

LICENCE 3 INFORMATIQUE

FINANCE

HISTOIRE

GÉOGRAPHIE

INFORMATIQUE

MATHÉMATIQUES

SCIENCES POUR L'INGÉNIEUR

FRANÇAIS LANGUE ÉTRANGÈRE

ADMINISTRATION ÉCONOMIQUE ET SOCIALE

DIPLÔME D'ACCÈS AUX ÉTUDES UNIVERSITAIRES

LICENCE MENTION INFORMATIQUE



Centre de Télé-enseignement
Universitaire

<http://ctu.univ-fcomte.fr>

FILIÈRE INFORMATIQUE

● **VVISEBD**

Bases de données - Travaux dirigés

Mme DAMY - SYLVIE
sylvie.damy@univ-fcomte.fr



**UNIVERSITÉ DE
FRANCHE-COMTÉ**



Cette brochure a été réalisée en L^AT_EX 2 ϵ .

Avertissements

Le cours de bases de données est composé de quatre parties subdivisées en neuf chapitres, correspondant à l'introduction de différentes notions très générales comme dans la partie 1 (chapitre 1), ou beaucoup plus précises telles que dans la partie 3 (chapitre 7) qui présente le langage SQL (DML).

Seules les parties 2, 3, et 4 donneront lieu à la présentation d'exercices. Les énoncés et les corrigés de ces TDs sont disponibles dans ce document, de la façon suivante :

- Partie 2 du cours : Modélisation relationnelle → chapitre 1 du document de TD
- Partie 3 du cours : SQL → chapitre 2 du document de TD
- Partie 4 du cours : Normalisation → chapitre 3 du document de TD.

La **réalisation des TDs** est importante car c'est avec ces exercices que vous pourrez vraiment tester votre compréhension du cours. La lecture seule du cours et des exercices du cours peut laisser l'impression que tout est compris alors que l'on ne s'est pas confronté aux difficultés.

Pour certains exercices, le **modèle de base de données proposé** peut vous sembler loin de la réalité. Attention, il ne s'agit pas d'essayer de transformer le modèle proposé pour qu'il réponde à votre vision des choses. Vous devez répondre à la question posée en tenant compte du modèle proposé.

Pour certains exercices, une **durée de réalisation de l'exercice** est donnée. Elle est juste informative. Vous pouvez si vous souhaitez partager ce type d'informations avec vos collègues et me permettre d'affiner ces informations, donner sur le forum votre temps de travail pour chaque exercice.

Contents

Avertissements	c
I Enoncés des exercices	1
1 Exercices de la partie 2 : Modélisation relationnelle	3
1.1 Exercice 1 - Partie 2: Le Centre Médical	3
1.2 Exercice 2 - Partie 2 : La multinationale	6
1.3 Exercice 3 - Partie 2 : Immeubles et occupants	7
1.4 Exercice 4 - Partie 2 : Les Cinémas	10
1.5 Exercice 5 - Partie 2 : La Coopérative Viticole	13
2 Exercices de la partie 3 : SQL	15
2.1 Exercice 1 - Partie 3 : Les cinémas	15
2.2 Exercice 2 - Partie 3 :Les cinémas	17
2.3 Exercice 3 - Partie 3 :Le centre médical	18
2.4 Exercice 4- Partie 3 : La Coopérative Viticole	21
2.5 Exercice 5 - Partie 3 :Bars et buveurs de bières	23
3 Exercices de la partie 4 : Normalisation	27
3.1 Exercice 1 - Partie 4 : Scolarité	27
3.2 Exercice 2 - Partie 4 : Père Mère et Enfants	27
3.3 Exercice 3 - Partie 4 : Professeurs - Cours et Matières	28
3.4 Exercice 4 - Partie 4 : Enfants et contrats	29
3.5 Exercice 5 - Partie 4 : Maîtresse de maison	30
3.6 Exercice 6 - Partie 4 : Entreprise	31
3.7 Exercice 7 - Partie 4 : Fournisseurs, Pièces et projets	32
3.8 Exercice 8 - Partie 4 : Les magasins, leurs produits et leurs fournisseurs	33
II Corrigés des exercices	35
4 Exercices de la Partie 2 : Modélisation relationnelle	37
4.1 Exercice 1 - Partie 2 : Le Centre Médical	37
4.2 Exercice 2 - Partie 2 : La multinationale	44
4.3 Exercice 3 - Partie 2 : Immeubles et occupants	48
4.4 Exercice 4 - Partie 2 : Les Cinémas	50
4.5 Exercice 5 - Partie 2 : La Coopérative Viticole	59

5 Exercices de la Partie 3 : SQL	63
5.1 Exercice 1 et 2 - Partie 3 : Les cinémas	63
5.2 Exercice 3 - Partie 3 : Le Centre Médical	72
5.3 Exercice 4 - Partie 3 : La Coopérative Viticole	79
5.4 Exercice 5 - Partie 3 : Bars et buveurs de bières	84
6 Exercices de la partie 4 : Normalisation	89
6.1 Exercice 1 - Partie 4 : Scolarité	89
6.2 Exercice 2 - Partie 4 : Père Mère et Enfants	90
6.3 Exercice 3 - Partie 4 : Professeurs - Cours et Matières	93
6.4 Exercice 4 - Partie 4 : Enfants et contrats	95
6.5 Exercice 5 - Partie 4 : Maîtresse de maison	98
6.6 Exercice 6 - Partie 4 : Entreprise	100
6.7 Exercice 7 - Partie 4 : Fournisseurs, Pièces et projets	103
6.8 Exercice 8 - Partie 4 : Les magasins, leurs produits et leurs fournisseurs	107

Part I

Enoncés des exercices

Chapter 1

Exercices de la partie 2 : Modélisation relationnelle

1.1 Exercice 1 - Partie 2: Le Centre Médical

Un centre médical est constitué de plusieurs médecins. Lors d'une consultation qui met en présence un seul médecin et un seul patient, plusieurs médicaments peuvent être prescrits. Un médecin a plusieurs patients et un même patient peut avoir plusieurs médecins.

- MEDECIN(*Matricule, Nom)
- CONSULTATION (*Numcons, Date, Matricule, NumSS)
- PRESCRIPTION (*Code, *Numcons, NbPrises)
- MEDICAMENT (*Code, Libelle)
- PATIENT (*NumSS, Nom)

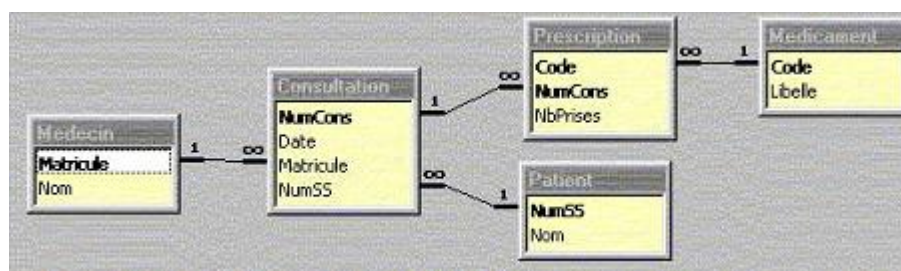


Figure 1.1 : Le modèle relationnel du centre médical

Objectif : Comprendre et interpréter les informations contenues dans un ensemble de relations. Notion de relation, clés primaires et étrangères, lien.

Remarques :

- Les questions sont toutes indépendantes les unes des autres. Pour répondre à une question utilisez systématiquement la base de données proposée avec l'énoncé (ne tenez pas compte des résultats obtenus aux questions précédentes).
- Le modèle relationnel présenté est celui obtenu avec le SGBD ACCESS. Un lien 1..n est noté 1..infini. Et un lien faisant intervenir plusieurs attributs est représenté par autant de traits que d'attributs.
- Le modèle présenté ici est un modèle relationnel, il ne s'agit ni d'un modèle conceptuel de données Merise ni d'un diagramme de classe UML.

Durée de réalisation de l'exercice : Environ 2h de travail.

Contenu des relations

Matricule	Nom
1	Pouthier
2	Receveur
3	Pourcelot

Table 1.1 : Relation MEDECIN

NumCons	Date	Matricule	NumSS
1	04/06/1999	1	1581225128745
2	05/06/1999	1	2781024147456
3	10/11/1999	2	2781024147456
4	10/09/2002	1	2781024147456

Table 1.2 : Relation CONSULTATION

Code	NumCons	NbPrises
1236	1	2
1236	2	1
1236	3	2
1478	1	1

Table 1.3 : Relation PRESCRIPTION

Code	Libelle
1236	Doliprane
1456	Aspegic 500
1478	Primpéran
2589	Vogalène

Table 1.4 : Relation MEDICAMENT

NumSS	Nom
1581225128745	Dupont
2781024147456	Faivre

Table 1.5 : Relation PATIENT

Question 1

En regardant le contenu des relations. Dire quel médecin, Dupont a rencontré lors de sa consultation, et que lui a-t-il prescrit ?

Question 2

En regardant le contenu des relations. Dire quel patient a déjà eu du Primpéran.

Question 3

Modifier les relations pour pouvoir ajouter l'information suivante : Monsieur DUPONT est allé voir en consultation, le 21/06/99, Mr Receveur, et celui-ci lui a prescrit 2 prises d'aspegic 500 ainsi qu'une prise de vogalène.

Question 4

Modifier les relations pour pouvoir ajouter l'information suivante : Melle Dholat a consulté Mr Pouthier le 21/06/00, qui lui a prescrit du magnogène (2 prises par jour) et du doliprane (1 prise par jour). Le numéro de SS de Melle Dholat est : 2701175034068.

Question 5

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Un patient peut-il effectuer plusieurs consultations ?

Question 6

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Un médecin peut-il recevoir plusieurs clients dans une même consultation ?

Question 7

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Plusieurs médicaments peuvent-ils être prescrit lors d'une même consultation ?

Question 8

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Deux médecins différents peuvent-ils prescrire le même médicament ?

Question 9

La sécurité sociale impose à chaque patient de n'avoir qu'un seul médecin généraliste. Modifier le modèle relationnel pour tenir compte de cette nouvelle contrainte.

Question 10

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type CASCADES. Que se passe-t-il si on supprime de la table CONSULTATION le t-uplet correspondant à la consultation N° 2.

Question 11

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type RESTRICTED. Que se passe-t-il si on supprime de la table CONSULTATION le t-uplet correspondant à la consultation N° 2.

Question 12

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type CASCADES. Que se passe-t-il si on modifie dans la table CONSULTATION le t-uplet correspondant à la consultation N° 2, en le passant au numéro 5.

Question 13

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type RESTRICTED. Que se passe-t-il si on modifie dans la table CONSULTATION le t-uplet correspondant à la consultation N° 2, en le passant au numéro 5.

Question 14

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type CASCADES. Que se passe-t-il si on modifie dans la table CONSULTATION le t-uplet correspondant à la consultation N° 2, en le passant au numéro 3.

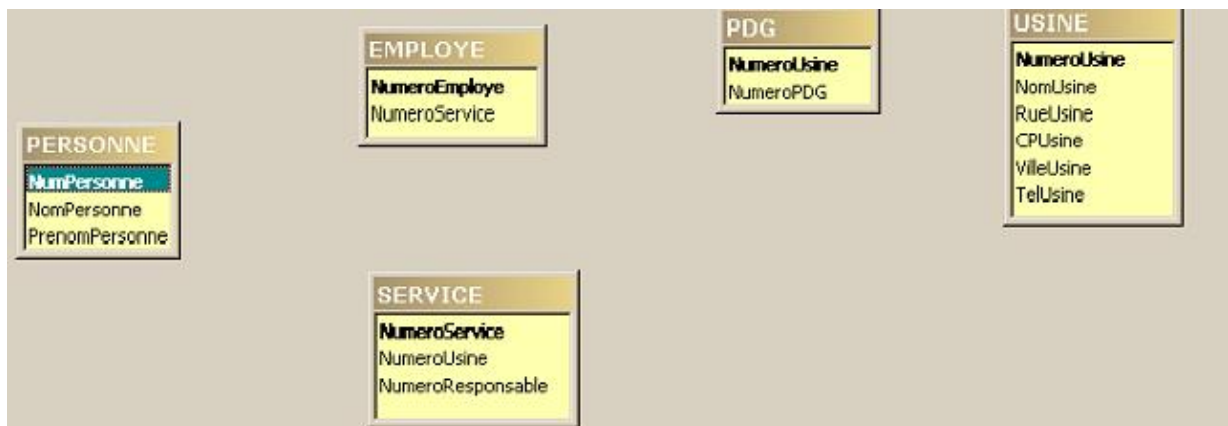
Question 15

Que se passe-t-il si on essaie d'ajouter le t-uplet suivant dans la table PRESCRIPTION (1111, 2, 2) ?

1.2 Exercice 2 - Partie 2 : La multinationale

Pour décrire les informations concernant les services, les employés, les responsables et les PDG des usines d'une multi-nationale, on propose de créer une base de données MULTINATIONALE.

- EMPLOYE (*NumeroEmploye, NumeroService)
- PDG (*NumeroUsine, NumeroPDG)
- PERSONNE (*NumPersonne, NomPersonne, PrenomPersonne)
- SERVICE (*NumeroService, Usine, NumeroResponsable)
- USINE (*NumeroUsine, NomUsine, RueUsine, VilleUsine, CPUusine, TelUsine)



Objectif : Travail complet sur la description des liens

Question 1

L'objectif final de cet exercice est de définir les liens dans le modèle proposé. Pour cette première question trouvez entre quelles relations on peut définir des liens.

Question 2

Définir le ou les liens qui utilisent l'attribut NumService.

Question 3

Définir les liens qui mettent en jeu les attributs NumPersonne, NumeroEmploye, NumeroPDG et NumeroResponsable.

Question 4

Définir les liens qui mettent en jeu l'attribut NumUsine.

Question 5

En résumé nous avons les liens.

1.3 Exercice 3 - Partie 2 : Immeubles et occupants

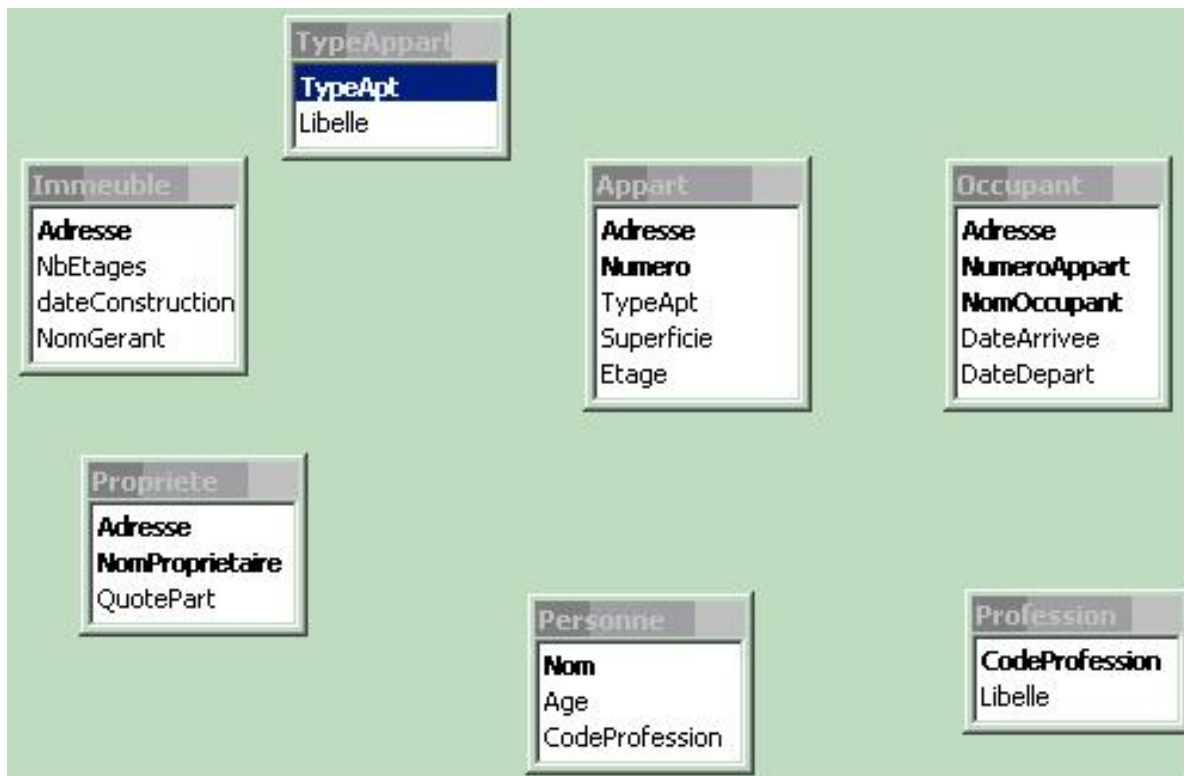
Pour décrire les informations concernant les occupants et les propriétaires d'appartements dans un ensemble d'immeubles, on propose de créer une base de données Immeuble.

- IMMEUBLE (*adresse, nbEtages, dateConstruction, nomGerant)
- PROPRIETE (*adresse, *nomProprietaire, quotePart)
- APPART (*adresse, *numero, typeApt, superficie, etage)
- PERSONNE (*nom, age, codeProfession)
- OCCUPANT (*adresse, *numeroAppart, *nomOccupant, dateArrivee, dateDepart)
- PROFESSION (*codeProfession, libelle)
- TYPEAPPART(*typeapt, libelle)

Objectif : Comprendre l'intérêt des clés candidates et étrangères.

Remarque : Il ne faut surtout pas répondre aux questions en se reposant sur le contenu des relations.

Durée de réalisation de l'exercice : Environ 1h. La première question étant la plus longue 1/2 h.



Contenu des relations

Adresse	NbEtages	dateConstruction	NomGerant
3 rue des granges	4	01/01/1920	DORNIER
Les Griffons	3	10/10/1993	CUENIN

Table 1.6 : Relation IMMEUBLE

Adresse	NomProprietaire	QuotePart
3 rue des granges	DUPONT	100
LES GRIFFONS	DUPONT	40
LES GRIFFONS	MAIRE	60

Table 1.7 : Relation PROPRIETE

Question 1

Définir les liens entre les relations présentées dans le modèle. Pour réaliser cette question il faut commencer par voir comment il est possible de construire de façon systématique les liens entre les relations proposées.

TypeApt	Libelle
00	Studio
01	F1
02	F2
03	F3
04	F4
05	F5
06	plus ...

Table 1.8 : Relation TYPEAPPART

Adresse	Numero	TypeApt	Superficie	Etage
3 RUE DES GRANGES	1	00	40	1
3 RUE DES GRANGES	2	01	60	1
LES GRIFFONS	1	05	150	2

Table 1.9 : Relation APPART

Nom	Age	CodeProfession
BARTHOD	30	04
DUPONT	40	02
MAIRE	50	03
SONZINI	23	01

Table 1.10 : Relation PERSONNE

Adresse	NumeroAppart	NomOccupant	DateArrivee	DateDepart
3 RUE DES GRANGES	1	SONZINI	10/10/1999	
les griffons	1	BARTHOD	14/12/1993	10/10/1998
les griffons	1	MAIRE	11/10/1998	

Table 1.11 : Relation OCCUPANT

CodeProfession	Libelle
01	Enseignant
02	Commerçant
03	Juriste
04	Employé

Table 1.12 : Relation PROFESSION

Question 2

Peut-il y avoir plusieurs immeubles à une même adresse ?

Question 3

Une personne peut-elle habiter plusieurs fois dans le même immeuble ? Dans le même appartement ?

Question 4

Peut-on connaître le propriétaire d'un appartement ?

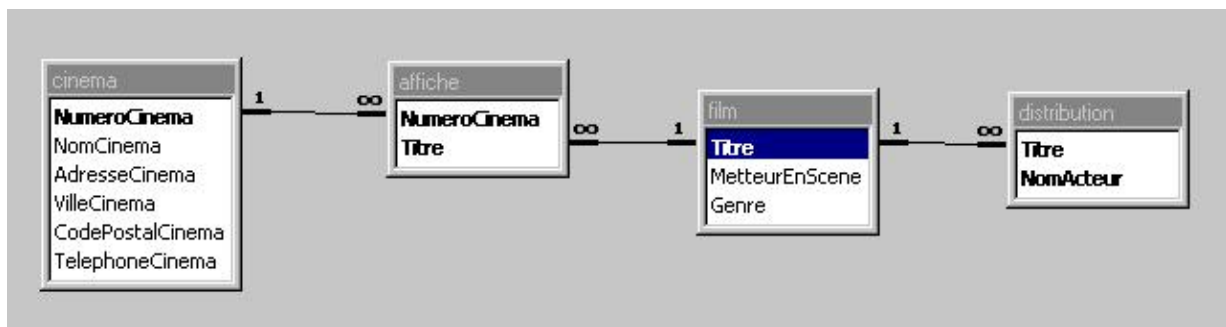
Question 5

Une personne peut-elle être propriétaire dans plusieurs immeubles ?

