

# TD N°1 : Modèle Relationnel

## Exercice 1

### 1. Question 1 : Schéma de la base de données

- En analysant les extensions des relations fournies, le schéma de la base de données serait :
  - **Voiture** (NumVoiture, TypeVoiture, Couleur, NumPropriétaire)
  - **Personne** (NumPersonne, NomPersonne)
  - **Modèle** (TypeVoiture, Puissance)

### 2. Question 2 : Insertion des tuples

- Tuple1 : (14, "Durand") dans la relation **Personne**
  - **Justification** : Ce tuple peut être inséré car NumPersonne = 14 n'existe pas dans **Personne**.
  - **Requête SQL** :

```
sql
Copier le code
INSERT INTO Personne (NumPersonne, NomPersonne) VALUES (14,
'Durand');
```
- Tuple2 : (24, "Arthur") dans la relation **Personne**
  - **Justification** : Ce tuple peut être inséré car NumPersonne = 24 est unique.
  - **Requête SQL** :

```
sql
Copier le code
INSERT INTO Personne (NumPersonne, NomPersonne) VALUES (24,
'Arthur');
```
- Tuple3 : ("234 HG 77", "Peugeot 307", "Rouge", 3) dans la relation **Voiture-1**
  - **Justification** : Ce tuple peut être inséré, car NumVoiture = "234 HG 77" n'existe pas dans **Voiture-1**.
  - **Requête SQL** :

```
sql
Copier le code
INSERT INTO Voiture (NumVoiture, TypeVoiture, Couleur,
NumPropriétaire) VALUES ('234 HG 77', 'Peugeot 307', 'Rouge',
3);
```
- Tuple4 : ("234 HG 77", "Peugeot 306", "Rouge", 5) dans la relation **Voiture-1**

- **Justification** : Ce tuple ne peut pas être inséré dans **Voiture-1** si NumVoiture est une clé primaire, car la valeur NumVoiture = "234 HG 77" existe déjà dans Tuple3.

### 3. Question 3 : Suppression des tuples

- Tuple1 : (1, "Martin") dans la relation **Personne**
  - **Justification** : Ce tuple peut être supprimé si aucune référence à NumPersonne = 1 n'existe dans la relation **Voiture**.
- Tuple2 : (2, "Faure") dans la relation **Personne**
  - **Requête SQL** :

```
sql
Copier le code
DELETE FROM Personne WHERE NumPersonne = 2;
```
- Tuple5 : ("1267 ZE 94", "Renault 25", "Verte", 15) dans la relation **Voiture-1**
  - **Justification** : Ce tuple peut être supprimé si aucune contrainte d'intégrité référentielle ne dépend de NumVoiture = "1267 ZE 94".
  - **Requête SQL** :

```
sql
Copier le code
DELETE FROM Voiture WHERE NumVoiture = '1267 ZE 94';
```

## Exercice 2

### 1. Question 1 : Création des tables

- **Table Fournisseur** :

```
sql
Copier le code
CREATE TABLE Fournisseur (
    NumF INT PRIMARY KEY,
    NomF VARCHAR(50),
    CatF INT,
    VilF VARCHAR(50)
);
```
- **Table Pièce** :

```
sql
Copier le code
CREATE TABLE Pièce (
    NumP INT PRIMARY KEY,
    NomP VARCHAR(50),
    ClrP VARCHAR(20),
    PdsP FLOAT
);
```
- **Table Livraison** :

```

sql
Copier le code
CREATE TABLE Livraison (
    NumF INT,
    NumP INT,
    Qte INT,
    PRIMARY KEY (NumF, NumP),
    FOREIGN KEY (NumF) REFERENCES Fournisseur(NumF),
    FOREIGN KEY (NumP) REFERENCES Pièce(NumP)
);

```

## 2. Question 2 : Exemples d'extensions

- Fournisseur : (1, 'Dupont', 5, 'Paris')
- Pièce : (101, 'Vis', 'Rouge', 0.5)
- Livraison : (1, 101, 100)

## TD N°2 : Algèbre Relationnelle et SQL

### Exercice 1

Requêtes algébriques :

- **R1**: Voiture-1  $\cup$  Voiture-2
- **R2**: Voiture-1 - Voiture-2
- **R3**: Voiture-2 - Voiture-1
- **R4**: Voiture-1  $\cap$  Voiture-2
- **R5**:  $\sigma[\text{Couleur} = \text{'Rouge'}](\text{Voiture-1})$
- **R6**:  $\sigma[\text{Puissance} \geq 7](\text{Modèle})$
- **R7**:  $\pi[\text{TypeVoiture}](\text{Voiture-1})$
- **R8**:  $\pi[\text{NomPersonne}](\text{Personne})$
- **R9**:  $\pi[\text{TypeVoiture}, \text{Couleur}](\text{Voiture-1})$
- **R10**:  $\sigma[(\text{TypeVoiture} = \text{'Renault 25'}) \vee (\text{Couleur} = \text{'Rouge'})](\text{Voiture-1})$

