`
- `Acteur (idActeur, nom, prenom, anneeNaiss)`
- `Role (#idPiece, #idActeur, nomRole)`

Dans ce schéma, les clés primaires sont soulignées et les clés étrangères sont précédées du symbole #.

L'attribut `idPiece` de la relation `Role` est une clé étrangère faisant référence à l'attribut `idPiece` de la relation `Piece`.

L'attribut `idActeur` de la relation `Role` est une clé étrangère faisant référence à l'attribut `idActeur` de la relation `Acteur`.

Tous les attributs dont le nom est préfixé par `id` sont des nombres entiers ainsi que l'attribut `anneeNaiss`. Les autres attributs sont des chaînes de caractères.

1. Expliquer pourquoi il n'est pas possible d'insérer une entrée dans la relation `Role` si les relations `Piece` et `Acteur` sont vides.
2. Dans la pièce « Le Tartuffe », l'acteur Micha Lescot a joué le rôle de Tartuffe. L'identifiant de Micha Lescot est 389761 et celui de cette pièce est 46721. Écrire une requête SQL permettant d'ajouter ce rôle dans la table (ou relation) `Role`.
3. Expliquer ce que fait la requête SQL suivante.

```
UPDATE Piece
SET langue = "Anglais"
WHERE langue = "Américain" OR langue = "Britannique";
```

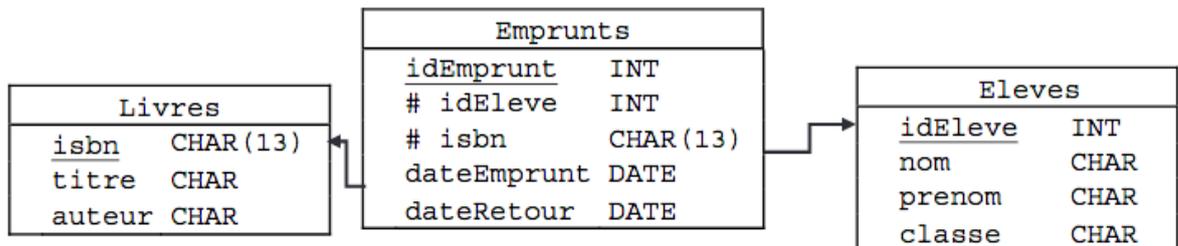
4. Pour chacun des quatre items suivants, écrire une requête SQL permettant d'extraire les informations demandées.
 - a. Le nom et prénom des artistes nés après 1990.
 - b. L'année de naissance du plus jeune artiste.
 - c. Le nom des rôles joués par l'acteur Vincent Macaigne.
 - d. Le titre des pièces écrites en Russe dans lesquelles l'actrice Jeanne Balibar a joué.

SUJET METROPOLE 2

L'énoncé de cet exercice utilise les mots du langage SQL suivants :

SELECT FROM, WHERE, JOIN ON, INSERT INTO VALUES, UPDATE, SET, DELETE, COUNT, AND, OR.

On considère dans cet exercice une gestion simplifiée des emprunts des ouvrages d'un CDI. La base de données utilisée sera constituée de trois relations (ou tables) nommées *Eleves*, *Livres* et *Emprunts* selon le schéma relationnel suivant :



Dans ce schéma relationnel, un attribut souligné indique qu'il s'agit d'une clé primaire. Le symbole # devant un attribut indique qu'il s'agit d'une clé étrangère et la flèche associée indique l'attribut référencé. Ainsi, l'attribut *idEleve* de la relation *Emprunts* est une clé étrangère qui fait référence à la clé primaire *idEleve* de la relation *Eleves*.

1. Expliquer pourquoi le code SQL ci-dessous provoque une erreur.

```
INSERT INTO Eleves VALUES (128, 'Dupont', 'Jean', 'T1') ;
INSERT INTO Eleves VALUES (200, 'Dupont', 'Jean', 'T1') ;
INSERT INTO Eleves VALUES (128, 'Dubois', 'Jean', 'T2') ;
```

2. Dans la définition de la relation *Emprunts*, qu'est-ce qui assure qu'on ne peut pas enregistrer un emprunt pour un élève qui n'a pas encore été inscrit dans la relation *Eleves* ?
3. Écrire une requête SQL qui renvoie les titres des ouvrages de Molière détenus par le CDI.

4. Décrire le résultat renvoyé par la requête ci-dessous.

```
SELECT COUNT(*)
FROM Eleves
WHERE classe = 'T2' ;
```

5. Camille a emprunté le livre *Les misérables*. Le code ci-dessous a permis d'enregistrer cet emprunt.

```
INSERT INTO Emprunts
VALUES (640, 192, '9782070409228', '2020-09-15', NULL);
```

Camille a restitué le livre le 30 septembre 2020.