1.4 Exercice 4 - Partie 2 : Les Cinémas

Gestion des cinémas : l'exemple du cours. Le modèle de cet exemple est le suivant :

- CINEMA (*NumeroCinema, NomCinema, AdresseCinema, VilleCinema, CodePostalCinema, TelephoneCinema)
- AFFICHE (*Titre, *NumeroCinema)
- FILM (*Titre, MetteurEnScene, Genre)
- DISTRIBUTION (*Titre, *NomActeur)

**Contenu des relations**

La relation CINEMA

Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69
2	Plazza	Rue Des Granges	Besancon	25000	03 81 26 35 98
3	Eldorado	Rue St. Lambert	Paris	75015	01 23 56 98 56
4	Pax	Avenue d'Italie	Paris	75013	01 23 56 98 78

NumeroCinema	Titre
1	Alerte
1	Junior
2	Alerte
2	Gazon Maudit
2	L'Effaceur
3	Alerte
3	Casper
4	Alerte
4	Casper
4	Harcelement

La relation AFFICHE

La relation FILM

Titre	MetteurEnScene	Genre
Alerte	Peterson	Aventure
Casper	Silberling	Comedie
Gazon Maudit	Balasko	Comedie
Harcelement	Levinson	Comedie Dramatique
Junior	Reitman	Comedie
L'Effaceur	Russel	Action
Les Voleurs	Techine	Comedie Dramatique
Ma Saison Preferee	Techine	Comedie Dramatique

La relation DISTRIBUTION

Titre	NomActeur
Alerte	Freeman
Alerte	Hoffman
Alerte	Russo
Casper	Pullman
Casper	Ricci
Gazon Maudit	Abril
Gazon Maudit	Balasko
Gazon Maudit	Chabat
Harcelement	Douglas
Harcelement	Moore
Junior	De Mornay
Junior	De Niro
Junior	Schwarzenegger
L'Effaceur	Schwarzenegger

Décrire ce que donne les requêtes suivantes et dire quel est l'opérateur principal des requêtes :

Remarque : La notation relationnelle proposée dans le cours présente deux avantages :

1. Elle peut être tapée en texte simplement. Il n'y a pas de symboles spécifiques que l'on peut rencontrer dans d'autres supports pédagogiques.
2. Le fait de ne pas combiner les opérateurs permet de lire plus simplement les requêtes. Chaque année certains étudiants réduisent l'écriture des requêtes en combinant les opérateurs. En cas de problème il devient très vite impossible de lire et/ou d'interpréter la requête.

Je préfère donc que vous utilisiez un opérateur à la fois (par ligne), comme je le fais dans ce cours, en particulier dans vos rendus (devoirs ou examens).

Question 1

```
Req1 = [VilleCinema, CodePostalCinema] CINEMA
```

Question 2

```
Req2 = S(Genre = "Comedie") FILM
```

Question 3

```
Req3 = FILM [FILM.Titre = AFFICHE.titre] AFFICHE
```

Question 4

```
CINE = [NumeroCinema] CINEMA
```

```
Req4 = AFFICHE [ AFFICHE.NumeroCinema / CINE.NumeroCinema] CINE
```

Question 5

```
REQ1 = S(Titre = "Casper")AFFICHE
```

```
REQ2 = REQ1 [ REQ1.numeroCinema = CINEMA.NumeroCinema] CINEMA
```

```
REQ3 = [NomCinema] REQ2
```

Question 6

```
REQ1 = S(Titre = "Casper") AFFICHE
```

```
REQ2 = REQ1 [ REQ1.NumeroCinema = CINEMA.NumeroCinema] CINEMA
```

```
REQ3 = S(Titre = "Alerte") AFFICHE
```

```
REQ4 = REQ3 [ REQ3.NumeroCinema = CINEMA.NumeroCinema] CINEMA
```

```
REQ5 = REQ2 INTER REQ4
```

```
REQ6 = [NomCinema] REQ5
```

Modifier la requête pour obtenir ce que l'on aurait voulu obtenir.

En utilisant les opérateurs de l'algèbre relationnelle, définir une requête permettant de définir une relation qui donne ce qui est demandé, et préciser quel est l'opérateur principal de votre requête :

Question 7

Le nom des metteurs en scène des films à l'affiche du PAX

Question 8

Le nom des acteurs apparaissant dans la distribution des films à l'affiche à Besançon.

Question 9

Le titre des films d'aventure à l'affiche à Besançon, et le nom du cinéma où ils sont à l'affiche.

Question 10

Le nom et l'adresse des cinémas projetant le film "Junior".

Question 11

Le titre des films ayant le même metteur en scène que le film "les voleurs".

Question 12

Le nom des cinémas de Paris ayant à l'affiche une comédie **et** une comédie dramatique.

Question 13

Le nom des cinémas de Besançon n'ayant pas de film d'action à l'affiche.

Question 14

Le numéro des cinémas ayant à l'affiche tous les genres de films.

Question 15

Proposer et écrire des requêtes dont l'opérateur principal est :

1. Un produit cartésien
2. Une union
3. Une intersection
4. Une différence
5. Une division

1.5 Exercice 5 - Partie 2 : La Coopérative Viticole

Un vin est caractérisé par un numéro entier, un cru, une année de production et un degré.

Un producteur est caractérisé par un numéro entier, un nom, un prénom et une région. Un producteur produit un ou plusieurs vins. Réciproquement, un vin est produit par un ou plusieurs producteurs (éventuellement aucun!).

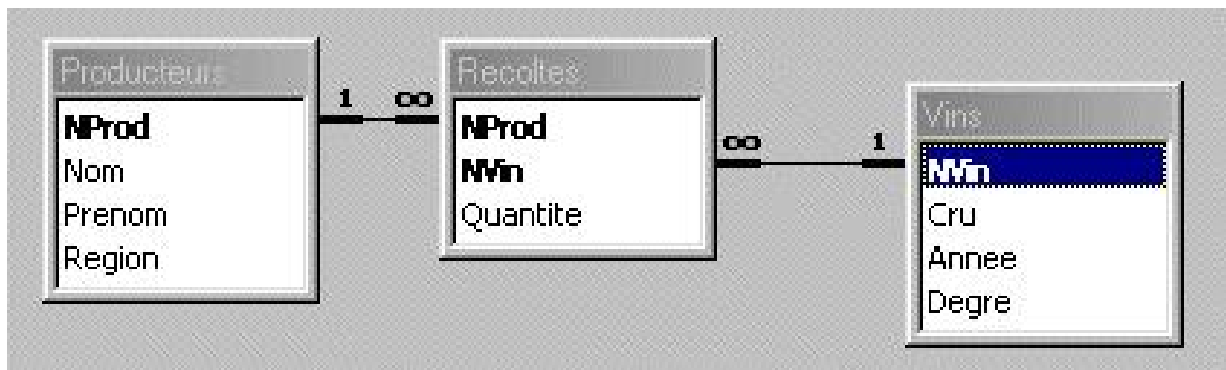
- VINS (*Nvin, Cru, Annee, Degre)
- PRODUCTEURS (*Nprod, Nom, Prenom, Region)
- RECOLTES (*Nprod, *Nvin, Quantite)

Objectif : Exercices sur les opérateurs relationnels

Durée de réalisation de l'exercice : Environ 2h.

Question 1 :

Donner les noms et prénoms des producteurs de la région du Beaujolais.

**Question 2 :**

Quels sont les producteurs de la région du Beaujolais qui ont récolté au moins un vin en quantité supérieur à 300 litres? On veut les noms et les prénoms des producteurs.

Question 3 :

Donnez les numéros des producteurs qui ont déjà produit du vin de degré égal à 12.

Question 4 :

Donner la liste des numéros de vin qui ont un degré supérieur à 12 et qui ont été produits par le producteur numéro 24.

Question 5 :

Quels sont les couples de producteurs qui habitent la même région?

- les t-uplets du résultat seront de la forme NProd1, Nom1, NProd2, Nom2, Région, avec NProd1 différent de NProd2
- la présence d'un t-uplet avec NProd1 suivi de NProd2 interdit la présence d'un t-uplet avec NProd2 suivi de NProd1

Question 6 :

Donnez les noms des producteurs qui ont produit tous les vins.

Question 7 :

Quels sont les producteurs qui produisent au moins tous les vins produits par le producteur 2?

Question 8 :

Donnez les numéros des producteurs qui n'ont jamais produit du vin de degré égal à 12.

Question 9 :

Donnez les numéros des producteurs qui n'ont produit que du vin de degré égal à 12.

Chapter 2

Exercices de la partie 3 : SQL

2.1 Exercice 1 - Partie 3 : Les cinémas

Gestion des cinémas : l'exemple du cours

- CINEMA (*NumeroCinema, NomCinema, AdresseCinema, VilleCinema, CodePostalCinema, TelephoneCinema)
- AFFICHE (*NumeroCinema, *Titre)
- FILM (*Titre, MetteurEnScene, Genre)
- DISTRIBUTION (*Titre, *NomActeur)

Le modèle relationnel est donné par la figure

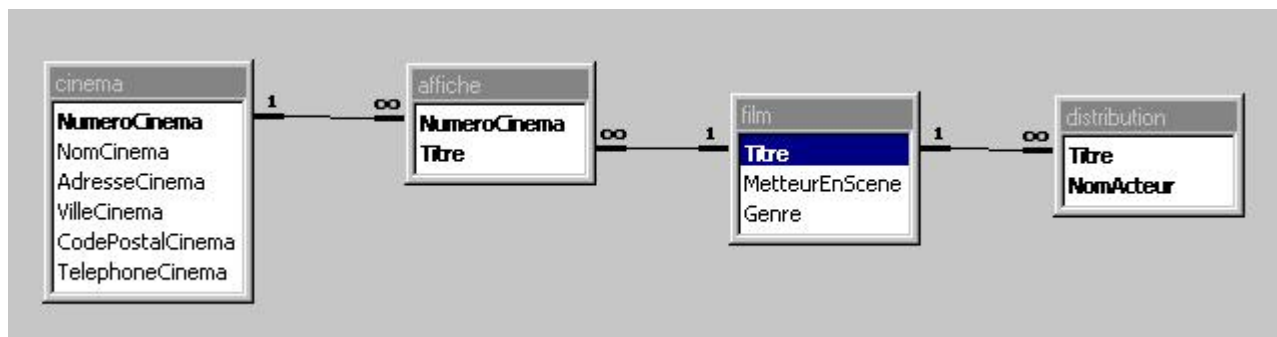


Figure 2.1 : Le modèle relationnel des cinémas

Objectif : Premières écritures de requêtes SQL

Contenu des tables

Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69
2	Plazza	Rue Des Granges	Besancon	25000	03 81 26 35 98
3	Eldorado	Rue St. Lambert	Paris	75015	01 23 56 98 56
4	Pax	Avenue d'Italie	Paris	75013	01 23 56 98 78

Table 2.1 : Table CINEMA

NumeroCinema	Titre
1	Alerte
1	Junior
2	Alerte
2	Gazon Maudit
2	L'Effaceur
3	Alerte
3	Casper
4	Alerte
4	Casper
4	Harcelement

Table 2.2 : Table AFFICHE

Titre	NomActeur
Alerte	Freeman
Alerte	Hoffman
Alerte	Russo
Casper	Pullman
Casper	Ricci
Gazon Maudit	Abril
Gazon Maudit	Balasko
Gazon Maudit	Chabat
Harcelement	Douglas
Harcelement	Moore
Junior	De Mornay
Junior	De Niro
Junior	Schwarzenegger
L'Effaceur	Schwarzenegger

Table 2.3 : Table DISTRIBUTION

Titre	MetteurEnScene	Genre
Alerte	Peterson	Aventure
Casper	Silberling	Comedie
Gazon Maudit	Balasko	Comedie
Harcelement	Levinson	Comedie Dramatique
Junior	Reitman	Comedie
L'Effaceur	Russel	Action
Les Voleurs	Techine	Comedie Dramatique
Ma Saison Preferee	Techine	Comedie Dramatique

Table 2.4 : Table FILM

Ecrire en SQL les requêtes suivantes :

Question 1

Nom des cinémas

Question 2

Le nom et l'adresse des cinémas projetant le film "Junior"

Question 3

Le nom des metteurs en scène des films à l'affiche du PAX

Question 4

Le nom des acteurs apparaissant dans la distribution des films à l'affiche à Besançon.

Question 5

Le titre des films avec Schwarzenegger à l'affiche à Besançon.

Question 6

Le titre des films d'aventure à l'affiche à Besançon, et le nom du cinéma où ils sont à l'affiche.

Question 7

Le titre des films ayant le même metteur en scène que "les voleurs"

2.2 Exercice 2 - Partie 3 :Les cinémas

Toujours l'exercice des cinémas, mais les requêtes sont ici un peu plus complexes.

Question 8

Le nom des cinémas de Paris ayant à l'affiche une comédie et une comédie dramatique

Question 9

Le nom des cinémas de Besançon n'ayant pas de film d'action à l'affiche

Question 10

Le nombre de Cinémas à Besançon

Question 11

Pour chaque cinéma, donnez le nombre de films à l'affiche

Question 12

Le nom des cinémas ainsi que le nombre de films à l'affiche dans ceux-ci

Question 13

Le numéro des cinémas ayant au moins deux films à l'affiche

Question 14

Le nom des cinémas ayant au moins deux films à l'affiche

Question 15

Le titre des films n'ayant pas Schwarzenegger dans leur distribution

Question 16

Le numéro des cinémas qui ont autant de films à l'affiche que le cinéma Numéro 1

Question 17

Pour chaque cinéma, le nombre de films qu'il a à l'affiche. Si le cinéma n'a aucun film à l'affiche, la valeur 0 devra apparaître en face du numéro de cinéma.

Question 18

Numéro des cinémas ayant tous les genres de films à l'affiche. (solution avec EXISTS)

2.3 Exercice 3 - Partie 3 :Le centre médical

Un centre médical est constitué de plusieurs médecins. Lors d'une consultation qui met en présence un seul médecin et un seul patient, plusieurs médicaments peuvent être prescrits. Un médecin a plusieurs patients et un même patient peut avoir plusieurs médecins.

- MEDECIN(*Matricule, Nom)
- CONSULTATION (*Numcons, Date, Matricule, NumSS)
- PRESCRIPTION (*Code, *Numcons, NbPrises)
- MEDICAMENT (*Code, Libelle)
- PATIENT (*NumSecu, Nom)

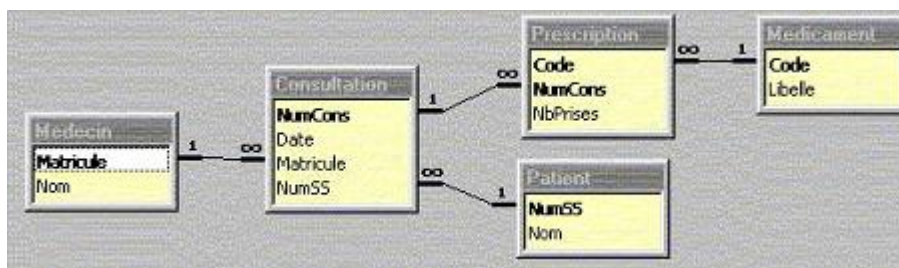


Figure 2.2 : Le modèle relationnel du centre médical

Objectif : Ecriture de requêtes SQL plus ou moins simples avec des variantes

Contenu des tables**Question 1**

Donnez le numéro de sécurité sociale des patients du médecin de matricule 1.

Matricule	Nom
1	Pouthier
2	Receveur
3	Pourcelot

Table 2.5 : Table MEDECIN

NumCons	Date	Matricule	NumSecu
1	04/06/2009	1	1581225128745
2	05/06/2009	1	2781024147456
3	10/11/2009	2	2781024147456
4	10/09/2009	1	2781024147456

Table 2.6 : Table CONSULTATION

Question 2

Donnez le nom des patients du médecin *Pouthier*.

Question 3

Donnez le nom des patients à qui l'on a déjà prescrit du *Doliprane*.

Question 4

Donnez le nom des patients qui ont déjà consulté un médecin qui a déjà eu comme patient, le patient de numéro sécu : *1581225128745*.

Question 5

Donnez le nom des patients qui ont déjà consulté un médecin qui a déjà eu comme patient, le patient *Dupont*.

Question 6

Combien de médicaments ont déjà été prescrits à Monsieur *Faivre* ? Si un médicament a été prescrit plusieurs fois on le compte plusieurs fois.

Question 7

Combien de médicaments ont déjà été prescrits à Monsieur *Faivre* ? On veut connaître le nombre de médicaments différents.

Question 8

Pour chaque patient, donnez le nombre de médicaments qui leur ont déjà été prescrits.

Question 9

Donnez le numéro de sécurité sociale des patients ayant eu au moins deux médicaments prescrits, ainsi que le nombre de médicaments qui leur ont été prescrits.

Code	NumCons	NbPrises
1236	1	2
1236	2	1
1236	3	2
1478	1	1

Table 2.7 : Table PRESCRIPTION

Code	Libelle
1236	Doliprane
1456	Aspegic 500
1478	Primpéran
2589	Vogalène

Table 2.8 : Table MEDICAMENT

Question 10

Donnez le nom des patients ayant eu au moins deux médicaments prescrits.

Question 11

Donnez pour chaque médecin son numéro de matricule et le nombre de consultations qu'il a réalisé.

Question 12

Donnez pour chaque médecin son nom et le nombre de consultations qu'il a réalisé.

Question 13

Donnez pour chaque médecin son nom et le nombre de consultations qu'il a réalisé avec comme patient Mr DUPONT.

Question 14

Donnez pour chaque médecin son nom et le nombre de consultations qu'il a réalisé avec chaque patient. On donnera dans le résultat le nom du médecin, le nom du patient, et le nombre de consultations.

Question 15

Donnez le nom du médecin qui a réalisé le plus de consultations.

Question 16

Donnez le nom du meilleur patient du médecin de matricule "1".

Question 17

Donnez pour chaque médecin son nom et le nom de son meilleur patient (c'est à dire celui ou ceux qui ont réalisés le plus de consultations).

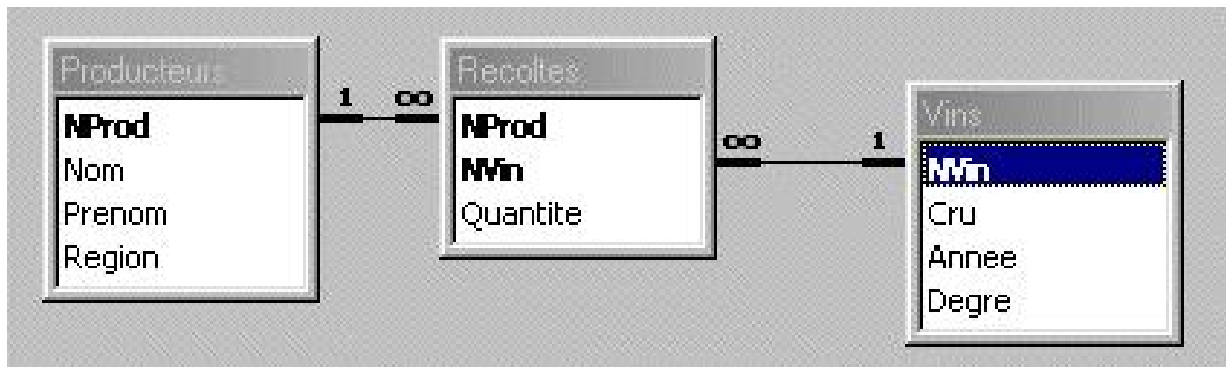
NumSecu	Nom
1581225128745	Dupont
2781024147456	Faivre

Table 2.9 : Table PATIENT

2.4 Exercice 4- Partie 3 : La Coopérative Viticole

Un vin est caractérisé par un numéro entier, un cru, une année de production et un degré. Un producteur est caractérisé par un numéro entier, un nom, un prénom et une région. Un producteur produit un ou plusieurs vins. Réciproquement, un vin est produit par un ou plusieurs producteurs (éventuellement aucun!).

