

## Bases de données

### Exercice 1

1. (a) Les attributs de la relation **Salle** sont :

Numéro, Capacité, Ordinateur, Accès

- (b) Le domaine des attributs **Numéro** et **Capacité** est INT. Le domaine des attributs **Ordinateur** et **Accès** est TEXT.
- (c) L'attribut **Numéro** peut être la clé primaire de la relation **Salle** car deux salles différentes ont nécessairement deux numéros distincts. En revanche, la capacité d'une salle ne permet pas de déterminer de manière unique la salle (on le constate avec les salles 302 et 323) et, de même pour l'accès à une salle.
2. Un enregistrement est une ligne de données. Ici, la relation compte 4 enregistrements.
3. Il suffit de récapituler les informations précédentes pour obtenir le schéma de relation :

**Salle** = ((**Numéro**:INT), (**Capacité**:INT), (**Ordinateur**:TEXT), (**Accès**:INT))

### Exercice 2

1. (a) Les livres de Jules Renard sont ceux dont l'**Id\_auteur** est 713. On constate en lisant la table **Livre** que leur état est bon.
- (b) La colonne qui sert à distinguer les différents livres de la table **Livre** est la colonne **Réf**. C'est donc la clé primaire de cette table.
2. La clé étrangère de la table **Livre** qui permet de la relier à la table **Écrivain** est la colonne **Id\_auteur** (qui permet d'identifier les écrivains de manière certaine).
3. On récapitule les attributs, leurs domaines respectifs, en soulignant les clés primaires et en signalant les clés étrangères à l'aide d'un dièse :

**Livre** = ((**Réf**:TEXT), (#**Id\_auteur**:INT), (**titre**:TEXT), (**Dispo**:TEXT), (**État**:TEXT))  
**Écrivain** = ((**Id\_auteur**:INT), (**Nom**:TEXT), (**Prénom**:TEXT), (**Naissance**:INT))

### Exercice 3

1. (a) On lit sur la table **Client** que l'**Id\_Client** de Louis Leclerc est le numéro 3. En se reportant à la table **Voiture**, on constate que sa voiture part à l'atelier **Carrosserie**, et la table **Tarif** indique que le chef de l'atelier est Gérard. C'est donc lui qui s'occupera de la voiture de Louis Leclerc.
- (b) Paul est le chef de l'atelier **Pneu** (se reporter à la table **Tarif**). En lisant la table **Voiture**, on constate que 3 voitures partent de son atelier. Paul s'occupera donc de 3 voitures dont 2 Renault.
2. (a) Les voitures de la table **Voiture** sont distinguées par leurs plaques minéralogiques, donc la colonne **Plaque** est la clé primaire de cette table.  
 Pour la table **Client**, nous avons le choix entre deux colonnes : **Id\_Client** et **Téléphone**. Nous choisirons la plus simple, **Id\_Client**.  
 Pour la table **Tarif**, la clé primaire est **Atelier**.
- (b) La colonne **Propriétaires** de la table **Voiture** fait référence à la clé primaire **Id\_client** de la table **Client**. C'est donc une clé étrangère.  
 La colonne **Atelier** de la table **Voiture** fait référence à la clé primaire **Atelier** de la table **Tarif**. C'est donc également une clé étrangère.
- (c) On récapitule toutes ces informations dans le schéma relationnel de ces trois tables :

**Voiture** = ((**Plaque**:TEXT), (#**Propriétaire**:INT), (**Marque**:TEXT),  
 (**Modèle**:TEXT), (#**Atelier**:TEXT))  
**Client** = ((**Id\_client**:INT), (**Nom**:TEXT), (**Prénom**:TEXT), (**Téléphone**:INT))  
**Tarif** = ((**Atelier**:TEXT), (**Chef**:TEXT), (**Prix**:INT))

3. On commence par les tables `Client` et `Tarif` et on finit par la table `Voiture`, car cette dernière table contient deux clés étrangères référençant les clés primaires des deux premières.

```
CREATE TABLE "Tarif" (
    "Atelier" TEXT,
    "Chef" TEXT,
    "Prix" INT,
    PRIMARY KEY ("Atelier")
);

INSERT INTO "Tarif" VALUES("Pneu","Paul",75);
INSERT INTO "Tarif" VALUES("Mécanique","Roger",120);
INSERT INTO "Tarif" VALUES("Carrosserie","Gérard",145);

CREATE TABLE "Client" (
    "Id_client" INT,
    "Nom" TEXT,
    "Prénom" TEXT,
    "Téléphone" INT,
    PRIMARY KEY ("Id_client")
);

INSERT INTO "Client" VALUES(1,"Perha","Marie",0668543252);
INSERT INTO "Client" VALUES(2,"Ridy","Anne",0675463309);
INSERT INTO "Client" VALUES(3,"Leclerc","Louis",0667240908);
INSERT INTO "Client" VALUES(4,"McGregor","Thomas",0679856423);
INSERT INTO "Client" VALUES(5,"Ricciardo","Constance",0619798669);
INSERT INTO "Client" VALUES(6,"Lauquie","Faustine",0638899821);

CREATE TABLE "Voiture" (
    "Plaque" TEXT,
    "Propriétaire" INT,
    "Marque" TEXT,
    "Modèle" TEXT,
    "Atelier" TEXT,
    PRIMARY KEY ("Plaque"),
    FOREIGN KEY ("Propriétaire") REFERENCES "Client"("Id_client"),
    FOREIGN KEY ("Atelier") REFERENCES "Tarif"("Atelier")
);

INSERT INTO "Voiture" VALUES("DG 103 HY",3,"Peugeot","807","Carrosserie");
INSERT INTO "Voiture" VALUES("EF 334 GA",1,"Renault","Clio","Pneu");
INSERT INTO "Voiture" VALUES("EI 189 KA",4,"Fiat","Uno","Pneu");
INSERT INTO "Voiture" VALUES("FA 934 TH",5,"Lancia","Delta","Mécanique");
INSERT INTO "Voiture" VALUES("BB 512 IJ",6,"Renault","Twingo","Pneu");
INSERT INTO "Voiture" VALUES("FD 246 MT",2,"Fiat","500","Mécanique");
```