## TD N°3 : Expression des Jointures en Algèbre Relationnelle et SQL

### Exercice 1

Pour les jointures naturelles et spécifiques, les expressions algébriques sont :

1. **R1: Propriétaire  $\bowtie$  Modèle**
  - **Explication** : Jointure naturelle sur l'attribut commun TypeVoiture.
  - **Schéma résultat** : (NumVoiture, TypeVoiture, Couleur, NumPropriétaire, Puissance).
2. **R2: Propriétaire  $\bowtie$  (Propriétaire.TypeVoiture = Modèle.TypeVoiture) Modèle**

- **Explication** : Jointure avec condition spécifique sur TypeVoiture.
  - **Schéma résultat** : Identique à R1, ajoute les informations de puissance de la voiture.
3. **R3: Propriétaire** □ **Personne**
- **Explication** : Jointure naturelle sur l'attribut NumPropriétaire = NumPersonne.
  - **Schéma résultat** : (NumVoiture, TypeVoiture, Couleur, NumPropriétaire, NomPersonne).
4. **R4: Propriétaire** □ **(NumPropriétaire = NumPersonne) Personne**
- **Explication** : Jointure spécifique entre Propriétaire et Personne sur NumPropriétaire = NumPersonne.
  - **Schéma résultat** : Similaire à R3 avec ajout de NomPersonne.
5. **R5: Modèle** □ **Propriétaire** □ **(NumPropriétaire = NumPersonne) Personne**
- **Explication** : Jointure combinée entre Modèle, Propriétaire, et Personne.
  - **Schéma résultat** : (TypeVoiture, Puissance, NumVoiture, Couleur, NumPropriétaire, NomPersonne).

## Exercice 2 : Traduction en SQL

1. **R1** :

```
sql
Copier le code
SELECT * FROM Propriétaire NATURAL JOIN Modèle;
```

2. **R2** :

```
sql
Copier le code
SELECT * FROM Propriétaire INNER JOIN Modèle ON Propriétaire.TypeVoiture =
Modèle.TypeVoiture;
```

3. **R3** :

```
sql
Copier le code
SELECT * FROM Propriétaire NATURAL JOIN Personne;
```

4. **R4** :

```
sql
Copier le code
SELECT * FROM Propriétaire INNER JOIN Personne ON
Propriétaire.NumPropriétaire = Personne.NumPersonne;
```

5. **R5** :

```
sql
Copier le code
SELECT *
FROM Modèle
NATURAL JOIN Propriétaire
```

```
INNER JOIN Personne ON Propriétaire.NumPropriétaire =  
Personne.NumPersonne;
```

### Exercice 3 : Exécution des Requêtes SQL

- **R6 :**

```
sql  
Copier le code  
SELECT NumVoiture, TypeVoiture, NumPropriétaire, NomPersonne  
FROM Propriétaire P JOIN Personne Q ON (P.NumPropriétaire = Q.NumPersonne)  
WHERE Couleur = 'bleue';
```

- **R7 :**

```
sql  
Copier le code  
SELECT P.TypeVoiture, Puissance, NomPersonne  
FROM Propriétaire P, Personne Q, Modèle M  
WHERE P.TypeVoiture = M.TypeVoiture AND P.NumPropriétaire = Q.NumPersonne;
```

- **R8 :**

```
sql  
Copier le code  
SELECT * FROM Propriétaire P RIGHT OUTER JOIN Modèle M ON (P.TypeVoiture =  
M.TypeVoiture);
```

## TD N°4 : Jointures en SQL

### Requêtes SQL

1. **Liste des noms et numéros des fournisseurs des pièces livrées :**

```
sql  
Copier le code  
SELECT DISTINCT NomF, NumF  
FROM Fournisseur F  
JOIN Livraison L ON F.NumF = L.NumF;
```

2. **Fournisseurs ayant livré la pièce numéro 4 en quantité inférieure à 50 :**

```
sql  
Copier le code  
SELECT NumF, NomF  
FROM Fournisseur F  
JOIN Livraison L ON F.NumF = L.NumF  
WHERE L.NumP = 4 AND L.Qte < 50;
```

3. **Numéros et noms des pièces avec les numéros et noms de leurs fournisseurs :**

```
sql  
Copier le code  
SELECT P.NumP, P.NomP, F.NumF, F.NomF  
FROM Pièce P  
JOIN Livraison L ON P.NumP = L.NumP  
JOIN Fournisseur F ON L.NumF = F.NumF;
```

4. **Pièces de couleur rouge livrées par des fournisseurs de Paris :**

```
sql
```

```
Copier le code
SELECT P.NumP, P.NomP
FROM Pièce P
JOIN Livraison L ON P.NumP = L.NumP
JOIN Fournisseur F ON L.NumF = F.NumF
WHERE P.ClrP = 'Rouge' AND F.VilF = 'Paris';
```

**5. Toutes les pièces avec éventuellement leur quantité livrée et le numéro du fournisseur :**

```
sql
Copier le code
SELECT P.NumP, P.NomP, L.Qte, L.NumF
FROM Pièce P
LEFT JOIN Livraison L ON P.NumP = L.NumP;
```