- VIN (*Nvin, Cru, Annee, Degré)
- PRODUCTEUR (*Nprod, Nom, Prenom, Region)
- RECOLTE (*Nprod, *Nvin, Quantite)



Objectif : Un exemple complet de requêtes SQL

Contenu des tables

NVin	Cru	Annee	Degre
1	St Julien	1998	12
2	Médoc	2001	13
12	St Armand	2003	14

Table 2.10 : Table VIN

Nprod	Nom	Prenom	Region
1	Renard	Alain	Beaujolais
2	Renard	Pierre	Beaujolais
3	Dupuis	Laurent	Beaujolais
24	Dupont	Jean	Beaujolais

Table 2.11 : Table PRODUCTEUR

Nprod	Nvin	Quantite
1	1	500
1	2	300
1	12	400
2	1	300
2	12	500
3	1	1000
3	2	50

Table 2.12 : Table RECOLTE

Question 1

Donner les noms et prénoms des producteurs de la région du *Beaujolais* triés par nom et prénom.

Question 2

Quels sont les producteurs de la région du Beaujolais qui ont récolté au moins un vin en quantité supérieur à 300 litres? On veut les noms et les prénoms des producteurs, avec un tri par ordre alphabétique.

Question 3

Donner la liste des numéros de vin qui ont un degré supérieur à 12 ou qui ont été produits par le producteur numéro 24.

Question 4

Quelle est la quantité totale de vin numéro 12 produite?

Question 5

Donner la liste des crus triée par ordre alphabétique. Pour chaque cru, on veut son nom et la quantité de vin produite par ce cru. Variante de cette question : Pour chaque vin on veut son cru et la quantité de vin produite pour ce vin. Et les résultats seront ordonnés par cru.

Question 6

Quels sont les producteurs qui produisent des vins issus de trois crus différents, au moins? Pour chaque producteur sélectionné, on veut le nom et le prénom.

Question 7

Quels sont les couples de producteurs qui habitent la même région?

- les t-uplets du résultat seront de la forme - Nom1, Prenom1, Nom2, Prenom2, Région -, avec NProd1 différent de NProd2
- la présence d'un t-uplet avec NProd1 suivi de NProd2 interdit la présence d'un t-uplet avec NProd2 suivi de NProd1

Question 8

Quels sont les producteurs qui ont produit tous les vins ? Ecrire la requête en utilisant la fonction COUNT.

Question 9

Quels sont les producteurs qui ont produit tous les vins (Nom et Prenom)? Ecrire la requête en utilisant le quantificateur EXISTS et l'opérateur logique NOT.

Question 10

Quels sont les producteurs qui produisent au moins tous les vins produits par le producteur 2 ? Ecrire la requête en utilisant la fonction COUNT.

Question 11

Quels sont les producteurs qui produisent au moins tous les vins produits par le producteur 2? Ecrire la requête en utilisant le quantificateur EXISTS et l'opérateur logique NOT.

2.5 Exercice 5 - Partie 3 :Bars et buveurs de bières

Pour décrire les informations concernant les bars de la région, ainsi que les bières qu'on y trouve, et les buveurs de bière, oncréé une base de données avec les six tables suivantes.

- BAR (*CBar, NomBar, AdresseBar, VilleBar, CPBar)
- SERT (*CBar, *CBiere)
- BIERE (*CBiere, NomBiere, Origine)
- FREQUENTE (*Cbuveur, *CBar)
- BUVEUR (*Cbuveur, NomBuveur, PrenomBuveur)
- AIME (*CBuveur, *CBiere)

Objectif : Quelques requêtes plus complexes

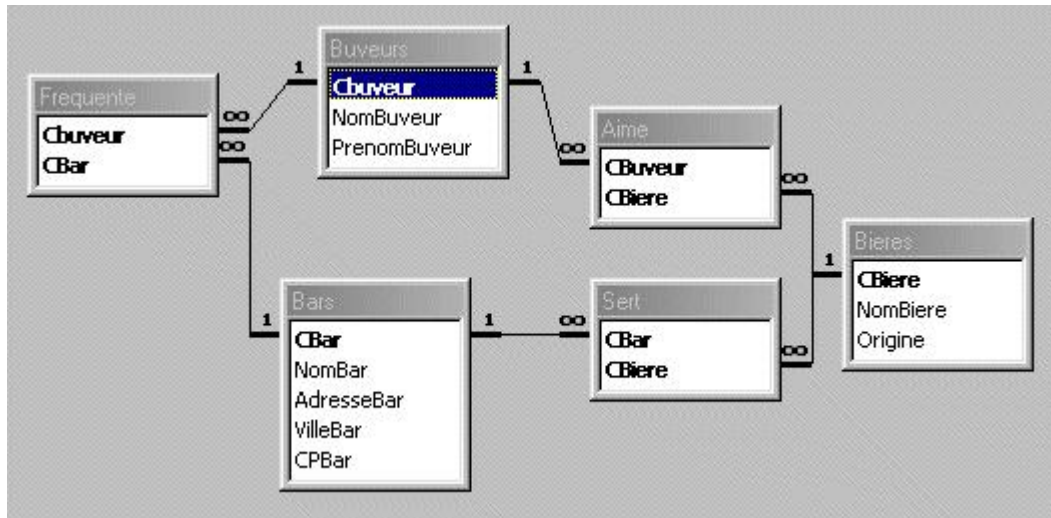


Figure 2.3 : Modèle relation de la BD Bars et Buveurs de Bières

CBar	NomBar	AdresseBar	VilleBar	CPBar
1	Lulu	2 rue des roses	Besançon	25000
2	Les loups	rue G. Courbet	Besançon	25000

Table 2.13 : Table BAR

CBar	CBiere
1	1
1	2
2	1
2	2
2	3

Table 2.14 : Table SERT

CBiere	NomBiere	Origine
1	Grimbergen	Belgique
2	1662	Hollande
3	Samuel Smith	Irlande

Table 2.15 : Table BIÈRE

Cbuveur	CBar
1	1
1	2
2	1
3	1
3	2

Table 2.16 : Table FREQUENTE

Question 1

Les noms des bars qui servent une bière appréciée par 'Dupont'.

Cbuveur	NomBuveur	PrenomBuveur
1	Dupont	Jean
2	Durand	Pierre
3	Maire	Jean

Table 2.17 : Table BUVEUR

CBuveur	CBiere
1	1
1	2
2	3
3	2

Table 2.18 : Table AIME

Question 2

Les noms des buveurs qui fréquentent au moins un bar où l'on sert une bière qu'ils aiment.

Question 3

Les noms des buveurs qui fréquentent tous les bars.

Question 4

Les numéros des buveurs qui fréquentent tous les bars qui servent au moins une bière qu'ils aiment.

Question 5

Les noms des buveurs qui ne fréquentent que les bars qui servent une bière qu'ils aiment.

Question 6

Les numéros des buveurs qui ne fréquentent aucun bar où l'on sert une bière qu'ils aiment.

Question 7

Donner pour chaque buveur, leur numéro et le nombre de bars servant une bière qu'ils aiment.

Question 8

Les numéros des buveurs qui fréquentent au moins deux bars où l'on sert une bière qu'ils aiment.

Question 9

Les numéros des buveurs qui vont dans au moins un bar fréquenté par Dupont.

Chapter 3

Exercices de la partie 4 : Normalisation

3.1 Exercice 1 - Partie 4 : Scolarité

On considère la relation SCOLARITE suivante :

SCOLARITE(NumEtudiant, NumUE, NbreHeureEnseignementUE, NumSSEnseignant, NomEnseignant, PrenomEnseignant, Grade, Indice, Année, UFRRattachement, NbreEnseignantUFR, ResultatUE, NomEtudiant, PrenomEtudiant)

Question 1

- R1 : R1 : Une unité d'enseignement, caractérisée par un numéro (NumUE) a un nombre d'heures d'enseignement
- R2 : Chaque UE est assurée par un enseignant
- R3 : Un enseignant qui est caractérisé par un numéro de sécurité sociale (NumSSEnseignant) a un nom, un prénom et un grade
- R4 : Un étudiant, caractérisé par son numéro (NumEtudiant) a un nom, un prénom
- R5 : Un étudiant peut s'inscrire à plusieurs UE
- R6 : Un enseignant est rattaché à une seule UFR (UFRRattachement)
- R7 : Pour un étudiant et une UE on a un résultat (ResultatUE)
- R8 : Un grade correspond à un indice
- R9 : Une UFR a un certain nombre d'enseignants (NbreEnseignantUFR).

Question 2

Définir la clé de la relation SCOLARITE.

3.2 Exercice 2 - Partie 4 : Père Mère et Enfants

On considère la relation PME suivante :

PME(NomH, PrenomH, NaissanceH, NomF, PrenomF, NaissanceF, NomE, PrenomE, NaissanceE, DateMariage),
telle que le t-uplet :

(Dupont, Jean, 12/11/1940, Jeancler, Marie, 28/10/1942, Dupont, Marc, 23/04/1964, 23/03/1963) a la signification suivante :

Jean Dupont né le 12/11/1940 a épousé Marie Jeancler née le 28/10/1942, le 23/03/1963, et ils ont eut un enfant : Marc Dupont né le 23/04/1964.

On suppose que les attributs **nom** et **prénom** définissent une seule personne. Une analyse de cette relation nous fournit un ensemble initial E de dépendances fonctionnelles :

- DF1 : NomH, PrenomH \rightarrow NaissanceH
- DF2 : NomF, PrenomF \rightarrow NaissanceF
- DF3 : NomE, PrenomE \rightarrow NaissanceE, NomH, PrenomH, NomF, PrenomF
- DF4 : NomF, PrenomF, NomH, PrenomH \rightarrow DateMariage

Question 1 :

Quelles sont les clés candidates de PME ?

Question 2 :

PME est-elle en 2NF ? Si non, la transformer en utilisant le théorème de décomposition.

Question 3 :

PME est-elle en 3NF ? Si non, la transformer en utilisant le théorème de décomposition.

3.3 Exercice 3 - Partie 4 : Professeurs - Cours et Matières

Soit la relation COURS (Matiere, Classe, Prof) complétée par les règles suivantes :

- un professeur n'enseigne qu'une matière
- une classe n'a qu'un seul professeur par matière

Matiere	Prof	Classe
Maths	Monigadon	6B
Maths	Quere	5A
Dessin	Maire	5A
Dessin	Maire	6B

Question 1 :

Quelles sont les dépendances fonctionnelles de cette relation?

Question 2 :

Quelles sont les clés candidates de la relation COURS ?

Question 3 :

Prenons (Classe, Prof) comme clé primaire. Montrer que cette relation n'est pas 2NF, et la transformer.

Question 4 :

Prenons (Classe, Matiere) comme clé primaire. Montrer que cette relation n'est pas BCNF, et la transformer.

3.4 Exercice 4 - Partie 4 : Enfants et contrats

On considère la relation MEMBRE suivante :

(N_Membre, NomMembre, PrenomMembre, N_Enfant, N_Contrat, CodeContrat, TypeContrat) telle que le t-uplet $(NM, nom, pre, NE, NC, CC, TC)$ signifie que le membre de Numéro NM qui s'appelle nom , pre , a un enfant qui a pour numéro : NE , et a un contrat qui a pour numéro NC , pour code CC , et pour type TC .

avec les dépendances fonctionnelles :

- DF1 : N_Membre \rightarrow NomMembre, PrenomMembre
- DF2 : N_Contrat \rightarrow CodeContrat
- DF3 : CodeContrat \rightarrow TypeContrat

Un même membre peut avoir plusieurs enfants et plusieurs contrats, comme le montre l'exemple de relation MEMBRE 3.1 .

N_Membre	Nom Membre	Prenom Membre	N_Enfant	N_Contrat	Code Contrat	Type Contrat
1	Dupont	Jean	3	011	AssV1	AssurVoiture
1	Dupont	Jean	3	012	AssV2	AssurVoiture
1	Dupont	Jean	3	123	AssVi	AssuranceVie
1	Dupont	Jean	4	011	AssV1	AssurVoiture
1	Dupont	Jean	4	012	AssV2	AssurVoiture
1	Dupont	Jean	4	123	AssVi	AssuranceVie
2	Dupont	Marie	3	124	AssVi	AssuranceVie
2	Dupont	Marie	4	124	AssVi	AssuranceVie

Table 3.1 : relation MEMBRE

Question 1

Définir les dépendances fonctionnelles de la relation MEMBRE, et en déduire sa clé candidate.

Question 2

Montrer que la relation MEMBRE n'est pas 2NF. Et la transformer en relations 2NF.

Question 3

Montrer que l'ensemble de relations obtenu n'est pas 3NF. Et le transformer en relations 3NF.

Question 4

Montrer que l'ensemble de relations obtenu n'est pas 4NF. La transformer afin qu'elle soit 4NF.

3.5 Exercice 5 - Partie 4 : Maîtresse de maison

Une maîtresse de maison s'est constitué une base de données sur les personnes qu'elle invite et les plats qu'elle leur sert. Cette base de données est composée d'une relation : LESREPAS

LESREPAS(Date, Plat, Invite, Preference)

Qui précise qu'elle a servi tel plat à telle date, avec tel invité qui aime tel plat.

Date	Plat	Personne	Preference
03/10/2009	gigot	alice	sorbet
03/10/2009	salade niçoise	alice	sorbet
03/10/2009	sorbet	alice	sorbet
03/10/2009	gigot	pierre	foie gras
03/10/2009	salade niçoise	pierre	foie gras
03/10/2009	sorbet	pierre	foie gras
03/10/2009	gigot	jean	dinde
03/10/2009	gigot	jean	soupe
03/10/2009	salade niçoise	jean	dinde
03/10/2009	salade niçoise	jean	soupe
03/10/2009	sorbet	jean	dinde
03/10/2009	sorbet	jean	soupe
03/10/2009	gigot	marie	lotte à l'américaine
03/10/2009	salade niçoise	marie	lotte à l'américaine
03/10/2009	sorbet	marie	lotte à l'américaine
26/12/2009	dinde	jean	dinde
26/12/2009	dinde	jean	soupe
26/12/2009	foie gras	jean	dinde
26/12/2009	foie gras	jean	soupe
26/12/2009	sorbet	jean	dinde
26/12/2009	sorbet	jean	soupe
26/12/2009	dinde	pierre	foie gras
26/12/2009	foie gras	pierre	foie gras
26/12/2009	sorbet	pierre	foie gras

Table 3.2 : Relation LESREPAS

Il n'y a aucune dépendance fonctionnelle dans cette relation.

Question 1

Quelles sont les clés candidates de R ?

Question 2

LESREPAS est elle en 2NF ? Si non, la transformer en utilisant le théorème de décomposition.

Question 3

Les relations obtenues sont-elles en 3NF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 4

Les relations obtenues sont-elles en BCNF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 5

Les relations obtenues sont-elles 4NF ? Si non les transformer en utilisant le théorème de FAGIN.

3.6 Exercice 6 - Partie 4 : Entreprise

Une Entreprise a des employés. Les employés sont affectés à des tâches et ont des enfants. On considère la relation universelle :

UNIV(NEntreprise, NomEntreprise, RueEntreprise, VilleEntreprise, CPEntreprise, NEmploye, NomEmploye, NTache, NEnfant, NResponsable, NomResponsable)

Avec les dépendances fonctionnelles suivantes :

N° Employe	Nom Employe	N° Entreprise	Nom Entreprise	Rue Entreprise	Ville Entreprise	CP Entreprise	N° Responsable	Nom responsable	N° Enfant	N° Tache
E1	DUPONT	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE1	T1
E1	DUPONT	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE1	T2
E2	DURAND	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE2	T1
E2	DURAND	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE3	T1
E3	DUBOIS	ENT2	OLSA	Grande Rue	PONTARLIER	25500	R4	ANDREY	EE4	T1
E3	DUBOIS	ENT2	OLSA	Grande Rue	PONTARLIER	25500	R4	ANDREY	EE4	T3

- DF1 : NEntreprise → NomEntreprise, RueEntreprise, VilleEntreprise, CPEntreprise, NResponsable
- DF2 : NEmploye → NomEmploye, NEntreprise
- DF3 : NResponsable → NomResponsable

Question 1

Définir la clé de la relation UNIV.

Question 2

La relation UNIV est elle en 2NF ? Si non, la transformer en utilisant le théorème de décomposition, en un ensemble de relations 2NF.

Question 3

L'ensemble des relations obtenues sont-elles en 3NF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 4

Les relations obtenues sont-elles en BCNF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 5

L'ensemble des relations obtenues sont-elles en 4NF ? Si non, les transformer en utilisant le théorème de Fagin.

3.7 Exercice 7 - Partie 4 : Fournisseurs, Pièces et projets

On considère la relation suivante :

UNIV(Fournisseur, Nom, Rue, CP, Ville, Piece, LibPiece, Projet, LibProjet, Responsable)

Avec les dépendances fonctionnelles suivantes :

- DF1 : Fournisseur \rightarrow Nom, Rue, CP
- DF2 : Piece \rightarrow LibPiece
- DF3 : Projet \rightarrow LibProjet, Responsable
- DF4 : CP \rightarrow Ville

Fournisseur	Nom	Rue	CP	Ville	Piece	LibPiece	Projet	LibProjet	Responsable
F123	Dupont	Rue des roses	25000	Besançon	P900	Boulon R	PRO23	Rénovation Usine2	Bailly
F123	Dupont	Rue des roses	25000	Besançon	P450	Vis	PRO23	Rénovation Usine2	Bailly
F123	Dupont	Rue des roses	25000	Besançon	P901	Boulon R1	PRO34	Réacteur XT6	Horlow
F555	Fager	Rue grenelle	21000	Dijon	P900	Boulon R	PRO23	Rénovation Usine2	Bailly
F555	Fager	Rue grenelle	21000	Dijon	P452	Vis 2	PRO23	Rénovation Usine2	Bailly
F555	Fager	Rue grenelle	21000	Dijon	P900	Boulon R	PRO77	Zanax	Hergé

Question 1

Définir la clé de la relation UNIV.

Question 2

La relation UNIV est-elle en 2NF ? Si non, la transformer en utilisant le théorème de décomposition, en un ensemble de relations 2NF.

Question 3

Les relations obtenues sont-elles en 3NF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 4

Les relations obtenues sont-elles en BCNF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 5

L'ensemble des relations obtenues sont-elles en 4NF ? Si non, les transformer en utilisant le théorème de Fagin.

Question 6

Les relations obtenues sont-elles en 5NF ? Si non, les transformer.

3.8 Exercice 8 - Partie 4 : Les magasins, leurs produits et leurs fournisseurs

On considère la relation universelle :

UNIV(NumProduit, Nomproduit, NumRayon, NomRayon, NumFournisseur, NomFournisseur, AdresseFournisseur, NumMagasin, NomMagasin, VilleMagasin)

Avec les dépendances fonctionnelles suivantes :

- DF1 : NumProduit ' Nomproduit, NumRayon
- DF2 : NumRayon ' NomRayon
- DF3 : NumFournisseur ' NomFournisseur, AdresseFournisseur
- DF4 : NumMagasin ' NomMagasin, VilleMagasin

La clé de cette relation est : NumProduit, NumFournisseur, NumMagasin

Question 1

La relation UNIV est elle en 2NF ? Si non, la transformer en utilisant le théorème de décomposition, en un ensemble de relations 2NF.

Question 2

Les relations obtenues sont-elles en 3NF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 3

Les relations obtenues sont-elles en BCNF ? Si non, les transformer en utilisant le théorème de décomposition.

Question 4

Décrire en termes de dépendances multivaluées ou de dépendances de jointure les règles de gestion suivantes :

Scénario 1

- Un produit peut être réalisé par plusieurs fournisseurs
- Un produit peut être à la vente dans plusieurs magasins

Scénario 2

- Un produit peut être réalisé par plusieurs fournisseurs
- Un produit peut être à la vente dans plusieurs magasins
- Un magasin a un ensemble de fournisseurs spécifiques

Scénario 3

- Un produit peut être réalisé par plusieurs fournisseurs
- Un produit peut être à la vente dans plusieurs magasins
- Un magasin peut choisir pour chaque produit ses fournisseurs (dans la liste des fournisseurs de ce produit évidemment).

