

# Exercice 1 - Contenu de la Base de Données Relationnelle

## 1.1) Peut-on avoir simultanément dans la relation CHAMBRE les 3 tuples suivants ?

- **Tuples donnés :**
  - $T_iT_i$  : (13, 61, 10, 20, 'Confort')
  - $T_jT_j$  : (13, 12, 5, 17, 'Confort')
  - $T_kT_k$  : (13, 61, 12, 17, 'Luxe')

**Réponse :** Non, on ne peut pas avoir simultanément les tuples  $T_iT_i$  et  $T_kT_k$  dans la relation CHAMBRE. Ces deux tuples partagent le même numéro de chambre ( $numCh = 61$ ) et le même numéro d'hôtel ( $numHot = 13$ ) mais ont des valeurs différentes pour  $nbrDouble$ ,  $nbrSimple$  et  $catCh$ . Cela viole le principe d'unicité de la clé primaire, qui doit être unique pour chaque enregistrement dans la table.

---

## 1.2) À quelles conditions peut-on supprimer les tuples suivants ?

- **Tuple (27, 12, 13, 17, 'Luxe') dans la relation CHAMBRE :**
    - Pour supprimer ce tuple, il ne doit pas y avoir de réservations actives associées à cette chambre dans la relation RÉSERVATION. En effet, supprimer une chambre alors qu'elle est encore réservée violerait l'intégrité référentielle. Il faudrait d'abord supprimer ou mettre à jour ces réservations.
  - **Tuple (75, 61, 74, '16-Novembre-2023', 5, 7) dans la relation RÉSERVATION :**
    - Ce tuple peut être supprimé sous réserve qu'il n'y ait pas de dépendances référentielles dans la relation CLIENT ou d'autres dépendances qui requièrent cette réservation. Sinon, il pourrait être nécessaire de supprimer les dépendances associées ou de mettre à jour les clés étrangères.
- 

## 1.3) À quelles conditions peut-on insérer les tuples suivants ?

- **Tuple (12, 03, 61, '13-Novembre-2023', 10, 3) dans la relation RÉSERVATION :**
    - Pour insérer ce tuple dans RÉSERVATION, il faut que les valeurs de  $numHot = 12$  et  $numCh = 03$  existent dans la table CHAMBRE, et que  $numCli = 61$  existe dans la table CLIENT. Cela assure que les clés étrangères sont valides. En outre, la date de début de réservation ( $dateDeb$ ) doit être libre pour cette chambre.
  - **Tuple (47, 'Beau séjour', 'Nice', 'PACA', 5) dans la relation HÔTEL :**
    - Ce tuple peut être inséré à condition que  $numHot = 47$  soit unique dans la table HÔTEL. Cela respecte la contrainte d'unicité de la clé primaire pour cette table.
-

## Exercice 2 - Requêtes d'Interrogation SQL en SQL-89 et SQL-92

### a) Liste des hôtels (numHot, nomHot) de la ville de 'Paris'

Correction :

```
sql
Copier le code
SELECT numHot, nomHot
FROM Hôtel
WHERE vilHot = 'Paris';
```

- Cette requête retourne le numéro et le nom des hôtels situés à Paris.
- 

### b) Liste (numHot, nomHot, vilHot, numCh, catCh) des chambres des hôtels de catégorie '3 étoiles' de la région 'Alpes'

Correction :

```
sql
Copier le code
SELECT h.numHot, h.nomHot, h.vilHot, c.numCh, c.catCh
FROM Hôtel h
JOIN Chambre c ON h.numHot = c.numHot
WHERE h.catHot = 3 AND h.régionHot = 'Alpes';
```

- **Explication** : Cette requête joint Hôtel et Chambre pour obtenir la liste des chambres des hôtels de catégorie 3 étoiles situés dans la région "Alpes".
- 

### c) Liste (numCli, nomCli, numHot, dateDeb) des clients qui ont une réservation de plus de 5 jours dans un hôtel de Nice, triée par numéro du client

Correction :

```
sql
Copier le code
SELECT r.numCli, cl.nomCli, r.numHot, r.dateDeb
FROM Réservation r
JOIN Client cl ON r.numCli = cl.numCli
JOIN Hôtel h ON r.numHot = h.numHot
WHERE r.nbJours > 5 AND h.vilHot = 'Nice'
ORDER BY r.numCli;
```