Recopier et compléter la requête ci-dessous de manière à mettre à jour la date de retour dans la base de données.

```
UPDATE Emprunts SET ..... WHERE .....
```

6. Décrire le résultat renvoyé par la requête ci-dessous.

```
SELECT DISTINCT nom, prenom
FROM Eleves, Emprunts
WHERE Eleves.idEleve = Emprunts.idEleve
AND Eleves.classe = 'T2' ;
```

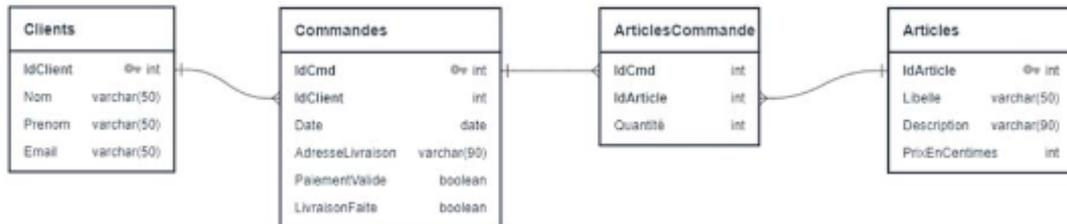
7. Écrire une requête SQL qui permet de lister les noms et prénoms des élèves qui ont emprunté le livre *Les misérables*.

POLYNESIE

Cet exercice traite principalement du thème « traitement de données en tables et bases de données ». Nous nous interrogerons dans cet exercice sur la modélisation et l'utilisation des données nécessaires aux fonctionnements de sites de vente en ligne.

Partie A : Modèle relationnel

Voici le modèle relationnel qui sera utilisé dans cet exercice :



Lecture :

- un symbole  identifie une clé primaire
- un trait  entre deux attributs indique qu'ils doivent partager les mêmes valeurs. Ces valeurs sont prises de manière unique dans la relation du côté + et zéro, une ou plusieurs fois du côté < à condition d'être présente du côté +.

1. Donner les clés primaires des relations Clients et Articles.
2. Les commandes ci-dessous ont été utilisées pour créer ces deux relations. Donner le domaine (c'est-à-dire le type) des attributs Email et Quantité.

```
CREATE TABLE Clients (  
  IdClient INT PRIMARY KEY,  
  Nom VARCHAR(50),  
  Prenom VARCHAR(50),  
  Email VARCHAR(50)  
);  
  
CREATE TABLE ArticlesCommande (  
  IdCmd INT,  
  IdArticle INT,  
  Quantite INT,  
  PRIMARY KEY (IdCmd,IdArticle)  
  FOREIGN KEY(IdArticle) REFERENCES Articles(IdArticle)  
);
```

3. En vous inspirant des commandes ci-dessus, recopier et compléter la commande suivante qui permet de créer la relation Commandes en précisant sa clé étrangère.

```
CREATE TABLE Commandes (  
  IdCmd INT PRIMARY KEY,  
  IdClient INT,  
  Date DATE,  
  AdresseLivraison VARCHAR(90),  
  PaiementValide BOOLEAN,  
  LivraisonFaite BOOLEAN,  
  FOREIGN KEY(.....) REFERENCES .....  
  (.....)  
);
```

Partie B : Site web

La plateforme de vente en ligne possède un site web pour ses clients qui passent des commandes en remplissant un formulaire.

1. Expliquer pourquoi, lorsqu'un formulaire contient une quantité importante de données, il est préférable d'utiliser la méthode POST plutôt que GET.
2. Pour la validation du paiement, est-il préférable d'utiliser le protocole HTTP ou HTTPS ? Pourquoi ?
3. Expliquer l'intérêt de vérifier, avant la validation du formulaire, le format des informations saisies (par exemple qu'il n'y a pas de chiffre dans le nom ou qu'il y a bien un @ dans l'adresse de courriel).

Partie C : Requêtes SQL

1. Écrire une requête SQL permettant de récupérer l'identifiant et le libellé de tous les articles coûtant moins de 15 euros.
2. Expliquer ce que fait la requête SQL suivante.

```
SELECT u.IdClient, u.Email, v.IdCmd, v.AdresseLivraison  
FROM Clients as u JOIN Commandes as v  
ON u.IdClient = v.IdClient  
WHERE v.PaiementValide = False;
```

3. Écrire une requête SQL permettant de récupérer le libellé des articles de la commande 1345.
4. On suppose que l'attribut IdArticle de la table Articles est auto-incrémenté et ne doit donc pas être précisé lors de l'ajout d'un nouvel article. Écrire une requête SQL permettant d'ajouter dans la base ce nouvel article.



« Cet imperméable se replie en forme de pochette. »

Prix : 9,99 euros