Pour chaque scénario, en fonction des dépendances mises en évidence proposez, si nécessaire, une décomposition de la relation concernée.

Part II

Corrigés des exercices

Chapter 4

Exercices de la Partie 2 : Modélisation relationnelle

Ce document contient le corrigé des énoncés d'exercice. Pour chaque exercice vous trouverez un rappel de l'énoncé ainsi qu'une première partie **Indications** qui pour chaque question vous donne des éléments pour répondre à la question, et ensuite le corrigé de chaque question.

4.1 Exercice 1 - Partie 2 : Le Centre Médical

Un centre médical est constitué de plusieurs médecins. Lors d'une consultation qui met en présence un seul médecin et un seul patient, plusieurs médicaments peuvent être prescrits. Un médecin a plusieurs patients et un même patient peut avoir plusieurs médecins.

- MEDECIN(*Matricule, Nom)
- CONSULTATION (*Numcons, Date, Matricule, NumSS)
- PRESCRIPTION (*Code, *Numcons, NbPrises)
- MEDICAMENT (*Code, Libelle)
- PATIENT (*NumSS, Nom)

Le modèle relationnel est donné par la figure

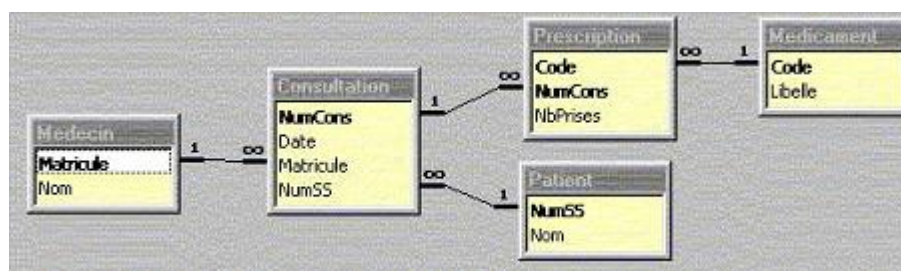


Figure 4.1 : Le modèle relationnel du centre médical

les indications

Question 1, 2 : L'objectif de cette question est de vous faire manipuler un modèle relationnel : comprendre comment des informations sont stockées dans les relations et comment on peut les retrouver.

Question 3 : L'objectif de cette question est toujours de vous faire manipuler un modèle

relationnel : comprendre comment des informations sont stockées dans les relations et comment on peut les retrouver. Ici nous allons passer du monde réel, aux données dans la base. Au fait dans la relation CONSULTATION quel numéro de consultation avez-vous donné à la consultation de Mr Dupont ? 5 ? Et pourquoi pas 3426 ?

Question 4 : L'objectif de cette question est toujours de vous faire manipuler un modèle relationnel : comprendre comment des informations sont stockées dans les relations et comment on peut les retrouver. Ici nous allons passer du monde réel, aux données dans la base. Au fait avez-vous déjà saisi le t-uplet dans la relation CONSULTATION et ensuite dans la relation PRESCRIPTION ou fait le contraire. L'ordre de saisie des différents t-uplets correspondant à cette question a-t-il de l'importance ?

Question 5 : L'objectif de cette question est toujours de vous faire manipuler un modèle relationnel. Nous nous intéressons ici à la structure des relations de la base, à savoir les clés primaires des relations et les attributs qui composent ces relations. Le contenu des relations peut être un indicateur pour confirmer votre réponse mais en aucun cas une justification. Pour répondre à une telle question, il faut être capable de dire si d'après le modèle proposé, une même valeur de NumSS peut apparaître plusieurs fois dans la relation CONSULTATION.

Question 6 : Nous nous intéressons toujours ici à la structure des relations de la base, à savoir les clés primaires des relations et les attributs qui composent ces relations. Le contenu des relations peut être un indicateur pour confirmer votre réponse mais en aucun cas une justification. Pour répondre à une telle question, il faut être capable de dire si d'après le modèle proposé, une même valeur de NumCons peut apparaître plusieurs fois dans la relation CONSULTATION.

Question 7 : Cette question est similaire aux questions 5, 6 et 8. Pour répondre à une telle question, il faut être capable de dire si d'après le modèle proposé, une même valeur de NumCons peut apparaître plusieurs fois dans la relation PRESCRIPTION.

Question 8 : Cette question est plus difficile que les précédentes. Il n'y a pas vraiment de "lien" dans le modèle entre les médecins et les médicaments.

Question 9 : Cette question permet de voir comment une modification du monde réel se répercute sur le modèle relationnel. Pour prendre en compte le fait qu'un patient ne puisse plus avoir qu'un seul médecin il faut modifier la structure de certaines relations et certains liens entre ces relations.

Question 10 : L'objectif de cette question est de comprendre à quoi sert la définition d'un lien entre deux relations, et ses implications. Ici on s'intéresse à la suppression d'un t-uplet dans une relation (la relation CONSULTATION). La question à se poser est : la clé primaire de cette relation est-elle utilisée comme clé étrangère dans d'autres relations ? Si oui, quelle est la politique de Maj pour la destruction ? Et en fonction de cette politique de mise à jour le SGBD va-t-il autoriser ou non la suppression du t-uplet.

Question 11 : Nous avons exactement le même type de question que pour la question 10. Seule la politique de maj est différente. L'objectif de cette question est de comprendre à quoi sert la définition d'un lien entre deux relations, et ses implications. Ici on s'intéresse à la suppression d'un t-uplet dans une relation (la relation CONSULTATION). La question à se poser est : la clé primaire de cette relation est-elle utilisée comme clé étrangère dans d'autres relations ? Si oui, quelle est la politique de Maj pour la destruction ? Et en fonction de cette politique de mise à jour le SGBD va-t-il autoriser ou non la suppression du t-uplet.

Question 12 : L'objectif de cette question est de comprendre à quoi sert la définition d'un lien entre deux relations, et ses implications. Ici on s'intéresse à la modification d'un t-uplet dans une relation (la relation CONSULTATION). La modification du t-uplet implique une modification de la valeur de la clé de la relation CONSULTATION. La première question à se poser est alors : Y-a-t-il duplication de la clé primaire ? Si oui la modification est rejetée par le SGBD. Si non, comme pour la suppression il faut se poser la question : la clé primaire de cette relation est-elle utilisée comme clé étrangère dans d'autres relations ? Si oui, quelle est la politique de Maj pour la modification ? Et en fonction de cette politique de mise à jour le SGBD va-t-il autoriser ou

non la modification du t-uplet.

Question 13 : Nous avons exactement le même type de question que pour la question 12. Seule la politique de maj est différente.

Question 14 : Nous avons exactement le même type de question que pour la question 12. Ici c'est la question : Y-a-t-il duplication de la clé primaire ? Qui pose problème.

Question 15 : L'objectif de cette question est de comprendre à quoi sert la définition d'un lien entre deux relations, et ses implications. Ici nous nous intéressons à l'ajout d'une donnée dans une relation. Avant toute chose il faut vérifier que l'on respecte la règle d'unicité de la clé de notre relation. Ensuite, s'il n'y a pas de problème de clé, la question à se poser ici est : le t-uplet que j'ajoute fait-il référence à des données dans d'autres relations ? (en d'autres termes : y-a-t-il des liens d'autres relations vers la relation dans laquelle j'ajoute une donnée ?) Si oui il faut vérifier que les clés étrangères sont unifiables.

Question 1

En regardant le contenu des relations. Dire quel médecin, Dupont a rencontré lors de sa consultation, et que lui a-t-il prescrit ?

En regardant le contenu de la relation (ou table) PATIENT, on peut trouver le numéro de sécurité sociale du patient Dupont, à savoir : 1581225128745.

Ensuite en regardant le contenu de la relation CONSULTATION, on y trouve que Dupont est le patient de la consultation 1, qui a été réalisée par le médecin de matricule 1, c'est-à-dire Mr Pouthier.

La relation PRESCRIPTION nous donne alors le détail de ce que le médecin a prescrit à Monsieur Dupont, Il suffit de repérer les t-uplets pour lesquels NumCons = 1, c'est -à dire : Les t-uplets avec comme code médicament 1236 (2 prises par jour), et 1478 (1 prise par jour).

Ces médicaments sont Le doliprane et le primpéran (informations trouvées avec la table PRESCRIPTION).

Question 2

En regardant le contenu des relations. Dire quel patient a déjà eu du Primpéran.

En regardant le contenu de la relation MEDICAMENT, on trouve le code du primpéran, à savoir 1478. Ensuite la relation PRESCRIPTION nous permet de savoir que lors de la consultation 1 du primpéran a été prescrit, enfin la relation CONSULTATION nous donne le numéro de SS du patient qui a été à la consultation 1 : Il s'agit de Monsieur DUPONT.

Question 3

Modifier les relations pour pouvoir ajouter l'information suivante : Monsieur DUPONT est allé voir en consultation, le 21/06/99, Mr Receveur, et celui-ci lui a prescrit 2 prises d'aspegic 500 ainsi qu'une prise de vogalène.

Il s'agit d'une nouvelle consultation : nous lui donnerons le numéro 5 (n'importe quel numéro non utilisé dans la relation CONSULTATION).

Dans la relation CONSULTATION le t-uplet suivant est ajouté : (5, 21/06/99, 2, 1581225128745)

De plus dans la relation PRESCRIPTION les deux t-uplets suivants sont ajoutés : (1456, 5, 2) et (2589, 5, 1)

Les autres relations restent inchangées.

				Médecin	
				Matricule	Nom
				1	Pouthier
				2	Receveur
				3	Pourcelot
Patient		Consultation			
NumSS	Nom	NumCons	Date	Matricule	NumSS
45	Dupont	1	04-juin-99	1	1581225128745
56	Faivre	2	05-juin-99	1	2781024147456
		3	10-nov-99	2	2781024147456
		4	10-sept-02	1	2781024147456
Prescription			Médicament		
Code	NumCons	NbPrises	Code	Libelle	
1236	1	2	1236	Doliprane	
1236	2	1	1456	Aspegic 500	
1236	3	2	1478	Primpéran	
1478	1	1	2589	Vogalène	

Médicament		Prescription		
Code	Libelle	Code	NumCons	NbPrises
1236	Doliprane	1236	1	2
1456	Aspegic 500	1236	2	1
1478	Primpéran	1236	3	2
2589	Vogalène	1478	1	1

Patient		Consultation			
NumSS	Nom	NumCons	Date	Matricule	NumSS
45	Dupont	1	04-juin-99	1	1581225128745
56	Faivre	2	05-juin-99	1	2781024147456
		3	10-nov-99	2	2781024147456
		4	10-sept-02	1	2781024147456

Question 4

Modifier les relations pour pouvoir ajouter l'information suivante : Melle Dholat a consulté Mr Pouthier le 21/06/00, qui lui a prescrit du magnogène (2 prises par jour) et du doliprane (1 prise par jour). Le numéro de SS de Melle Dholat est : 2701175034068.

Comme précédemment nous créons une nouvelle consultation à laquelle nous donnons le numéro 5. De plus ici nous créons un t-uplet dans la relation PATIENT : (2701175034068, Dholat).

Et un t-uplet dans la relation MÉDICAMENT :

Dans l'énoncé je n'ai donné aucune valeur pour le code du magnogène. Je voulais que les étudiants

Patient		Consultation			
NumSS	Nom	NumCons	Date	Matricule	NumSS
745	Dupont	1	04-juin-99	1	1581225128745
56	Faivre	2	05-juin-99	1	2781024147456
		3	10-nov-99	2	2781024147456
		4	10-sept-02	1	2781024147456
		5	21-juin-99	2	1581225128745

Prescription		
Code	NumCons	NbPrises
1236	1	2
1236	2	1
1236	3	2
1478	1	1
1456	5	2
2589	5	1

se rendent compte par eux-même qu'ils n'avaient pas toutes les informations. A partir de là pour créer le t-uplet correspondant au magnogène, il faut créer un code. Celui-ci peut à priori prendre n'importe quelle valeur (dans le domaine adéquat bien sur) sauf celles existantes dans la relation. Nous prendrons la valeur 1111. (1111, Magnogène)

Dans la relation CONSULTATION le t-uplet suivant est ajouté : (5, 21/06/00, 1, 2701175034068)
De plus dans la relation PRESCRIPTION les deux t-uplets suivants sont ajoutés : (1111, 5, 2) et (1236, 5, 1)

Les autres relations restent inchangées.

Attention l'ordre dans lequel les informations sont mises dans la base de données a une importance. Nous ne saisissons pas le t-uplet (1111, 5, 2) dans la relation PRESCRIPTION avant d'avoir saisi le t-uplet (1111, Magnogène) dans la relation MEDICAMENT.

Question 5

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Un patient peut-il effectuer plusieurs consultations ?

La clé primaire de la relation CONSULTATION est NumCons, ce qui signifie qu'une même valeur de NumSS peut apparaître plusieurs fois avec chaque fois bien évidemment un numéro de consultation différent.

UN PATIENT PEUT DONC REALISER PLUSIEURS CONSULTATIONS.

Remarque : Le lien [1..n] entre PATIENT et CONSULTATION est une conséquence du fait qu'un numéro de patient peut apparaître plusieurs fois dans la table CONSULTATION, mais pas une cause.

Question 6

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Un médecin peut-il recevoir plusieurs clients dans une même consultation ?

Pour pouvoir recevoir plusieurs patients a une même consultation, il faudrait pouvoir stocker dans la relation CONSULTATION plusieurs t-uplets avec le même numéro de consultation et des numéros de SS différents.

OR CECI EST IMPOSSIBLE CAR NUMCONS EST CLE PRIMAIRE DE LA RELATION CONSULTATION, UNE MEME VALEUR DE NUMCONS NE PEUT DONC PAS APPARAÎTRE PLUSIEURS FOIS.

Question 7

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Plusieurs médicaments peuvent-ils être prescrit lors d'une même consultation ?

A priori OUI puisque dans la relation PRESCRIPTION nous avons déjà traité le cas de consultations donnant lieu à la prescription de plusieurs médicaments.

Maintenant si nous regardons la clé de la relation PRESCRIPTION, il s'agit de (NumCons, Code) : on peut donc avoir plusieurs fois le même numéro de consultation.

Question 8

En tenant compte du modèle relationnel proposé, répondre à la question suivante en justifiant votre réponse : Deux médecins différents peuvent-ils prescrire le même médicament ?

Deux médecins peuvent prescrire le même médicament. En effet dans la relation PRESCRIPTION plusieurs t-uplets peuvent avoir la même valeur pour l'attribut Code, sans aucun problème, pour des valeurs de NumCons quelconque et donc correspondant à priori à des médecins différents.

Question 9

La sécurité sociale impose à chaque patient de n'avoir qu'un seul médecin généraliste. Modifier le modèle relationnel pour tenir compte de cette nouvelle contrainte. Si chaque patient ne peut plus avoir qu'un seul médecin, les consultations concernant un patient X donné, feront intervenir toujours le même médecin. Ce n'est donc plus aux consultations qu'il faut lier les médecins mais aux patients.

Le nouveau modèle est :

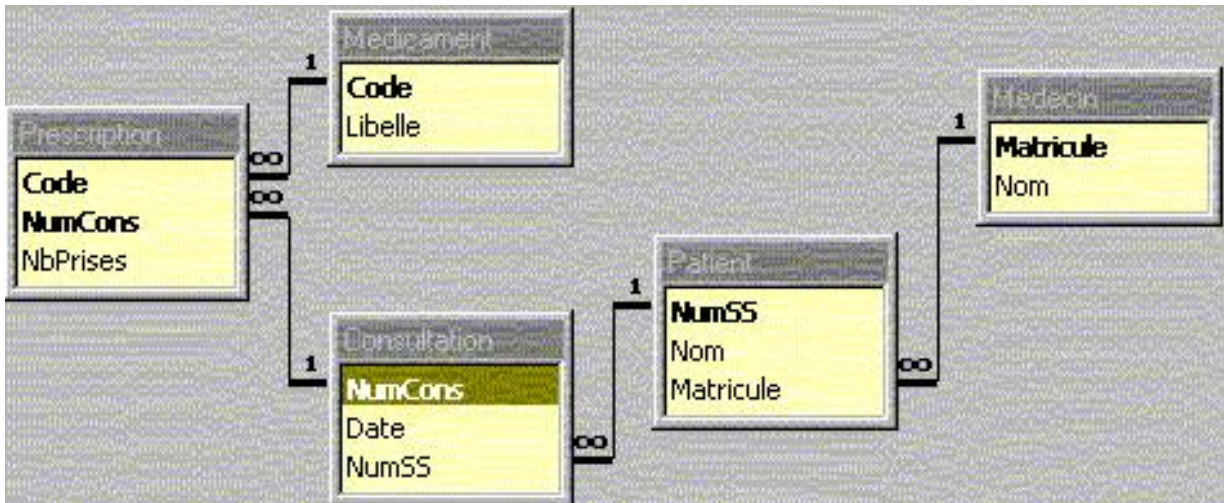
- PATIENT (*NumSS, Nom, Matricule)
- CONSULTATION(*NumCons, Date, NumSS)
- et PATIENT [NumSS] (1..n) -> [NumSS] Consultation.
- Et MEDECIN [Matricule] (1..n) -> [Matricule] Patient.

Question 10

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type CASCADES. Que se passe-t-il si on supprime de la table CONSULTATION le t-uplet correspondant à la consultation 2. Le t-uplet (2, 05/06/99,1, 2781024147456) est supprimé de la table CONSULTATION, de plus le numéro de consultation 2 étant référencé dans la table PRESCRIPTION le t-uplet (1236, 2, 1) est supprimé de cette table.

Question 11

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type RESTRICTED. Que se passe-t-il si on supprime de la table CONSULTATION le t-uplet correspondant à la consultation 2.



Consultation					Prescription		
NumCons	Date	Matricule	NumSS		Code	NumCons	NbPrises
1	04-juin-99	1	1581225128745		1236	1	2
2	05-juin-99	1	2781024147456	→	1236	2	1
3	10-nov-99	2	2781024147456		1236	3	2
4	10-sept-02	1	2781024147456		1478	1	1

↓

Consultation					Prescription		
NumCons	Date	Matricule	NumSS		Code	NumCons	NbPrises
1	04-juin-99	1	1581225128745		1236	1	2
3	10-nov-99	2	2781024147456		1236	3	2
4	10-sept-02	1	2781024147456		1478	1	1

Le t-uplet correspondant à la consultation 2 ne peut pas être supprimé car il est utilisé dans la table PRESCRIPTION. La base reste inchangée.

Question 12

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type CASCADES. Que se passe-t-il si on modifie dans la table CONSULTATION le t-uplet correspondant à la consultation 2, en le passant au numéro 5.

Le t-uplet (2, 05/06/99,1, 2781024147456) est modifié dans la table CONSULTATION, il devient (5, 05/06/99,1, 2781024147456)

de plus le numéro de consultation 2 étant référencé dans la table PRESCRIPTION le t-uplet (1236, 2, 1) est modifié dans cette table en (1236, 5, 1).

Question 13

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type RESTRICTED. Que se passe-t-il si on modifie dans la table CONSULTATION le t-uplet correspondant à la consultation 2, en le passant au numéro 5.

Consultation				Prescription		
NumCons	Date	Matricule	NumSS	Code	NumCons	NbPrises
1	04-juin-99	1	1581225128745	1236	1	2
2	05-juin-99	1	2781024147456	1236	2	1
3	10-nov-99	2	2781024147456	1236	3	2
4	10-sept-02	1	2781024147456	1478	1	1
↓						
Consultation				Prescription		
NumCons	Date	Matricule	NumSS	Code	NumCons	NbPrises
1	04-juin-99	1	1581225128745	1236	1	2
5	05-juin-99	1	2781024147456	1236	5	1
3	10-nov-99	2	2781024147456	1236	3	2
4	10-sept-02	1	2781024147456	1478	1	1

Le t-uplet correspondant à la consultation 2 ne peut pas être modifié car il est utilisé dans la table PRESCRIPTION. La base reste inchangée.

Question 14

On suppose que l'on a défini une règle de mise à jour concernant les opérations de modification ou de suppression des données de type CASCADES. Que se passe-t-il si on modifie dans la table CONSULTATION le t-uplet correspondant à la consultation 2, en le passant au numéro 3.