- **Explication** : Cette requête filtre les réservations pour obtenir uniquement celles de plus de 5 jours dans les hôtels situés à Nice, triées par numéro de client.
- 

### d) Liste (numHot, nomHot, vilHot, numCh, numCli, nomCli) des chambres des hôtels de la région 'Bretagne' réservées par des clients de la ville de 'Paris'

Correction :

```

sql
Copier le code
SELECT h.numHot, h.nomHot, h.vilHot, c.numCh, cl.numCli, cl.nomCli
FROM Hôtel h
JOIN Chambre c ON h.numHot = c.numHot
JOIN Réservation r ON c.numHot = r.numHot AND c.numCh = r.numCh
JOIN Client cl ON r.numCli = cl.numCli
WHERE h.régionHot = 'Bretagne' AND cl.vilCli = 'Paris';

```

- **Explication** : Cette requête permet d'obtenir les chambres des hôtels situés en Bretagne, réservées par des clients de Paris.
- 

## e) Liste des clients (numCli, nomCli) qui n'ont pas de location le 1er décembre 2023

### Correction :

```

sql
Copier le code
SELECT numCli, nomCli
FROM Client
WHERE numCli NOT IN (
    SELECT numCli
    FROM Réservation
    WHERE dateDeb <= '2023-12-01'
    AND dateDeb + INTERVAL nbJours DAY > '2023-12-01'
);

```

- **Explication** : La sous-requête identifie les clients ayant une réservation active le 1er décembre 2023, et **NOT IN** retourne ceux sans réservation ce jour-là.

## Correction pour la question f

### Requête en SQL-92 :

```

sql
Copier le code
SELECT c.numCh, c.catCh, r.numCli, r.dateDeb, r.nbJours
FROM Chambre c
LEFT JOIN Réservation r ON c.numHot = r.numHot AND c.numCh = r.numCh
WHERE c.numHot = 74;

```

### Explication :

- **LEFT JOIN** : Utilisé pour inclure toutes les chambres de l'hôtel numéro 74, même celles sans réservation (les valeurs des colonnes de **Réservation** seront **NULL** pour les chambres sans réservation).
- **ON c.numHot = r.numHot AND c.numCh = r.numCh** : Joint les tables **Chambre** et **Réservation** sur **numHot** et **numCh** pour lier chaque chambre à ses éventuelles réservations.
- **WHERE c.numHot = 74** : Filtre pour ne sélectionner que les chambres de l'hôtel numéro 74.

## Résultat

Cette requête retourne les colonnes suivantes pour chaque chambre de l'hôtel numéro 74 :

- **numCh** : Numéro de la chambre.
- **catCh** : Catégorie de la chambre.
- **numCli** : Numéro du client ayant réservé la chambre (sera NULL si aucune réservation).
- **dateDeb** : Date de début de la réservation (sera NULL si aucune réservation).
- **nbJours** : Nombre de jours de la réservation (sera NULL si aucune réservation).

## Exercice 4 - SQL

**g) Insérer le client de numéro 77, de nom 'Dupond', qui habite 'Rue de la Promenade' à 'Nice'**

**Correction :**

```
sql
Copier le code
INSERT INTO Client (numCli, nomCli, adrCli, vilCli)
VALUES (77, 'Dupond', 'Rue de la Promenade', 'Nice');
```

**Explication :**

- Utilise `INSERT INTO` pour ajouter un nouveau client avec les valeurs spécifiées.
- 

**h) Supprimer toutes les réservations du client de numéro 48**

**Correction :**

```
sql
Copier le code
DELETE FROM Réservation
WHERE numCli = 48;
```

**Explication :**

- `DELETE FROM` avec `WHERE` cible uniquement les réservations du client 48, garantissant que seules ces lignes sont supprimées.
- 

**i) Modifier la catégorie de l'hôtel de numéro 10 pour qu'elle devienne 4**

**Correction :**

```
sql
Copier le code
UPDATE Hôtel
SET catHot = 4
WHERE numHot = 10;
```

**Explication :**

- UPDATE avec SET catHot = 4 modifie la catégorie de l'hôtel de numéro 10. La condition WHERE cible cet hôtel spécifique.

-