4. (a) Cette requête permet de remplacer le nom du chef de l'atelier `Pneu` : `Yves` remplace `Paul`.  
 (b) Cette requête permet de sélectionner les marques et les modèles des voitures envoyées à l'atelier `Pneu`, c'est-à-dire :

Renault	Clio
Fiat	Uno
Renault	Twingo

- (c) Cette requête permet de sélectionner les propriétaires des voitures de marque `Renault` qui sont envoyées à l'atelier `Pneu`, c'est-à-dire :

1
6

(d) Cette requête permet de supprimer de la table *Voiture* la voiture dont la plaque minéralogique est DG 103 HY.

5. (a) La requête est : `SELECT Plaque,Atelier FROM Voiture`  
 Le résultat est :

Voiture

DG 103 HY	Carrosserie
EF 334 GA	Pneu
EI 189 KA	Pneu
FA 934 TH	Mécanique
BB 512 IJ	Pneu
FD 246 MT	Mécanique

(b) La requête est :

```
SELECT Nom FROM Client
INNER JOIN Voiture
ON Voiture.Propriétaire=Client.Id_client
WHERE Marque="Peugeot"
```

Le résultat est :

Leclerc
---------

(c) La requête est : `SELECT Atelier FROM Tarif WHERE Prix<100`  
 Le résultat est :

Pneu
------

(d) La requête est :

```
SELECT Téléphone FROM Client
INNER JOIN Voiture
ON Client.Id_client=Voiture.Propriétaire
WHERE (Marque="Renault") AND (Atelier="Pneu")
```

Le résultat est :

668543252
638899821

(e) La requête est :

```
SELECT Marque,Modèle FROM Voiture
WHERE (Atelier="Carrosserie") OR (Atelier="Mécanique")
```

On aurait également pu faire :

```
SELECT Marque,Modèle FROM Voiture WHERE NOT (Atelier="Pneu")
```

Le résultat est ;

Peugeot	807
Lancia	Delta
Fiat	500

(f) La requête est :

```
UPDATE Client SET Téléphone=0651096754 WHERE Nom="Ridy"
```

(g) Il faut dans un premier temps récupérer l'Id\_client de Thomas McGregor dans la table Client, ce que l'on fait à l'aide de la requête :

```
SELECT Id_client FROM Client WHERE Nom="McGregor"
```

Le résultat est :

4
---

Il faut maintenant supprimer la ligne de la table Voiture correspondant au propriétaire dont le numéro est 4. La requête est :

```
DELETE FROM Voiture WHERE Propriétaire=4
```

On aurait pu rassembler les deux requêtes en une seule :

```
DELETE FROM Voiture
WHERE Propriétaire=(SELECT Id_client FROM Client WHERE Nom="McGregor")
```

#### Exercice 4

1. (a) La requête est : `SELECT nom FROM regions`  
 (b) La requête est : `SELECT nom,dep FROM communes`  
 (c) La requête est : `SELECT nom FROM communes WHERE pop > 100000`  
 (d) La requête est : `SELECT nom FROM communes WHERE (pop > 100000) AND (dep=51)`
2. (a) La requête est : `SELECT COUNT(*) FROM communes`  
 (b) La requête est : `SELECT MIN(pop),MAX(pop) FROM communes`
3. (a) La requête est :  
`SELECT c.nom,d.nom FROM communes AS c INNER JOIN departements AS d ON c.dep=d.id`  
 On a utilisé ici la commande AS pour définir un raccourci à communes et departements et éviter de les réécrire plusieurs fois.  
 (b) La requête est :  
`SELECT c.nom,d.nom,r.nom FROM communes AS c INNER JOIN departements AS d ON c.dep=d.id INNER JOIN regions AS r ON r.id=d.reg`  
 (c) La requête est :  
`SELECT AVG(c.pop) FROM communes AS c INNER JOIN departements AS d ON c.dep=d.id INNER JOIN regions AS r ON r.id=d.reg WHERE r.nom="Champagne-Ardenne"`
4. (a) La requête est :  
`SELECT nom FROM communes WHERE pop > (SELECT MAX(pop) FROM communes AS c JOIN departements AS d ON c.dep = d.id WHERE d.nom = "Ardennes")`  
 (b) La requête est :  
`SELECT nom FROM communes WHERE pop > (SELECT AVG(pop) FROM communes)`