Cette modification n'est pas possible, car elle invaliderait la règle d'unicité des valeurs de clé. En effet on aurait deux t-uplets dans CONSULTATION qui auraient la même valeur pour NumCons (3) ce qui est impossible, car NumCons est la clé de cette relation. La base reste inchangée.

Question 15

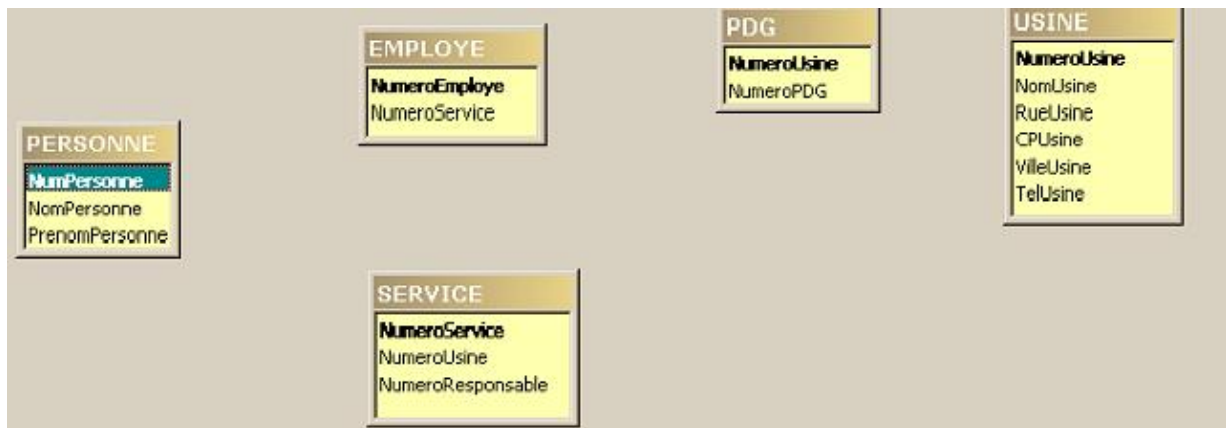
Que se passe-t-il si on essaie d'ajouter le t-uplet suivant dans la table PRESCRIPTION (1111, 2, 2) ?

Cet ajout est refusé car il n'existe pas dans la table MEDICAMENT de t-uplet ayant pour valeur de l'attribut Code la valeur 1111. L'ajout est impossible car on ne respecte pas la règle d'intégrité référentielle associée au lien entre les relations MEDICAMENT et PRESCRIPTION. La base reste inchangée.

4.2 Exercice 2 - Partie 2 : La multinationale

Pour décrire les informations concernant les services, les employés, les responsables et les PDG des usines d'une multi-nationale, on propose de créer une base de données MULTINATIONALE.

- EMPLOYE (*NumeroEmploye, NumeroService)
- PDG (*NumeroUsine, NumeroPDG)
- PERSONNE (*NumPersonne, NomPersonne, PrenomPersonne)
- SERVICE (*NumeroService, Usine, NumeroResponsable)
- USINE (*NumeroUsine, NomUsine, RueUsine, VilleUsine, CPUusine, TelUsine)



Indications

Question 1 : Prendre la clé primaire de chaque relation. Puis, rechercher ce groupe d'attributs dans les autres relations, ou un groupe d'attributs qui ne portent pas le même nom mais prennent leurs valeurs dans les mêmes domaines.

Question 2 : Il faut tout d'abord définir le sens du lien et ensuite son type (1..1 ou 1..n).

Question 3 : Nous avons ici le cas particulier d'attributs qui prennent leurs valeurs dans le même domaine, mais qui jouent des rôles différents dans la base. D'autre part plusieurs de ces attributs sont clés primaires de la relation où ils se trouvent.

Question 4 : Nous traitons ici le cas particulier de lien 1..1. Dans quel sens faut-il le définir ? Et dans le cas où cet attribut est utilisé dans d'autres relations avec des liens de type 1..N, quels sont les liens à définir ? (depuis toutes les relations liées par des liens 1..1 ou depuis seulement une ?)

Question 5 : Dans cet exercice nous avons abordé les questions à se poser pour définir correctement les liens à partir de la définition d'un ensemble de relations.

Question 1

L'objectif final de cet exercice est de définir les liens dans le modèle proposé. Pour cette première question trouvez entre quelles relations on peut définir des liens.

Commençons par le cas le plus simple, l'attribut NumService apparaît dans les relations SERVICE et EMPLOYE. On définira donc un lien entre ces deux relations. Considérons ensuite l'attribut NumUsine qui apparaît dans les relations USINE, SERVICE et PDG. Nous pouvons avoir ici à priori des liens entre ces trois relations. Enfin si l'on considère les attributs NumPersonne, NumeroEmploye, NumeroPDG et NumeroResponsable. Ces quatre attributs ont le même domaine à savoir des numéros qui permettent d'identifier une "personne".

Question 2

Définir le ou les liens qui utilisent l'attribut NumService.

Le lien va de la clé primaire vers la clé étrangère, c'est à dire de la relation SERVICE vers la relation EMPLOYE. Pour préciser le type du lien il faut se poser la question suivante : L'attribut NumService peut-il prendre plusieurs fois la même valeur dans EMPLOYE ? OUI car qu'il n'est pas clé candidate de la relation EMPLOYE. Le lien est donc de type 1..N SERVICE [NumService] (1..n) → [NumService] EMPLOYE



Question 3

Définir les liens qui mettent en jeu les attributs NumPersonne, NumeroEmploye, NumeroPDG et NumeroResponsable.

NumeroPDG et NumeroResponsable ne sont pas clés candidates. Ces deux attributs ne peuvent donc pas servir de source à des liens, ce sont des clés étrangères.

Regardons les attributs NumPersonne et NumeroEmploye. Ce sont deux clés primaires. Ainsi le lien qui lie ces deux attributs est de type 1..1 car de part et d'autre ces deux attributs ne peuvent pas prendre plusieurs fois la même valeur.

Le problème ici est de définir le sens de ce lien. Quelle est la relation la "plus importante" (ici nous avons une vision plus application que modèle) ? Doit-on créer une personne et ensuite l'employé ou au contraire créer un employé et ensuite la personne ? Nous choisissons ici comme relation source du lien la relation PERSONNE.

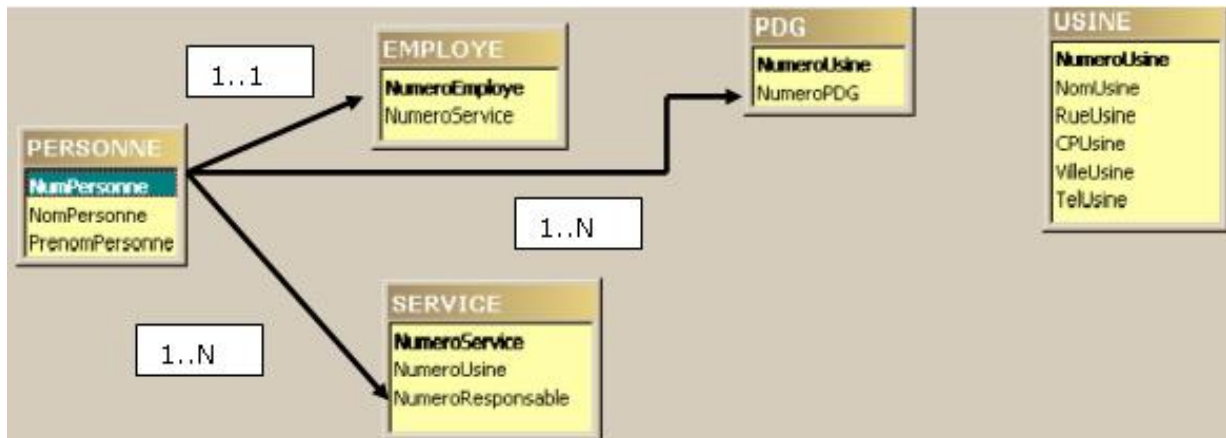
Ce qui nous donne les liens suivants :

- PERSONNE [NumPersonne] (1..1) -> [NumeroEmploye] EMPLOYE
- PERSONNE [NumPersonne] (1..N) -> [NumeroPDG] PDG
- PERSONNE [NumPersonne] (1..N) -> [NumeroResponsable] SERVICE

Question 4

Définir les liens qui mettent en jeu l'attribut NumUsine.

NumUsine est la clé primaire des relations USINE et PDG. Nous aurons donc un lien 1..1 entre

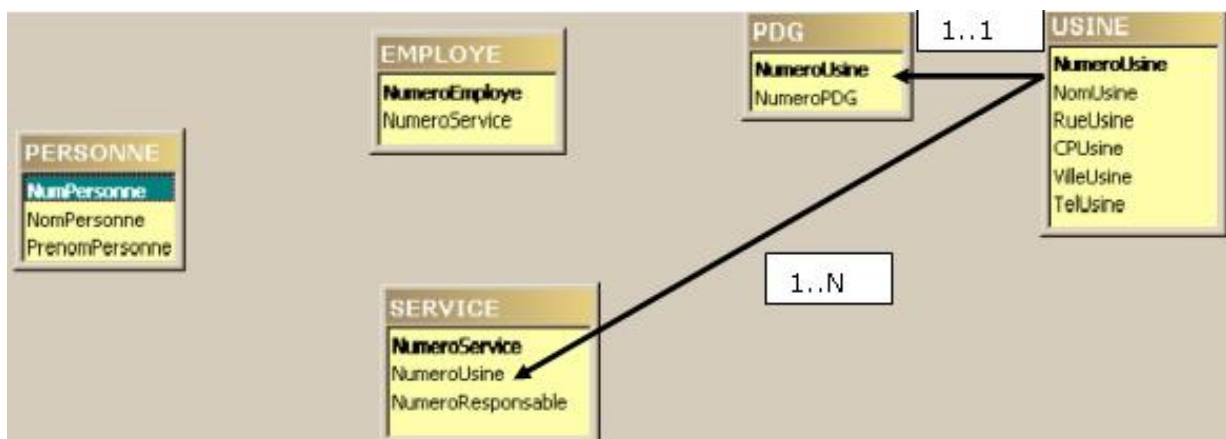


ces deux relations. Pour définir le sens du lien il faut se poser la question : Quelle est la relation la plus importante ? C'est à dire doit-on créer une usine et ensuite son PDG, ou au contraire créer un PDG et ensuite l'usine. La réponse est bien évidemment dans le cas présent on crée l'usine et ensuite son PDG. Le lien ira donc de USINE vers PDG.

USINE [NumUsine] (1..1) → [NumUsine] PDG

Ensuite nous créons un lien de USINE vers SERVICE de type (1..N). L'attribut NumUsine dans la relation SERVICE n'est pas clé candidate, il peut donc prendre plusieurs fois la même valeur. Ce qui nous donne :

USINE [NumUsine] (1..n) → [NumUsine] SERVICE



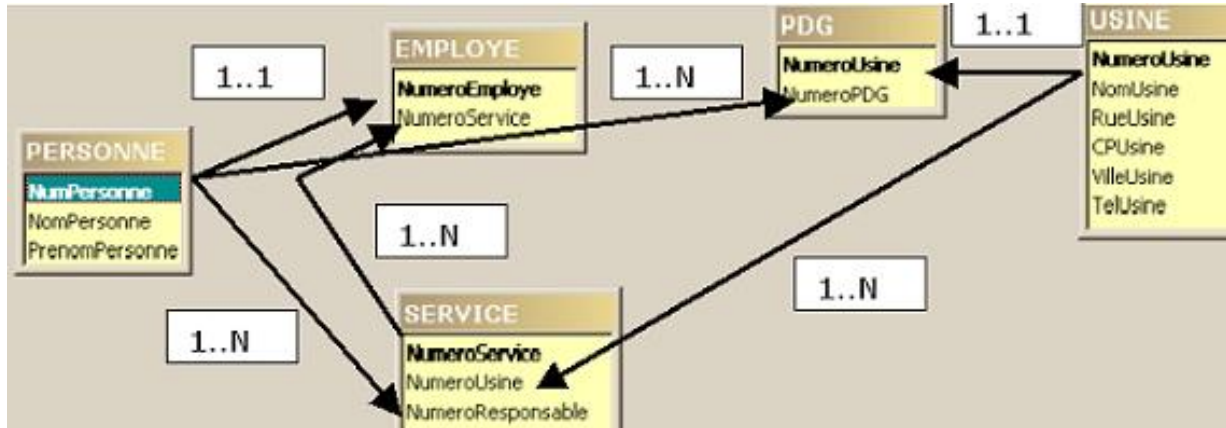
Question 5

En résumé nous avons les liens.

Nous reprenons l'ensemble des liens proposés aux questions précédentes pour obtenir le modèle suivant :

- SERVICE [NumService] (1..n) → [NumService] EMPLOYE
- PERSONNE [NumPersonne] (1..1) → [NumeroEmploye] EMPLOYE

- PERSONNE [NumPersonne] (1..N) -> [NumeroPDG] PDG
- PERSONNE [NumPersonne] (1..N) -> [NumeroResponsable] SERVICE
- USINE [NumUsine] (1..1) -> [NumUsine] PDG
- USINE [NumUsine] (1..n) -> [NumUsine] SERVICE



4.3 Exercice 3 - Partie 2 : Immeubles et occupants

Pour décrire les informations concernant les occupants et les propriétaires d'appartements dans un ensemble d'immeubles, on propose de créer une base de données Immeuble.

- IMMEUBLE (*adresse, nbEtages, dateConstruction, nomGerant)
- PROPRIETE (*adresse, *nomProprietaire, quotePart)
- APPART (*adresse, *numero, typeApt, superficie, etage)
- PERSONNE (*nom, age, codeProfession)
- OCCUPANT (*adresse, *numeroAppart, *nomOccupant, dateArrivee, dateDepart)
- PROFESSION (*codeProfession, libelle)

Indications

Question 1 : Pour construire de façon systématique les liens il faut répondre aux deux questions suivantes :

- Où y-a-t-il des liens ?

Pour cela il faut prendre chaque clé candidate de relation et voir si cette clé peut être une clé étrangère dans une autre relation. Si oui, le lien va de la clé candidate vers la clé étrangère. – Quelle est la nature de ces liens ?

Ici, il faut se poser la question suivante : Est-ce qu'une même valeur de la clé primaire peut apparaître plusieurs fois dans la relation cible? Si oui Le lien est de type 1..n, si non de type 1..1.

Question 2 : Nous nous intéressons toujours ici à la structure des relations de la base, à savoir les clés primaires des relations et les attributs qui composent ces relations. Le contenu des relations peut être un indicateur pour confirmer votre réponse mais en aucun cas une justification. Pour répondre à une telle question, il faut être capable de dire si d'après le modèle proposé, une même valeur de Adresse peut apparaître plusieurs fois dans la relation IMMEUBLE.

Question 3 : Cette question est similaire à la question 2. Pour répondre à une telle question, il faut être capable de dire si d'après le modèle proposé, une même valeur de NumOccupant peut

apparaître plusieurs fois dans la relation OCCUPANT.

Question 4 : Peut-on "associer" un propriétaire et un appartement. Il n'existe aucune relation dans laquelle on peut trouver un propriétaire et un appartement. Regardons alors si en utilisant plusieurs relations on est capable de "lier" ces deux informations. En utilisant les relations PROPRIETE et APPART (par jointure sur l'adresse) on peut avoir des t-uplets dans lesquels on a un propriétaire et un appartement. Cependant le numéro d'appartement peut apparaître avec plusieurs noms de propriétaires. Ce qui signifie que pour un appartement le modèle donne plusieurs propriétaires.

Question 5 : Cette question est similaire aux question 2 et 3. Pour répondre à une telle question, il faut être capable de dire si d'après le modèle proposé, une même valeur de NomProprietaire peut apparaître plusieurs fois dans la relation PROPRIETE.

Question 1

Définir les liens entre les relations présentées dans le modèle.

Pour réaliser cette question il faut commencer par voir comment il est possible de construire de façon systématique les liens entre les relations proposées. Les liens sont les suivants :

TYPEAPPART [TypeAp] (1..n) -> [TypeAp] APPART

PROFESSION [Code] (1..n) -> [CodeProfession] PERSONNE

Ce lien permet de lier des attributs qui ne portent pas le même nom. Ils doivent cependant être de même type.

- IMMEUBLE [Adresse] (1..n) -> [Adresse] PROPRIETE
- IMMEUBLE [Adresse] (1..n) -> [Adresse] APPART
- PERSONNE [Nom] (1..n) -> [NomProprietaire] PROPRIETE
- PERSONNE [Nom] (1..n) -> [NomOccupant] OCCUPANT
- PERSONNE [Nom] (1..n) -> [NomGerant] IMMEUBLE
- APPART [Adresse, Numero] (1..n) -> [Adresse, NumeroAppart] OCCUPANT

Ce dernier lien fait intervenir un groupe d'attributs ce qui est tout à fait possible.

Question 2

Peut-il y avoir plusieurs immeubles à une même adresse ?

NON. L'adresse est la clé primaire de la relation IMMEUBLE, ce qui signifie qu'un immeuble est caractérisé par son adresse, et qu'il ne peut y avoir deux la même valeur d'adresse.

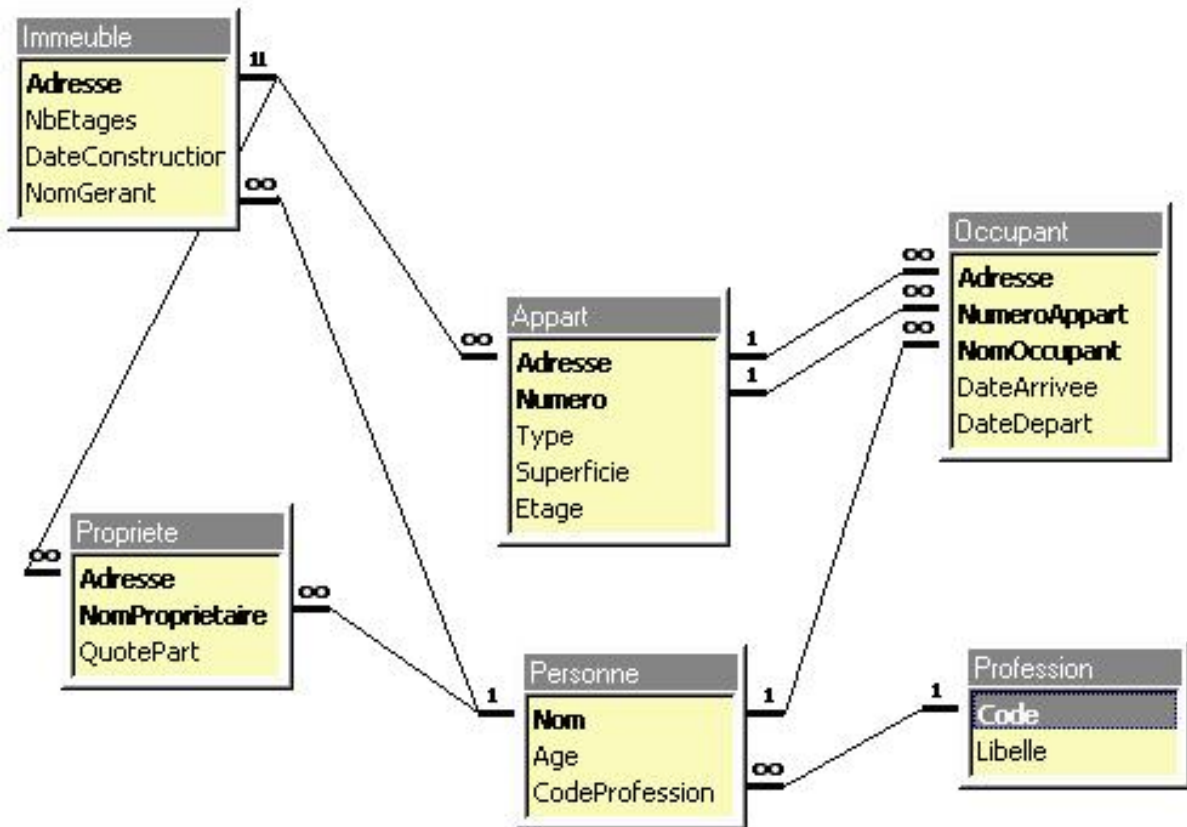
Question 3

Une personne peut-elle habiter plusieurs fois dans le même immeuble ? Dans le même appartement ?

