

Exercice 10

BDD

Dans cet exercice on utilise les mots du langage SQL suivant:
SELECT, FROM, WHERE, JOIN, INSERT INTO, VALUES, COUNT, ORDER BY.

Dans une entreprise imaginaire de location de films, les données sont regroupées dans 3 tables dont les schémas relationnels sont:

Table Clients (codecli, prenomcli, nomcli, ruecli, cpcli, villecli)
Clé primaire : codecli

Table Films (codefilm, nomfilm)
Clé primaire : codefilm

Table Locations (codecli, codefilm, datedebut, duree)
Clé primaire : codecli, codefilm
Clé étrangère : codefilm de la table Films, codecli de la table Clients

Voici un extrait de la table Clients:

Table : Clients

codecli	prenomcli	nomcli	ruecli	cpcli	villecli
1	Alberto	Dubois	3 Rue du Pont	57500	Saint-Avold
2	Mi	Volond	4 rue de la liberté	57500	Saint-Avold
3	Roger	Botas	5 place du marché	57500	Saint-Avold
4	Edouard	Noulas	41 rue de l'église	57600	Forbach
5	Paul	Lontague	21 Boulevard des oiseaux	57800	Freyming
6	Eric	Pondier	14, rue des Agates	57600	Forbach
7	Thomas	Malon	12, rue des lapins	57600	Forbach
8	Rénato	Point	451, rue de légalité	57500	Saint-Avold
9	Michel	Botas	17, rue des hochets	57500	Saint-Avold
10	David	Collague	14, rue Utrillo	57600	Forbach
11	Simon	Potillon	17, rue des marguerittes	57800	Freyming

? **QUESTION 1:**

1. Pourquoi as-t-on été obligé d'utiliser l'attribut codecli comme clé primaire de la table Clients?
2. L'entreprise enregistre son 100ème client, M. Térieur Alain domicilié 12 rue de l'extérieur 57600 Forbach.
Écrire une requête SQL permettant d'enregistrer ce nouveau client dans la table Clients
3. On souhaite modifier le prénom du second client de la table par Michel (Mi était une erreur d'écriture).
Écrire une requête SQL permettant cette modification

Voici des extraits des deux autres tables

Table : Films

codefilm	nomfilm
1	C'est arrivé près de chez vous
2	Bernie
3	Dans la peau de John Malkovitch
4	Intouchables
5	Ong Bak
6	Shoot' Em UP
7	Tigres et dragons
8	Matrix 1
9	Machete
10	Boulevard de la mort
11	Brain dead

Table : Locations

codecli	codefilm	datedebut	duree
1	2	2013-04-11	1
1	4	2013-04-12	3
1	5	2013-04-13	3
2	1	2013-04-09	2
3	2	2013-04-15	5
4	1	2013-04-17	1
4	6	2013-04-21	2
5	2	2013-04-25	3
6	8	2013-05-01	2
7	7	2013-04-09	1
7	9	2012-12-31	4

? QUESTION 2:

1. Que produit cette requête SQL?
`SELECT nomfilm FROM Films;`
2. Que produit cette requête SQL?
`SELECT * FROM clients WHERE villecli="Saint-Avold";`
3. On rappelle qu'en SQL, la fonction d'agrégation COUNT() permet de compter le nombre d'enregistrements dans une table.
Écrire une requête qui permet d'obtenir le nombre de films proposé à la location.

? QUESTION 3:

Voici une requête SQL:

```
SELECT Clients.nomcli, Films.nomfilm, Locations.datedebut
FROM Films
JOIN Locations
  ON Films.codefilm=Locations.codefilm
JOIN Clients
  ON Locations.codecli=Clients.codecli
WHERE Clients.villecli="Forbach"
ORDER BY Locations.datedebut;
```

En vous basant uniquement sur les extraits des tables proposées, donner le tableau résultant de cette requête.

? QUESTION 4:

Le service Marketing de l'entreprise souhaite avoir la liste des clients (avec leurs adresses) ayant loué le film "Bernie" afin de leur proposer à la location la suite du film: "Bernie - le retour".

Écrire une requête permettant d'obtenir cette liste.