

Devoir surveillé n°3 – NSI
25/11/2024

Les exercices suivants portent sur les bases de données relationnelles et les requêtes SQL.
En particulier, les mots-clés suivants peuvent être utilisés :
SELECT, CREATE TABLE, FROM, WHERE, JOIN ON, INSERT INTO, VALUES, UPDATE, SET, COUNT, DELETE, DISTINCT, AND, OR, AS.

Rappels Voici quelques éléments qui pourront être utiles dans l’un ou l’autre des exercices :

- **SELECT COUNT(*)** renvoie le nombre de lignes d’une requête ;
- le mot-clé **DISTINCT** permet de ne pas prendre en compte les doublons ;
- La clause **ORDER BY** suivie d’un attribut permet de trier les résultats par ordre croissant de l’attribut précisé ;

Exercice 1 (6 points)

Une entreprise de location de voitures propose à ses clients la possibilité de rapporter le véhicule dans une autre agence. Les informations correspondantes sont rangées dans une base de données. Voici les extraits de deux tables utilisées :

Agences					
Agence	Ville	CP	Departement	Adresse	Telephone
Licorne	Paris	75001	Paris	123 Rue de la Révolution	0123456789
Le Carrosse	Paris	75001	Paris	456 Virtual Street	0156789012
Vroum	Lyon	69001	Rhône	789 Virtual Lane	0456789034
Rapide	Toulouse	31000	Haute Garonne	321 Virtual Avenue	0567890123
Deep Place	Bordeaux	33000	Gironde	987 Virtual Road	0567890145

Voitures						
id_voiture	marque	modele	kilometrage	nombre_places	type	carburant
1	Renault	Clio	64022	5	Berline	Essence
2	Renault	Clio	50350	5	Berline	Essence
3	Dacia	Sandero	62031	5	Berline	Essence
4	Dacia	Sandero	58955	5	Berline	Essence
5	Dacia	Sandero	65779	5	Berline	Essence
6	Dacia	Sandero	56253	5	Berline	Essence
7	Renault	Clio	49660	5	Berline	Essence
8	Fiat	500	2545	4	Citadine	Electrique
9	Fiat	500	1953	4	Citadine	Electrique
10	Fiat	500	549	4	Citadine	Electrique

1. Donner pour la table **Agences** un type possible pour l’attribut **CP** qui indique le code postal.
2. Donner le résultat de la requête suivante pour les extraits des tables données :

```
SELECT COUNT(DISTINCT 'Telephone')
FROM 'Agences';
```

3. Expliquer à quelle condition l’attribut **Telephone** pourrait servir de clé primaire pour la table **Agences**.

Certaines agences ont décidé de partager un même standard téléphonique. On ajoute à la table **Agences** l'attribut **id_agence** qui est utilisé comme clé primaire. Voici la nouvelle table obtenue :

Agences						
id_agence	Agence	Ville	CP	Departement	Adresse	Telephone
1	Licorne	Paris	75001	Paris	123 Rue de la Révolution	0123456789
2	Le Carrosse	Paris	75001	Paris	456 Virtual Street	0156789012
3	Vroum	Lyon	69001	Rhône	789 Virtual Lane	0456789034
4	Rapide	Toulouse	31000	Haute Garonne	321 Virtual Avenue	0567890123
5	Deep Place	Bordeaux	33000	Gironde	987 Virtual Road	0567890145

Nous souhaitons créer une table **couple_voitures_agences**.

Elle permettra d'associer la table **Agences** à la table **Voitures**.

On précise que la table **Voitures** a pour clé primaire **id_voiture**.

4. Décrire les attributs de cette nouvelle table ; une clé primaire sera soulignée et une clé étrangère commencera par un dièse (#).
5. Écrire une requête qui permet d'enregistrer dans la table **couple_voitures_agences** le fait que le véhicule 2 est associé à l'agence Deep Place.
6. Écrire une requête qui permet d'actualiser la table pour indiquer que le véhicule 2 se trouve maintenant à l'agence Le Carrosse.
7. Écrire une requête qui permet d'afficher le type, la marque et le nom de l'agence pour l'ensemble des véhicules.

Exercice 2 (10 points)

Amélie souhaite organiser sa collection de CD. Elle a commencé par enregistrer toutes les informations sur un fichier CSV mais elle trouve que la recherche d'informations est longue et fastidieuse. Elle repense à son cours sur les bases de données et elle se dit qu'elle doit pouvoir utiliser une base de données relationnelle pour organiser sa collection.

Partie A (4 points)

Dans cette partie on utilise une seule table.

Voici un extrait de la table **Chanson**.

Chanson			
id	titre	album	groupe
1	Sunburn	Showbiz	Muse
2	Muscle Museum	Showbiz	Muse
3	Showbiz	Showbiz	Muse
4	New Born	Origin of Symmetry	Muse
5	Sing for Absolution	Absolution	Muse
6	Hysteria	Absolution	Muse
7	Welcome too the Jungle	Appetite for Destruction	Guns N' Roses
8	Muscle Museum	Hullabaloo	Muse
9	Showbiz	Hullabaloo	Muse

1. L'attribut **titre** peut-il être une clé primaire pour la table **Chanson** ? Justifier.
2. Donner le résultat de la requête suivante :

```
SELECT titre, album
FROM Chanson
WHERE groupe = 'Guns N' Roses';
```

3. Écrire une requête SQL permettant d'obtenir tous les titres des chansons de l'album Showbiz dans l'ordre croissant.
4. Écrire une requête SQL permettant d'ajouter la chanson dont le titre est Megalomania de l'album Hullabaloo du groupe Muse.

Amélie a remarqué une faute de frappe dans la chanson Welcome too the Jungle qui s'écrit normalement Welcome to the Jungle.

5. Écrire une requête SQL permettant de corriger cette erreur.

Partie B (6 points)

Dans cette partie on utilise trois tables.

Voici des extraits des trois tables **Chanson**, **Album** et **Groupe**.

Chanson		
id	titre	id_album
1	Sunburn	1
2	Muscle Museum	1
3	Showbiz	1
4	New Born	2
5	Sing for Absolution	4
6	Hysteria	4
7	Welcome to the Jungle	5
8	Muscle Museum	3
9	Showbiz	3

Groupe	
id	nom
1	Muse
2	Guns N' Roses

Album			
id	titre	année	id_groupe
1	Showbiz	1999	1
2	Origin of Symmetry	2001	1
3	Hullabaloo	2002	1
4	Absolution	2003	1
5	Appetite for Destruction	1987	2

- Expliquer l'intérêt d'utiliser trois tables **Chanson**, **Album** et **Groupe** au lieu de regrouper toutes les informations dans une seule table.
- Expliquer le rôle de l'attribut **id_album** de la table **Chanson**.
- Proposer alors un schéma relationnel pour cette version de la base de données. On pensera à bien spécifier les clés primaires en les soulignant et les clés étrangères en les faisant précéder par le symbole #.
- Écrire une requête SQL permettant d'obtenir tous les noms des albums contenant la chanson Showbiz.
- Écrire une requête SQL permettant d'obtenir tous les titres avec le nom de l'album des chansons du groupe Muse.
- Décrire par une phrase ce qu'effectue la requête SQL suivante :

```
SELECT COUNT(*) AS tot
FROM Album AS a
JOIN Groupe AS g ON a.id_groupe = g.id
WHERE g.nom = 'Muse';
```