OUI. Car la clé primaire de la relation OCCUPANT contient les attributs (Adresse, NumeroAppart, NomOccupant), on peut donc avoir dans cette relation les deux t-uplets suivants par exemple : (les griffons, 1, Maire,) et (les griffons, 2, maire).

Ce qui signifie qu'un occupant peut avoir simultanément ou dans le temps plusieurs appartements dans un même immeuble.

Par contre il n'est pas possible à une personne d'occuper plusieurs fois le même appartement même à des dates différentes. Pour que cela soit possible il aurait fallu ajouter dans la clé la date d'arrivée (ou de départ).



Question 4

Peut-on connaître le propriétaire d'un appartement ?

A priori NON, car un propriétaire possède une quote-part d'un immeuble et non tel ou tel appartement. Le seul cas où il est possible de connaître le propriétaire est lorsque celui-ci possède 100% de l'immeuble.

Question 5

Une personne peut-elle être propriétaire dans plusieurs immeubles ?

OUI. Une personne peut être propriétaire d'autant d'immeubles qu'elle le souhaite. La clé de la relation PROPRIETE est (Adresse, NomProprietaire). Un nom de propriétaire peut donc apparaître avec plusieurs adresses et donc plusieurs immeubles.

4.4 Exercice 4 - Partie 2 : Les Cinémas

Gestion des cinémas : l'exemple du cours. Le modèle de cet exemple est le suivant :

- CINEMA (*NumeroCinema, NomCinema, AdresseCinema, VilleCinema, CodePostalCinema, TelephoneCinema)
- AFFICHE (*Titre, *NumeroCinema)
- FILM (*Titre, MetteurEnScene, Genre)
- DISTRIBUTION (*Titre, *NomActeur)

Indications

Pour ce premier exercice de travail sur les requêtes je vous encourage à décrire à partir des relations le résultat de chaque requête, étape par étape pour bien appréhender les opérateurs relationnels.

Question 1 : Exercice de présentation de l'opérateur de projection.

Question 2 : Exercice de présentation de l'opérateur de sélection.

Question 3 : Exercice de présentation de l'opérateur de jointure.

Question 4 : Exercice de présentation de l'opérateur de division.

Question 5 : Première vraie requête avec les trois opérateurs relationnels les plus utilisés.

Question 6 : L'objectif de cette requête est la manipulation de l'opérateur d'intersection. Dans la dernière intersection proposée nous utilisons des relations contenant le numéro et le nom des cinémas, car les noms de cinémas ne sont pas uniques.

Question 7 : Ecriture d'une première requête. L'opérateur principal de cette requête est la sélection. Cette requête très simple est un exemple d'utilisation des trois opérateurs les plus utilisés à savoir : la projection, la sélection, et la jointure.

Question 8 : Cette requête très simple est un exemple d'utilisation des trois opérateurs les plus utilisés à savoir : la projection, la sélection, et la jointure.

Question 9 : Cette requête est un exemple d'utilisation des trois opérateurs les plus utilisés à savoir : la projection, la sélection, et la jointure. Deux sélections sont effectuées dans cette requête.

Question 10 : Cette requête est un exemple d'utilisation des trois opérateurs les plus utilisés à savoir : la projection, la sélection, et la jointure.

Question 11 : Exemple particulier d'autojointure. Pour identifier ce type de requête penser à repérer les données dont vous avez besoin sur le modèle (ensemble des relations) et à voir quelles relations vous devez utiliser pour votre requête. Ici la relation FILM sera utilisée deux fois.

Question 12 : Opérateur principal : l'intersection. Cette requête nécessite de réaliser l'intersection des deux tables représentant respectivement les cinémas de Paris ayant une comédie à l'affiche, et les cinémas de Paris ayant à l'affiche une comédie dramatique. Attention il faut tenir compte des remarques que l'on a faites pour la question 6.

Question 13 : Cette requête très simple est un exemple d'utilisation des trois opérateurs les plus utilisés à savoir : la projection, la sélection, et la jointure.

Question 14 : Première requête un peu compliquée. L'opérateur principal de cette requête est la différence. Il faut définir une relation donnant les informations sur les cinémas ayant à l'affiche des films d'action, et faire la différence Films - FilmsAction. Comme chaque fois que l'on utilise un opérateur ensembliste (union, intersection, différence) il faut bien vérifier que les deux relations ont le même ent-tête et d'autre part que l'on travaille sur des données significatives.

Question 15 : L'opérateur principal de cette requête est la division. Il faut définir par division une relation donnant des numéros de cinémas et cette division sera effectuée par une relation donnant les genres des films. La division fera intervenir une relation contenant (numeroCinema, Genre) et une relation contenant (genre).

Décrire ce que donne les requêtes suivantes et dire quel est l'opérateur principal des requêtes :

Question 1

Req1 = [VilleCinema, CodePostalCinema] CINEMA

Cette requête sélectionne des colonnes dans une relation. On obtient la liste des villes et des codes postaux de la relation CINEMA.

Il s'agit d'une projection. Ce type d'opération modifie le degré de la relation CINEMA, mais on

peut voir aussi que la cardinalité de la relation est modifiée. En effet nous ne pouvons pas dans la théorie relationnelle avoir deux fois le même t-uplet dans une relation. Nous obtenons donc un seul t-uplet (Besançon, 25000).

VilleCinema	CodePostalCinema
Besancon	25000
Paris	75015
Paris	75013

Question 2

Req2 = S(Genre = "Comedie") FILM Cette requête sélectionne des lignes (t-uplets) dans la relation FILM. On obtient les informations concernant les comédies.

Il s'agit d'une sélection.

Titre	MetteurEnScene	Genre
Casper	Silberling	Comedie
Gazon Maudit	Balasko	Comedie
Junior	Reitman	Comedie

Question 3

Req3 = FILM [FILM.Titre = AFFICHE.titre] AFFICHE

Cette requête réalise la jointure des relations FILM et AFFICHE selon l'attribut Titre. Ceci permet d'obtenir toutes les informations concernant les films qui sont actuellement à l'affiche dans au moins un cinéma.

FILM.Titre	MetteurEnScene	Genre	NumeroCinema	AFFICHE.Titre
Alerte	Peterson	Aventure	1	Alerte
Alerte	Peterson	Aventure	2	Alerte
Alerte	Peterson	Aventure	3	Alerte
Alerte	Peterson	Aventure	4	Alerte
Casper	Silberling	Comedie	3	Casper
Casper	Silberling	Comedie	4	Casper
Gazon Maudit	Balasko	Comedie	2	Gazon Maudit
Harcelement	Levinson	Comedie Dramatique	4	Harcelement
Junior	Reitman	Comedie	1	Junior
L'Effaceur	Russel	Action	2	L'Effaceur

Question 4

CINE = [NumeroCinema] CINEMA

Req4 = AFFICHE [AFFICHE.NumeroCinema / CINE.NumeroCinema]CINE Cette requête retourne les titres des films qui sont à l'affiche de tous les cinémas. Pour cela on commence par récupérer tous les numéros de cinémas, ensuite on divise la relation AFFICHE par la première relation définie. S'il existe un titre associé à tous les numéros de cinéma, la division le donnera.

Titre
Alerte

Question 5

REQ1 = S(Titre = "Casper")AFFICHE

REQ2 = REQ1 [REQ1.numeroCinema = CINEMA.NumeroCinema] CINEMA

REQ3 = [NomCinema] REQ2

Cette requête retourne le nom des cinémas qui ont Casper à l'affiche.

Il s'agit de notre première vraie requête, qui utilise les trois opérateurs les plus courants, à savoir : la projection, la sélection, et la jointure.

NomCinema
Eldorado
Pax

Question 6

REQ1 = S(Titre = "Casper") AFFICHE

REQ2 = REQ1 [REQ1.NumeroCinema = CINEMA.NumeroCinema] CINEMA

REQ3 = S(Titre = "Alerte") AFFICHE

REQ4 = REQ3 [REQ3.NumeroCinema = CINEMA.NumeroCinema] CINEMA

REQ5 = REQ2 INTER REQ4

REQ6 = [NomCinema] REQ5

Modifier la requête pour obtenir ce que l'on aurait voulu obtenir.

La requête suivante devrait donner le nom des cinémas qui ont à l'affiche le film "Casper" et le film "Alerte". Hélas nous obtenons ici une relation vide car si nous regardons bien le détail de la requête, nous comparons des lignes qui contiennent des titres différents. Ainsi REQ2 et REQ4 sont égales à :

Titre	Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema
Casper	3	Eldorado	Rue St. Lambert	Paris	75015	01 23 56 98 56
Casper	4	Pax	Avenue d'Italie	Paris	75013	01 23 56 98 78

Table 4.1 : REQ2

Titre	Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema
Alerte	1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69
Alerte	2	Plazza	Rue Des Granges	Besancon	25000	03 81 26 35 98
Alerte	3	Eldorado	Rue St. Lambert	Paris	75015	01 23 56 98 56
Alerte	4	Pax	Avenue d'Italie	Paris	75013	01 23 56 98 78

Table 4.2 : REQ4

Il est immédiat qu'aucune ligne commune n'appartient à ces deux relations.

La solution décrite ci-dessous permet quant à elle de trouver le résultat attendu.

```

REQ1 = S(Titre = "Casper") AFFICHE
REQ2 = REQ1 [ REQ1.NumeroCinema = CINEMA.NumeroCinema] CINEMA
REQ3 = S(Titre = "Alerte") AFFICHE
REQ4 = REQ3 [ REQ3.NumeroCinema = CINEMA.NumeroCinema] CINEMA
REQ5 = [NomCinema, NumCinema] REQ2
REQ6 = [NomCinema, NumCinema] REQ4
REQ7 = REQ5 INTER REQ6
REQ7 = [Nomcinema] REQ7

```

Ce qui donne le résultat décrit dans la deuxième partie de l'image.

Numero Cinema	Nom Cinema
3	Eldorado
4	Pax

Table 4.3 : REQ5

Numero Cinema	Nom Cinema
1	Vox
2	Plazza
3	Eldorado
4	Pax

Table 4.4 : REQ6

Ce qui donne :

Remarquons que l'intersection (REQ7) est réalisée sur des relations qui ont pour attributs

Numero Cinema	Nom Cinema
3	Eldorado
4	Pax

Table 4.5 : REQ7

(NomCinema, NumeroCinema). Ceci est dû au fait qu'il n'est pas possible de faire l'intersection sur les noms de cinémas seulement. En effet ceux-ci ne sont pas caractéristiques d'un cinéma. Supposons en effet qu'il existe un cinéma appelé STELLA à Paris qui a à l'affiche Casper mais pas Alerte et un autre cinéma STELLA à Orléans qui a Alerte à l'affiche mais pas Casper. En réalisant directement l'intersection sur les noms on obtiendrait dans notre résultat le cinéma STELLA. Alors qu'aucun cinéma STELLA n'a à l'affiche en même temps Casper et Alerte.

Attention les deux erreurs présentées dans cette question : intersection sur des relations regroupant trop d'informations, et intersection sur des informations non caractéristiques, sont des erreurs classiques. Tout comme le choix d'une union à la place d'une intersection.

En utilisant les opérateurs de l'algèbre relationnelle, définir une requête permettant de définir une relation qui donne ce qui est demandé, et préciser quel est l'opérateur principal de votre requête :

Question 7

Le nom des metteurs en scène des films à l'affiche du PAX

CINEPAX = S(NomCinema = "Pax") CINEMA

Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema
4	Pax	Avenue d'Italie	Paris	75013	01 23 56 98 78

Table 4.6 : CINEPAX

AFFICHEPAX = CINEPAX[CINEPAX.NumeroCinema = AFFICHE.NumeroCinema] AFFICHE

Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema
Alerte 01 23 56 98 78	4	Pax	Avenue d'Italie	Paris	75013
Casper 01 23 56 98 78	4	Pax	Avenue d'Italie	Paris	75013
Harcelement 01 23 56 98 78	4	Pax	Avenue d'Italie	Paris	75013

Table 4.7 : AFFICHEPAX

FILMPAX = AFFICHEPAX [AFFICHEPAX.titre = FILM.titre] FILM

Titre MetteurEnScene	Numero Genre Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CP Cinema	Telephone Cinema
Alerte Peterson	4 Aventure	Pax	Ave d'Italie	Paris	75013	0123569878
Casper Silberling	4 Comedie	Pax	Ave d'Italie	Paris	75013	0123569878
Harcelement Levinson	4 Comedie Dramatique	Pax	Avenue d'Italie	Paris	75013	0123569878

Table 4.8 : FILMPAX

METTEURENSCENEPAX = [MetteurEnScene] FILMPAX

MetteurEnScene
Peterson
Silberling
Levinson

Table 4.9 : METTEURENSCENEPAX

Remarquons que dans l'énoncé le nom du cinéma PAX a été donné en majuscule. Dans la requête nous l'avons écrit avec des majuscules et des minuscules. Il n'y a pas à priori de distinction entre les majuscules et les minuscules dans les bases de données lorsque l'on manipule des informations de type chaînes de caractères. Dans certains cas (code,) il y aura une distinction mais là l'utilisateur devra le préciser.

Question 8

Le nom des acteurs apparaissant dans la distribution des films à l'affiche à Besançon. Cette requête très simple est un exemple d'utilisation des trois opérateurs les plus utilisés à savoir : la projection, la sélection, et la jointure.

CINEBESAC = S(VilleCinema="Besancon")CINEMA

Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69
2	Plazza	Rue Des Granges	Besancon	25000	03 81 26 35 98

Table 4.10 : CINEBESAC

AFFICHEBESAC = CINEBESAC[CINEBESAC.NumeroCinema = AFFICHE.NumeroCinema]AFFICHE

Numero Cinema	Nom Cinema	Adresse Cinema	Ville Cinema	CodePostal Cinema	Telephone Cinema	Titre
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69	Alerte
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69	Junior
2	Plazza	Rue Des Granges	Besancon	25000	03 81 26 35 98	Alerte
2	Plazza	Rue Des Granges	Besancon	25000	03 81 26 35 98	Gazon Maudit
2	Plazza	Rue Des Granges	Besancon	25000	03 81 26 35 98	L'Effaceur

Table 4.11 : AFFICHEBESAC

TITREBESAC = [titre] AFFICHEBESAC

Titre
Alerte
Junior
Gazon Maudit
L'Effaceur

Table 4.12 : TITREBESAC

DISTBESAC = TITREBESAC [TITREBESAC.titre = DISTRIBUTION.titre] DISTRIBUTION

Titre	NomActeur
Alerte	Freeman
Alerte	Hoffman
Alerte	Russo
Gazon Maudit	Abril
Gazon Maudit	Balasko
Gazon Maudit	Chabat
Junior	De Mornay
Junior	De Niro
Junior	Schwarzenegger
L'Effaceur	Schwarzenegger

Table 4.13 : DISTBESAC

ACTEURSBESAC = [NomActeur] DISTBESAC

NomActeur
Freeman
Hoffman
Russo
Abril
Balasko
Chabat
De Mornay
De Niro
Schwarzenegger

Table 4.14 : ACTEURSBESAC

Question 9

Le titre des films d'aventure à l'affiche à Besançon, et le nom du cinéma où ils sont à l'affiche. Nous avons ici une requête classique, c'est à dire basée sur les opérateurs : Sélection, jointure et projection.

```

CINEBESAC = S(VilleCinema="Besancon")CINEMA
AFFICHEBESAC = CINEBESAC[CINEBESAC.NumeroCinema = AFFICHE.NumeroCinema]AFFICHE
FILMBESAC = AFFICHEBESAC [AfficheBesac.Titre = Film.Titre] FILM
FILMAVENTURE = S(Genre="Aventure") FILMBESAC
FILMAVENTURE = [NomCinema, Titre] FILMAVENTURE

```

Titre	NomCinema
Alerte	Vox
Alerte	Plazza

Table 4.15 : FILMAVENTURE

Question 10

Le nom et l'adresse des cinémas projetant le film "Junior".

```

AFFICHEJUNIOR = S(Titre = "Junior") AFFICHE
CINEJUNIOR = AFFICHEJUNIOR [ AFFICHEJUNIOR.NumeroCinema = CINEMA.NumeroCinema] CINEMA
ADRESSEJUNIOR= [NomCinema, AdresseCinema, VilleCinema] CINEJUNIOR

```

Nom Cinema	Adresse Cinema	Ville Cinema
Vox	Grande Rue	Besancon

Table 4.16 : ADRESSEJUNIOR

Question 11

Le titre des films ayant le même metteur en scène que le film "les voleurs".

```

FILMVOLEUR = S(Titre = "Les voleurs") FILM
VOLEURMETTEUR = [MetteurEnScene]FILMVOLEUR
METTEURENSCENEVOLEUR = VOLEURMETTEUR [VOLEURMETTEUR.MetteurEnScene = FILM.MetteurEnScene] FILM
MEMEMETTEUR = [titre] METTEURENSCENEVOLEUR

```

Titre
Les voleurs
Ma saison Preferee

Table 4.17 : MEMEMETTEUR

Question 12

Le nom des cinémas de Paris ayant à l'affiche une comédie **et** une comédie dramatique.

```

CINEPARIS = S(VilleCinema = "Paris") CINEMA
AFFICHEPARIS = CINEPARIS [CINEPARIS.NumeroCinema=AFFICHE.numeroCinema] AFFICHE
FILMPARIS = AFFICHEPARIS[AFFICHEPARIS.Titre = FILM.Titre] FILM
COMEDIES = S(Genre="comedie") FILMPARIS
DRAMES = S(Genre="comedie dramatique") FILMPARIS
CINECOMEDIES = [NomCinema, NumeroCinema] COMEDIES
CINEDRAMES = [NomCinema, NumeroCinema] DRAMES
CINECOMEDIEDRAME = CINECOMEDIE INTER CINEDRAMES
NOMCINE = [NomCinema]CINECOMEDIEDRAME

```

La solution proposée ci-dessus est basée sur l'opérateur d'intersection. Mais il existe d'autres solutions, basées sur la jointure.

NomCinema
Pax

Table 4.18 : NOMCINE

Question 13

Le nom des cinémas de Besançon n'ayant pas de film d'action à l'affiche.

```

CINEBESAC = S(VilleCinema="Besancon") CINEMA
AFFICHEBESAC = CINEBESAC [CINEBESAC.NumeroCinema=AFFICHE.numeroCinema] AFFICHE
FILMBESAC = AFFICHEBESAC [AFFICHEBESAC.Titre = FILM.Titre] FILM
ACTIONBESAC = S(Genre="Action") FILMBESAC
NOMBESAC = [NomCinema, NumeroCinema] CINEBESAC
NOMACTIONBESAC = [NomCinema, NumeroCinema] ACTIONBESAC
CINESANSACTION = NOMBESAC - NOMACTIONBESAC
CINESANSACTION = [NomCinema] CINESANSACTION

```

Ce qui donne comme résultat :

NomCinema
Vox

Table 4.19 : CINESANSACTION

Si vous avez écrit une solution avec une sélection de la forme : Genre <> "Action", la figure 4.2 montre le problème.

Reprenez les informations dans les tables et testez votre solution jusqu'au bout pour comprendre pourquoi elle n'est pas correcte.

Numero	Nom	Adresse	Ville	CodePostal	Telephone
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98

Numero	Nom	Adresse	Ville	CodePostal	Telephone	NumeroCinema	Titre
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69		1 Alerte
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69		1 Junior
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 Alerte
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 Gazon maudit
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 L'effaceur

Numero	Nom	Adresse	Ville	CodePostal	Telephone	NumeroCinema	Titre	Titre
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69		1 Alerte	Alerte
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69		1 Junior	Junior
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 Alerte	Alerte
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 Gazon maudit	Gazon maudit
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 L'effaceur	L'effaceur

Numero	Nom	Adresse	Ville	CodePostal	Telephone	NumeroCinema	Titre	Titre
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69		1 Alerte	Alerte
1	Vox	Grande Rue	Besancon	25000	03 81 82 05 69		1 Junior	Junior
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 Alerte	Alerte
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 Gazon maudit	Gazon maudit
2	Plazza	Rue des Granges	Besancon	25000	03 81 26 35 98		2 L'effaceur	L'effaceur

Numero	Nom
1	Vox
2	Plazza

Figure 4.2 : Différence incorrecte

Question 14

Le numéro des cinémas ayant à l'affiche tous les genres de films.

Pour trouver le numéro des cinémas ayant à l'affiche tous les genres de films, il faut réaliser une division d'une relation contenant les informations concernant les genres des films à l'affiche dans les cinémas par une table contenant les genres de films. Il ne faut surtout pas garder d'informations "parasites" pour la division. Le résultat risquant de ne pas être celui attendu (toujours vide).

GENRES = [Genre] FILM

CINEAFFICHE = AFFICHE [AFFICHE.Titre = FILM.Titre] FILM

GENREAFFICHE = [NumeroCinema, Genre] CINEAFFICHE

TOUSGENRE = GENREAFFICHE [GENREAFFICHE.Genre / GENRES.Genre] GENRE

Question 15

Proposer et écrire des requêtes dont l'opérateur principal est :

1. Un produit cartésien
2. Une union
3. Une intersection
4. Une différence
5. Une division

4.5 Exercice 5 - Partie 2 : La Coopérative Viticole

Un vin est caractérisé par un numéro entier, un cru, une année de production et un degré.

Un producteur est caractérisé par un numéro entier, un nom, un prénom et une région. Un producteur produit un ou plusieurs vins. Réciproquement, un vin est produit par un ou plusieurs producteurs (éventuellement aucun!).

- VINS (*Nvin, Cru, Annee, Degre)
- PRODUCTEURS (*Nprod, Nom, Prenom, Region)
- RECOLTES (*Nprod, *Nvin, Quantite)

Indications

Question 1 : Requête simple avec sélection, projection et jointure.

Question 2 : Opérateur principal : Sélection. Se rappeler que dans les conditions de sélection on peut utiliser d'autres opérateurs de comparaison que l'égalité, et même des opérateurs booléens tels que le ET et le OU.

Question 3 : Requête simple avec sélection, projection et jointure.

Question 4 : Opérateur principal : Sélection. Requête simple avec sélection, projection et jointure.

Question 5 : Opérateur principal : Jointure La création de couples, triplets, à partir d'une relation peut être réalisée en effectuant des jointures d'une table sur elle-même par rapport à des attributs liés à la définition des n-uplets à définir. Ici ce qui lie les différents producteurs, c'est leur région, on définit donc une jointure de PRODUCTEURS sur elle-même par rapport à l'attribut Region. Ensuite pour éviter d'avoir plusieurs fois le même couple ((a1,a2) = (a2,a1)). Nous proposons de définir des couples tels que $a1 < a2$

Question 6 : Division classique de RECOLTES par VINS.

Question 7 : Il s'agit ici de réaliser la division de la relation RECOLTES par une relation contenant tous les numéros de vins produits par le producteur 2.

Question 8 : Utilisation de l'opérateur de différence.

Question 9 : Utilisation de l'opérateur de différence.

Question 1 :

Donner les noms et prénoms des producteurs de la région du Beaujolais.

```
PRODBEAUJOLAIS = S(Region = "Beaujolais") PRODUCTEURS
```

```
PRODBEAUJOLAIS = [Nom, Prenom] PRODBEAUJOLAIS
```

Question 2 :

Quels sont les producteurs de la région du Beaujolais qui ont récolté au moins un vin en quantité supérieur à 300 litres? On veut les noms et les prénoms des producteurs.

```
RECOLTES300 = S(Quantite >= 300) RECOLTES
```

```
PRODBEAUJOLAIS = S(Region = "Beaujolais") PRODUCTEURS
```

```
RECOLTESPROD = RECOLTES300 [RECOLTES300.Nprod = PRODBEAUJOLAIS.Nprod] PRODBEAUJOLAIS
```

```
GROSPRODUCTEURS = [Nom, Prenom] RECOLTESPROD
```

Question 3 :

Donnez les numéros des producteurs qui ont déjà produit du vin de degré égal à 12.

```
VINS12 = S(Degre = 12) VINS
```

```
RECOLTES12 = RECOLTES [RECOLTES.Nvin = VINS12.Nvin] VINS12
```

```
PROD12 = [Nprod] RECOLTES12
```

Question 4 :

Donner la liste des numéros de vin qui ont un degré supérieur à 12 et qui ont été produits par le producteur numéro 24.

1ère solution - avec des sélections :

```
DEGRESUP12 = S(Degre >= 12) VINS
PRODVINS = RECOLTES[Nvin]DEGRESUP12
VINPROD24 = S(Nprod = 24) PRODVINS
VINPROD24 = [Nvin] VVINPROD24
```

2ième solution - avec une intersection :

```
DEGRESUP12 = S(Degre >= 12) VINS
VINPROD24 = S(Nprod = 24) RECOLTES
DEGRESUP12 = [Nvin] DEGRESUP12
VINPROD24 = [Nvin] VVINPROD24
DEGRE12ETPROD24 = DEGRESUP12 INTER VINPROD24
```

Question 5 :

Quels sont les couples de producteurs qui habitent la même région?

- les t-uplets du résultat seront de la forme NProd1, Nom1, NProd2, Nom2, Région, avec NProd1 différent de NProd2
- la présence d'un t-uplet avec NProd1 suivi de NProd2 interdit la présence d'un t-uplet avec NProd2 suivi de NProd1

```
PROD1 = PRODUCTEURS
COUPLEPROD = PRODUCTEURS [PRODUCTEURS.Region = PROD1.Region] PROD1
COUPLEPROD = S(PRODUCTEURS.Nprod < PROD1.Nprod) COUPLEPROD
COUPLE = [PRODUCTEURS.Nprod, PRODUCTEURS.nom, PROD1.Nprod, PROD1.nom] COUPLEPROD
```

Question 6 :

Donnez les noms des producteurs qui ont produit tous les vins.

```
LESVINS = [Nvin] VINS
LESRECOLTES = [Nvin, Nprod] RECOLTES
PRODTOUSVINS = LESRECOLTES [LESRECOLTES.Nvin / LESVINSs.Nvins] LESVINS
PRODTOUSVINS = [Nprod] PRODTOUSVINS
NOMPROD = PRODTOUSVINS [PRODTOUSVINS.Nprod = PRODUCTEURS.Nprod] PRODUCTEURS
NOMPROD = [Nom, Prenom] NOMPROD
```

Question 7 :

Quels sont les producteurs qui produisent au moins tous les vins produits par le producteur 2?

```
VINPROD2 = S(Nprod = 2) RECOLTES
VINPROD2 = [Nvin] VINPROD2
LESRECOLTES = [Nvin, Nprod] RECOLTES
PRODCOMME2 = LESRECOLTES [LESRECOLTES.Nvin / VINPROD2.Nvin] VINPROD2
PRODCOMME2 = [Nprod] PRODCOMME2
```

Question 8 :

Donnez les numéros des producteurs qui n'ont jamais produit du vin de degré égal à 12.

```
VINS12 = S(Degre = 12) VINS
RECOLTES12 = RECOLTES [ RECOLTES.Nvin = VINS12.Nvin] VINS12
PROD12 = [Nprod] RECOLTES12
PROD = [Nprod] PRODUCTEURS
PRODSans12 = PROD - PROD12
```

Question 9 :

Donnez les numéros des producteurs qui n'ont produit que du vin de degré égal à 12.

```
VINS12 = S(Degre = 12) VINS
RECOLTES12 = RECOLTES[RECOLTES.Nvin = VINS12.Nvin] VINS12
PROD12 = [Nprod] RECOLTES12
VINSAUTRE12 = S(Degre <> 12) VINS
RECOLTESAUTRE12 = RECOLTES[RECOLTES.Nvin = VINSAUTRE12.Nvin] VINSAUTRE12
PRODAUTRE12 = [Nprod] RECOLTESAUTRE12
PRODQue12 = PROD12 - PRODAUTRE12
```

Chapter 5

Exercices de la Partie 3 : SQL

5.1 Exercice 1 et 2 - Partie 3 : Les cinémas

5.1.1 les indications

Question 1 : Requête avec les deux clauses obligatoires. Utilisation du terme DISTINCT

NOMCINEMA
Eldorado
Vox
Plazza
Pax

Figure 5.1 : Exo 1 - Question 1

Question 2 : Une requête toute simple à écrire. Première jointure entre deux tables.

NOMCINEMA	ADRESSECINEMA
Vox	Grande Rue
Plazza	Rue des granges
Eldorado	Rue St. Lambert

Figure 5.2 : Exo 1 - Question 2

Question 3 : Pour obtenir le nom des metteurs en scène nous avons besoin de la table **FILM**, et pour obtenir l'information Nom de cinéma nous avons besoin de la table **CINEMA**. Pour réaliser le lien entre ces deux tables, nous introduisons la table **AFFICHE**. D'où deux jointures. Pour choisir les cinémas *Pax*, nous réalisons une sélection à l'aide de la clause **WHERE**.

Question 4 : Comme pour la requête précédente nous avons besoin de trois tables **DISTRIBUTION**, **CINEMA** et **AFFICHE**. Nous utilisons de plus un ordre **DISTINCT** de manière à ne pas avoir plusieurs fois le même nom d'acteur.

Question 5 : Un exemple de jointure exprimé avec l'ordre **JOIN**.

Question 6 : Le but de cette requête est de composer deux conditions de sélection. Les films

METTEURENSCENE
Reitman
Levinson
Balasko
Peterson
Téchiné
Russel
Silberling

Figure 5.3 : Exo 1 - Question 3

NOMACTEUR
Hoffman
Balasko
Russo
Schwarzenegger
De Mornay
De Niro
Freeman
Chabat
Abril

Figure 5.4 : Exo 1 - Question 4

TITRE
Junior
L'Effaceur

Figure 5.5 : Exo 1 - Question 5

doivent être des films d'aventure ET ils doivent être à l'affiche dans un cinéma de Besançon.

Question 7 : Il existe différentes façons de traduire cette requête :

TITRE	NOMCINEMA
Alerte	Vox
Alerte	Plazza

Figure 5.6 : Exo 1 - Question 6

- en réalisant une jointure de la table FILM sur elle même, par rapport à l'attribut pour lequel on veut avoir concordance. Ici l'attribut `MetteurEnScene`.
- en utilisant une sous-requête qui donne le nom du metteur en Scène des voleurs, et dans la

requête principale on sélectionne les films qui ont un metteur en scène égal à celui donné par la sous-requête.

TITRE
Les Voleurs
Ma saison préférée

Figure 5.7 : Exo 1 - Question 7

Question 8 : La solution la plus immédiate serait de réaliser l'intersection de deux requêtes donnant respectivement les noms des cinémas de Paris ayant à l'affiche des comédies et les noms des cinémas de Paris ayant à l'affiche une comédie dramatique.

Une autre solution est d'utiliser une sous-requête donnant les numéros des cinémas de Paris ayant à l'affiche une comédie dramatique, et la requête principale sélectionne les noms des cinémas de Paris ayant à l'affiche une comédie et dont le numero apparaît dans le résultat de la sous requête.

aucune donnée n'a été trouvée

Question 9 : Attention l'erreur classique est de penser qu'il suffit de sélectionner les films qui ne sont pas des films d'actions et de donner le résultat de la requête à partir de cette liste de films. Il faut en fait écrire un sous-requête donnant le numéro des cinémas ayant à l'affiche un film d'action, et dire dans la requête principale que les cinémas qui nous intéressent n'appartiennent pas au résultat de la sous-requête.

NOMCINEMA
Vox

Figure 5.8 : Exo 1 - Question 9

Question 10 : Il faut sélectionner les cinémas de Besançon, et compter combien on a de t-uplets dans le résultat. Première utilisation d'une fonction de groupe.

Question 11 : Pour pouvoir compter pour chaque cinéma le nombre de films à l'affiche, il faut

NBCINEMAS
2

Figure 5.9 : Exo 1 - Question 10

regrouper par cinéma et ensuite compter dans chaque groupe combien on a de t-uplets (chaque t-uplet représente un film).

Question 12 : Pour compter le nombre de films à l'affiche de chaque cinéma, il faut réaliser un regroupement par NumeroCinema. Le regroupement par Nom de cinéma seul n'est pas correct car on peut avoir deux cinémas portant le même nom. Une autre solution consiste à écrire une requête principale qui manipule les noms de cinémas qui s'appuie sur une requête qui elle regroupe les cinémas par Numéro.

Question 13 : Pour savoir quels cinémas ont plus de 2 films à l'affiche, il faut compter pour

NUMEROCINEMA	NBREFILMS
1	2
2	4
3	2
4	1

Figure 5.10 : Exo 1 - Question 11

NOMCINEMA	NBFILMS
Vox	2
Plazza	4
Eldorado	2
Pax	1

Figure 5.11 : Exo 1 - Question 12

chaque cinéma combien on a de films à l'affiche, et ensuite sélectionner ceux qui ont plus de 2 films à l'affiche.

Question 14 : Pour savoir quels cinémas ont plus de 2 films à l'affiche, il faut compter pour

NUMEROCINEMA
1
2
3

Figure 5.12 : Exo 1 - Question 13

chaque cinéma combien on a de films à l'affiche, et ensuite sélectionner ceux qui ont plus de 2 films à l'affiche. Mais attention pour pouvoir donner à la fin le nom des cinémas ceux-ci doivent apparaître dans le regroupement.

Question 15 : Attention l'erreur classique est de penser qu'il suffit de sélectionner, dans la

NOMCINEMA
Vox
Plazza
Eldorado

Figure 5.13 : Exo 1 - Question 14

table DISTRIBUTION, les t-uplets ne correspondant pas à *Schwarzenegger* et de donner le résultat de la requête à partir de cette liste de films. Il faut en fait écrire un sous-requête donnant le titre des films ayant dans leur distribution *Schwarzenegger* et dire que les films qui nous intéressent n'appartiennent pas au résultat de la sous requête.

Question 16 : Il faut compter pour chaque cinéma le nombre de films qu'il a à l'affiche et

TITRE
Alerte
Casper
Gazon Maudit
Harcelement
Les Voleurs
Ma saison préférée

Figure 5.14 : Exo 1 - Question 15

comparer ce nombre au nombre de films à l'affiche du cinéma 2.

Question 17 : Pour traiter cette requête il faut pouvoir garder la référence à un cinéma après

NUMEROCINEMA
1
3

Figure 5.15 : Exo 1 - Question 16

la jointure CINEMA-AFFICHE même s'il n'existe aucun t-uplet dans AFFICHE faisant référence à ce cinéma Nous allons donc utiliser pour la première fois une jointure externe.

Question 18 : Il faut réaliser une division On sélectionne les cinémas pour lesquels il n'existe

NUMEROCINEMA	NBREFILMS
1	2
2	4
3	2
4	1

Figure 5.16 : Exo 1 - Question 17

pas de genre de film qu'ils n'ont pas à l'affiche.

aucune donnée n'a été trouvée

5.1.2 Corrigé

Question 1 :

```
SELECT DISTINCT NomCinema
FROM CINEMA
```

Question 2 : Le nom et l'adresse des cinémas projetant le film *Junior*

```
SELECT NomCinema, AdresseCinema
FROM CINEMA JOIN AFFICHE ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE AFFICHE.Titre = 'Junior'
```

Question 3 : Le nom des metteurs en scène des films à l'affiche du *Pax*

```
SELECT DISTINCT MetteurEnScene
FROM FILM JOIN AFFICHE ON FILM.titre= AFFICHE.Titre
JOIN CINEMA ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE CINEMA.Nomcinema = 'Pax'
```

Question 4 : Le nom des acteurs apparaissant dans la distribution des films à l'affiche à *Besançon*.

```
SELECT DISTINCT NomActeur
FROM DISTRIBUTION JOIN AFFICHE ON DISTRIBUTION.titre= AFFICHE.Titre
JOIN CINEMA ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE CINEMA.VilleCinema = 'Besançon'
```

Question 5 : Le titre des films avec *Schwarzenegger* à l'affiche à *Besançon*.

```
SELECT DISTINCT DISTRIBUTION.titre
FROM DISTRIBUTION INNER JOIN AFFICHE ON DISTRIBUTION.titre = AFFICHE.Titre
INNER JOIN CINEMA ON AFFICHE.NumeroCinema = CINEMA.NumeroCinema
WHERE CINEMA.VilleCinema = 'Besançon'
AND NomActeur ='Schwarzenegger'
```

Question 6 : Le titre des films *d'aventure* à l'affiche à *Besançon*, et le nom du cinéma où ils sont à l'affiche.

```
SELECT FILM.Titre, NomCinema
FROM FILM INNER JOIN AFFICHE ON FILM.Titre = AFFICHE.Titre
INNER JOIN CINEMA ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE FILM.Genre = 'Aventure'
AND CINEMA.VilleCinema = 'Besançon'
```

Question 7 : Le titre des films ayant le même metteur en scène que *Les Voleurs*.

```
SELECT F1.Titre
FROM FILM F1 INNER JOIN FILM F2 ON F1.MetteurEnScene = F2.MetteurEnScene
WHERE F2.Titre='Les Voleurs'
```

Question 8 : Le nom des cinémas de *Paris* ayant à l'affiche une *comédie* et une *comédie dramatique*. Voici deux solutions traitant cette requête :

```
SELECT CINEMA.NomCinema
FROM CINEMA
WHERE NumeroCinema IN(
SELECT CINEMA.NumeroCinema
FROM FILM INNER JOIN AFFICHE ON FILM.Titre = AFFICHE.Titre
INNER JOIN CINEMA ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE CINEMA.VilleCinema='Paris')
```

```

    AND Genre='Comédie Dramatique'
INTERSECT
SELECT CINEMA.NumeroCinema
    FROM FILM INNER JOIN AFFICHE ON FILM.Titre = AFFICHE.Titre
    INNER JOIN CINEMA ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE CINEMA.VilleCinema='Paris'
AND Genre='Comédie' )

```

Avec une sous-requête :

```

SELECT CINEMA.NomCinema
    FROM FILM INNER JOIN AFFICHE ON FILM.Titre = AFFICHE.Titre
    INNER JOIN CINEMA ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE CINEMA.VilleCinema='Paris'
AND Genre='Comédie'
AND CINEMA.NumeroCinema IN
(SELECT CINEMA.NumeroCinema
    FROM FILM INNER JOIN AFFICHE ON FILM.Titre = AFFICHE.Titre
    INNER JOIN CINEMA ON CINEMA.NumeroCinema = AFFICHE.NumeroCinema
WHERE FILM.Genre='Comédie dramatique' )

```

Question 9 : Le nom des cinémas de Besançon n'ayant pas de film *d'action* à l'affiche.

```

SELECT DISTINCT CINEMA.NomCinema
    FROM CINEMA
WHERE VilleCinema='Besançon'
AND NumeroCinema NOT IN
(SELECT AFFICHE.NumeroCinema
    FROM FILM, AFFICHE
WHERE FILM.Titre = AFFICHE.Titre
AND FILM.Genre='Action' )

```

Question 10 : Le nombre de Cinémas à Besançon

```

SELECT COUNT(*) AS NBCinemas
    FROM CINEMA
WHERE VilleCinema='Besançon'

```

On obtient la même chose avec : COUNT(NumeroCinema) et COUNT(Distinct NumeroCinema)

Question 11 : Pour chaque cinéma, donnez le nombre de films à l'affiche.

```

SELECT NumeroCinema, COUNT(*) AS NbreFilms
    FROM AFFICHE
GROUP BY NumeroCinema

```

Question 12 : Le nom des cinémas ainsi que le nombre de films à l'affiche dans ceux-ci

```

SELECT NomCinema, COUNT(Titre) As NBFilms
    FROM CINEMA, AFFICHE
WHERE CINEMA.NumeroCinema = AFFICHE.NumeroCinema
GROUP BY CINEMA.NumeroCinema, NomCinema;

```

Question 13 : Le numéro des cinémas ayant au moins deux films à l'affiche

```

SELECT NumeroCinema
    FROM AFFICHE

```

```
GROUP BY NumeroCinema
HAVING COUNT(*) >= 2
```

Question 14 : Le nom des cinémas ayant au moins deux films à l'affiche

```
SELECT NomCinema
FROM AFFICHE, CINEMA
WHERE AFFICHE.NumeroCinema = CINEMA.NumeroCinema
GROUP BY CINEMA.NumeroCinema, NomCinema
HAVING COUNT(*) >= 2
```

Question 15 : Le titre des films n'ayant pas *Schwarzenegger* dans leur distribution

```
SELECT Titre
FROM FILM
MINUS
SELECT Titre
FROM FILM
WHERE NomActeur = 'Schwarzenegger'
ou avec une sous-requête :
SELECT DISTINCT Titre
FROM FILM
WHERE Titre NOT IN
(SELECT Titre
FROM DISTRIBUTION
WHERE NomActeur = 'Schwarzenegger');
```

Question 16 : Le numéro des cinémas qui ont autant de films à l'affiche que le cinéma Numéro 1.

```
SELECT affiche.NumeroCinema
FROM AFFICHE
GROUP BY NumeroCinema
HAVING COUNT(Titre) =
( SELECT COUNT(Titre)
FROM AFFICHE
WHERE NumeroCinema = 1);
```

Question 17 : Pour chaque cinéma, le nombre de films qu'il a à l'affiche. Si le cinéma n'a aucun film à l'affiche, la valeur 0 devra apparaître en face du numéro de cinéma.

```
SELECT cinema.NumeroCinema, Count(Titre) AS NbreFilms
FROM cinema LEFT JOIN affiche ON cinema.NumeroCinema = affiche.NumeroCinema
GROUP BY cinema.NumeroCinema
```

Question 18 : Numéro des cinémas ayant tous les genres de films à l'affiche. (solution avec EXISTS)

```
SELECT NumeroCinema
FROM CINEMA
WHERE NOT EXISTS
( SELECT Genre
FROM FILM
WHERE NOT EXISTS
(SELECT *
FROM FILM F1, AFFICHE
```

```
WHERE F1.Titre = AFFICHE.Titre  
AND AFFICHE.NumeroCinema = CINEMA.NumeroCinema  
AND F1.Genre = FILM.Genre ))
```

5.2 Exercice 3 - Partie 3 : Le Centre Médical

5.2.1 Les indications

Question 1 : Requête simple impliquant uniquement la table CONSULTATION, et nécessitant une sélection sur le numéro de matricule du médecin.

NUMSECU
1581225128745
2781024147456

Figure 5.17 : Exo 2 - Question 1

Question 2 : Requête simple nécessitant la jointure de 3 tables, puis une sélection sur le nom du médecin.

Question 3 : Encore une requête utilisant plusieurs jointures et une sélection simple.

NOM
Dupont
Faivre

Figure 5.18 : Exo 2 - Question 2

NOM
Dupont
Faivre
Faivre

Figure 5.19 : Question 3

Question 4 : Requête utilisant une sous-requête indépendante pouvant retourner 0 ou plusieurs t-uplets.

NOM
Dupont
Faivre

Figure 5.20 : Exo2 - Question 4

Question 5 : Toujours l'expression d'une sous-requête, en donnant maintenant le nom du patient et non plus son numéro d'iDE sécu.

NOM
Dupont
Faivre

Figure 5.21 : Exo2 - Question 5

Question 6 : Utilisation simple d'une fonction de groupe. Ici on compte le nombre de t-uplets obtenus après avoir réalisé une sélection sur les médicaments déjà prescrits à Mr Faivre.

NOMBREMEDICAMENTS
2

Figure 5.22 : Exo2 - Question 6

Question 7 : Comme précédemment on a une utilisation simple d'une fonction de groupe. On compte le nombre de t-uplets obtenus après avoir réalisé une sélection sur les médicaments déjà prescrits à Mr Faivre qui n'autorise pas les doublons. Au fait ici je ne crois pas que vous puissiez tester votre solution directement.

NOMBREMEDICAMENTS
1

Figure 5.23 : Exo2 - Question 7

Question 8 : On continue à réaliser des comptages. Mais ici il s'agit de tous les patients. Nous allons donc regrouper les t-uplets par patients et ensuite compter pour chaque groupe le nombre de médicaments.

NUMSECU	NOMBREMEDICAMENTS
1581225128745	2
2781024147456	1

Figure 5.24 : Exo2 - Question 8

Question 9 : Il faut exprimer une condition sur un regroupement → Clause HAVING.

Question 10 : Toujours une clause HAVING, mais ici il faut faire attention au choix des attributs de regroupement de manière à être capable de sortir dans le résultat le nom du patient.

Question 11 : Toujours une requête nécessitant un regroupement.

Question 12 : Toujours une requête nécessitant un regroupement.

NUMSECU	NOMBREMEDICAMENTS
1581225128745	2

Figure 5.25 : Exo2 - Question 9

MATRICULE	NOMBRECONSULTATION
1	3
2	1

Figure 5.26 : Exo 2 - Question 11

NOM	NOMBRECONSULTATION
Pouthier	3
Receveur	1

Figure 5.27 : Exo 2 - Question 12

Question 13 : Il s'agit toujours d'une requête avec un regroupement. Nous avons en plus une sélection sur le nom du patient. Vous n'avez pas ici à utiliser une clause **HAVING**.

NOM	NOMBRECONSULTATION
Pouthier	1

Figure 5.28 : Exo 2 - Question 13

Question 14 : Il s'agit toujours de réaliser des regroupements, mais avec des attributs provenant de différentes relations.

MÉDECIN	NOMBRECONSULTATION	PATIENT
Receveur	1	Faivre
Pouthier	2	Faivre
Pouthier	1	Dupont

Figure 5.29 : Exo 2 - Question 14

Question 15 : Il faut toujours écrire une requête avec un regroupement et une clause **HAVING**, mais en plus il faut utiliser une sous-requête dans l'expression de la clause **HAVING**. Il faut que le nombre de consultations du médecin qui nous intéresse soit supérieur ou égal au nombre de consultations de tous les médecins.

Question 16 : Il faut toujours écrire une requête avec un regroupement et une clause **HAVING**,

NOM	NOMBRECONSULTATION
Pouthier	3

Figure 5.30 : Exo 2 - Question 15

mais en plus il faut utiliser une sous-requête dans l'expression de la clause **HAVING**. Il faut que le nombre de consultations du patient qui nous interesse soit supérieur ou égal au nombre de consultations de tous les autres patients.

MEILLEURPATIENT
Faivre

Figure 5.31 : Exo2 - Question 16

Question 17 : Nous regroupons par médecin et patient. Ensuite nous comptons pour chaque patient le nombre de consultations et celui-ci doit être supérieur ou égal à tous les nombres de consultations des patients du médecin considéré.

MEDECIN	NOMBRECONSULTATION	PATIENT
Receveur	1	Faivre
Pouthier	2	Faivre

Figure 5.32 : Exo 2 - Question 17

5.2.2 Corrigés

Question 1 : Donnez le numéro de sécurité sociale des patients du médecin de matricule 1.

```
SELECT DISTINCT NumSS
  FROM CONSULTATION
 WHERE Matricule = 1
```

Question 2 : Donnez le nom des patients du médecin *Pouthier*.

```
SELECT DISTINCT PATIENT.Nom
  FROM PATIENT INNER JOIN CONSULTATION ON
    PATIENT.NumSS = CONSULTATION.NumSS INNER JOIN MEDECIN ON
    MEDECIN.Matricule = CONSULTATION.Matricule
 WHERE MEDECIN.Nom='Pouthier'
```

Question 3 : Donnez le nom des patients à qui l'on a déjà prescrit du *Doliprane*.

```
SELECT Nom
  FROM PATIENT INNER JOIN CONSULTATION ON
    PATIENT.NumSS = CONSULTATION.NumSS INNER JOIN
    PRESCRIPTION ON CONSULTATION.NumCons = PRESCRIPTION.NumCons
    INNER JOIN MEDICAMENT on PRESCRIPTION.Code = MEDICAMENT.Code
 WHERE Libelle = 'Doliprane'
```

Question 4 : Donnez le nom des patients qui ont déjà consulté un médecin qui a déjà eu comme patient, le patient de numéro sécu *1581225128745*.

```
SELECT DISTINCT PATIENT.Nom
  FROM PATIENT INNER JOIN CONSULTATION ON
    PATIENT.NumSS = CONSULTATION.NumSS
 WHERE CONSULTATION.Matricule IN
 (SELECT Matricule from CONSULTATION
  WHERE NumSS = '1581225128745')
```

Question 5 : Donnez le nom des patients qui ont déjà consulté un médecin qui a déjà eu comme patient, le patient *Dupont*.

```
SELECT DISTINCT Nom
  FROM PATIENT INNER JOIN CONSULTATION ON
    PATIENT.NumSS = CONSULTATION.NumSS
 WHERE CONSULTATION.Matricule IN
 (SELECT Matricule
  FROM CONSULTATION INNER JOIN PATIENT ON
    PATIENT.NumSS = CONSULTATION.NumSS
  WHERE Nom = 'Dupont' )
```

Question 6 : Combien de médicaments ont déjà été prescrits à Monsieur *Faivre* ? Si un médicament a été prescrit plusieurs fois on le compte plusieurs fois.

```
SELECT COUNT(PRESCRIPTION.code) AS NombreMédicaments
  FROM PATIENT INNER JOIN CONSULTATION ON
    PATIENT.NumSS = CONSULTATION.NumSS INNER JOIN
    PRESCRIPTION ON CONSULTATION.NumCons = PRESCRIPTION.NumCons
 WHERE PATIENT.Nom = 'Faivre'
```

Question 7 : Combien de médicaments ont déjà été prescrits à Monsieur *Faivre* ? On veut connaître le nombre de médicaments différents.

```
SELECT COUNT( DISTINCT PRESCRIPTION.code) AS NombreMédicaments
  FROM PATIENT INNER JOIN CONSULTATION ON
    PATIENT.NumSS = CONSULTATION.NumSS INNER JOIN
```

```

    PRESCRIPTION ON CONSULTATION.NumCons = PRESCRIPTION.NumCons
    WHERE PATIENT.Nom = 'Faivre'

```

Question 8 : Pour chaque patient, donnez le nombre de médicaments différents qui leur ont déjà été prescrits.

```

SELECT PATIENT.NumSS, COUNT(DISTINCT PRESCRIPTION.Code) AS NombreMédicaments
FROM PATIENT INNER JOIN CONSULTATION
    ON PATIENT.NumSS = CONSULTATION.NumSS INNER JOIN
    PRESCRIPTION ON CONSULTATION.NumCons = PRESCRIPTION.NumCons
GROUP BY PATIENT.NumSS

```

Question 9 : Donnez le numéro de sécurité sociale des patients ayant eu au moins deux médicaments prescrits, ainsi que le nombre de médicaments qui leur ont été prescrits.

```

SELECT consultation.NumSS, COUNT(DISTINCT PRESCRIPTION.Code)
    AS NombreMédicaments
FROM CONSULTATION INNER JOIN PRESCRIPTION ON
    CONSULTATION.NumCons = PRESCRIPTION.NumCons
GROUP BY consultation.NumSS
HAVING COUNT(DISTINCT PRESCRIPTION.Code)>=2

```

Question 10 : Donnez le nom des patients ayant eu au moins deux médicaments prescrits.

```

SELECT PATIENT.Nom
FROM CONSULTATION INNER JOIN PRESCRIPTION ON
    CONSULTATION.NumCons = PRESCRIPTION.NumCons INNER JOIN
    PATIENT ON PATIENT.NumSS = CONSULTATION.NumSS
GROUP BY consultation.NumSS, PATIENT.Nom
HAVING COUNT(DISTINCT PRESCRIPTION.Code)>=2

```

Question 11 : Donnez pour chaque médecin son numéro de matricule et le nombre de consultations qu'il a réalisé.

```

SELECT Matricule, COUNT(CONSULTATION.NumCons) AS NombreConsultation
FROM CONSULTATION
GROUP BY Matricule

```

Question 12 : Donnez pour chaque médecin son nom et le nombre de consultations qu'il a réalisé.

```

SELECT MEDECIN.Nom, COUNT(CONSULTATION.NumCons) AS NombreConsultation
FROM MEDECIN INNER JOIN CONSULTATION ON
    MEDECIN.Matricule = CONSULTATION.Matricule
GROUP BY MEDECIN.Nom, MEDECIN.Matricule

```

Question 13 : Donnez pour chaque médecin son nom et le nombre de consultations qu'il a réalisé avec comme patient Mr DUPONT.

```

SELECT MEDECIN.Nom, Count(CONSULTATION.NumCons) AS NombreConsultation
FROM MEDECIN INNER JOIN CONSULTATION ON
    MEDECIN.Matricule = CONSULTATION.MAtricule INNER JOIN
    PATIENT ON PATIENT.NumSS = CONSULTATION.NumSS
WHERE PATIENT.Nom = 'Dupont'
GROUP BY MEDECIN.Nom, MEDECIN.Matricule

```

Question 14 : Donnez pour chaque médecin son nom et le nombre de consultations qu'il a réalisé avec chaque patient. On donnera dans le résultat le nom du médecin, le nom du patient, et le nombre de consultations.

```

SELECT MEDECIN.Nom AS Médecin , COUNT(CONSULTATION.NumCons)
    AS NombreConsultation, PATIENT.Nom AS Patient

```

```

FROM MEDECIN INNER JOIN CONSULTATION ON
    MEDECIN.Matricule = CONSULTATION.Matricule INNER JOIN
    PATIENT ON PATIENT.NumSS = CONSULTATION.NumSS
GROUP BY MEDECIN.Nom, PATIENT.Nom, MEDECIN.Matricule, PATIENT.NumSS

```

Question 15 : Donnez le nom du médecin qui a réalisé le plus de consultations.

```

SELECT MEDECIN.Nom, Count(CONSULTATION.NumCons) AS NombreConsultation
FROM MEDECIN INNER JOIN CONSULTATION ON
    MEDECIN.Matricule = CONSULTATION.Matricule
GROUP BY MEDECIN.Nom, MEDECIN.Matricule
HAVING COUNT(CONSULTATION.NumCons) >= ALL
(SELECT COUNT(CONSULTATION.NumCons) AS NombreConsultation
FROM MEDECIN INNER JOIN CONSULTATION ON
    MEDECIN.Matricule = CONSULTATION.Matricule
GROUP BY MEDECIN.Nom, MEDECIN.Matricule)

```

Question 16 : Donnez le nom du meilleur patient du médecin de matricule 1.

```

SELECT PATIENT.Nom AS MeilleurPatient
FROM PATIENT INNER JOIN CONSULTATION
    ON PATIENT.NumSS = CONSULTATION.NumSS
WHERE CONSULTATION.Matricule =1
GROUP BY PATIENT.Nom, PATIENT.NumSS
HAVING Count(CONSULTATION.NumCons) >=ALL
(SELECT Count(CONSULTATION.NumCons) AS NombreConsultation
FROM CONSULTATION
WHERE CONSULTATION.Matricule = 1
GROUP BY CONSULTATION.NumSS )

```

Question 17 : Donnez pour chaque médecin son nom et le nom de son meilleur patient (c'est à dire celui ou ceux qui ont réalisés le plus de consultations).

```

SELECT M1.Nom AS Medecin, Count(CONSULTATION.NumCons)
AS NombreConsultation, PATIENT.Nom AS Patient
FROM MEDECIN M1 INNER JOIN CONSULTATION ON
    M1.Matricule = CONSULTATION.Matricule INNER JOIN
    PATIENT ON PATIENT.NumSS = CONSULTATION.NumSS
GROUP BY M1.Nom, PATIENT.Nom, M1.Matricule, PATIENT.NumSS
HAVING Count(CONSULTATION.NumCons) >= ALL
(SELECT Count(C1.NumCons)
FROM MEDECIN M2 INNER JOIN CONSULTATION C1 ON
    M2.Matricule = C1.Matricule INNER JOIN
    PATIENT P1 ON P1.NumSS = C1.NumSS
WHERE M2.Matricule = M1.Matricule
GROUP BY M2.Matricule, P1.NumSS);

```

5.3 Exercice 4 - Partie 3 : La Coopérative Viticole

5.3.1 Les indications

Question 1 : Il s'agit d'une requête portant sur une seule table, la table `PRODUCTEUR`, avec une sélection, et l'utilisation de la clause `ORDER` pour trier les résultats.

Question 2 : Pour obtenir le résultat demandé, il faut récupérer des informations dans les tables

NOM	PRENOM
Dupont	Jean
Dupuis	Laurent
Renard	Alain
Renard	Pierre

Figure 5.33 : Exo 3 - Question 1

`PRODUCTEUR` et `RECOLTE`. Nous réalisons donc tout d'abord une jointure entre ces deux tables et ensuite une sélection permet de choisir les producteurs du *Beaujolais* et les récoltes supérieures à 300 litres.

Question 3 : Il s'agit ici d'écrire une sélection utilisant des opérateurs logiques `AND` et `OR`.

NOM	PRENOM
Dupuis	Laurent
Renard	Alain
Renard	Alain
Renard	Alain
Renard	Pierre
Renard	Pierre

Figure 5.34 : Exo 3 - Question 2

Question 4 : Pour décrire cette requête il faut utiliser une fonction d'agrégation.

NVIN
1
2
12

Figure 5.35 : Exo 3 - Question 3

Question 5 : Pour calculer pour chaque cru, la quantité totale de vin produite, il faut réaliser un regroupement par cru, et ensuite sommer dans chaque regroupement, les quantités de vin produites.

Question 6 : Il faut compter pour chaque producteur le nombre de crus qu'il produit. Pour cela il faut réaliser un regroupement. Ensuite une `SELECTION SUR LES REGROUPEMENTS` permet de ne garder que ceux pour lesquels le nombre de crus produits est supérieur ou égal à 3.

QUANTITETOTALE
900

Figure 5.36 : Exo 3 - Question 4

CRU	QUANTITEPARCRU
Médoc	350
St Armand	900
St Julien	1800

Figure 5.37 : Exo 3 - Question 5

Question 7 : Il faut réaliser une jointure de la table `PRODUCTEUR` sur elle-même. La difficulté

NOM	PRENOM
Renard	Alain

Figure 5.38 : Exo 3 - Question 6

est alors de bien choisir le critère de jointure. Dans un même t-uplet on aura les informations concernant deux producteurs, quel est le lien entre ces producteurs ? Autre problème : Comment faire pour ne pas obtenir deux fois la même information ? Une solution est de dire que le numéro du producteur issu de la première table, doit être strictement inférieur au numéro du producteur issu de la deuxième table (`PRODUCTEUR`).

NOM	PRENOM	NOM	PRENOM	REGION
Renard	Alain	Renard	Pierre	Beaujolais
Renard	Pierre	Dupuis	Laurent	Beaujolais
Renard	Alain	Dupuis	Laurent	Beaujolais
Dupuis	Laurent	Dupont	Jean	Beaujolais
Renard	Pierre	Dupont	Jean	Beaujolais
Renard	Alain	Dupont	Jean	Beaujolais

Figure 5.39 : Exo 3 - Question 7

Question 8 : Il s'agit ici de décrire une division en SQL. La solution demandée est celle utilisant la fonction `COUNT`. Il faut que le nombre de vins produits par un producteur soit égal au nombre de vins existants.

Question 9 : Il s'agit ici de décrire une division en SQL. La solution demandée est celle utilisant `EXISTS` et `NOT`. Il faut sélectionner dans la table `PRODUCTEUR`, les producteurs pour lesquels il n'existe pas de vin dans la table `VINS` qu'ils ne produisent pas.

Question 10 : Il s'agit toujours de faire une division, la difficulté étant ici de bien penser à

NPROD
1

Figure 5.40 : Exo 3 - Question 8

NOM	PRENOM
Renard	Alain

Figure 5.41 : Exo 3 - Question 9

compter le nombre de vins produits par un producteur donné et qui sont aussi produits par le producteur N°2.

NPROD
1
2

Figure 5.42 : Exo 3 - Question 10

Question 11 : IL s'agit d'une division un peu plus complexe. On ne divise pas la table RECOLTE par la table VIN, mais par la liste des vins produits par le producteur N°2.

5.3.2 Corrigés

Question 1 : Donner les noms et prénoms des producteurs de la région du *Beaujolais* triés par nom et prénom.

```
SELECT Nom, Prenom
      FROM PRODUCTEUR
      WHERE Region = 'Beaujolais'
      ORDER BY Nom, Prenom
```

Question 2 : Quels sont les producteurs de la région du *Beaujolais* qui ont récolté au moins un vin en quantité supérieur à 300 litres ? On veut les noms et les prénoms des producteurs, avec un tri par ordre alphabétique.

```
SELECT Nom, Prenom
      FROM PRODUCTEUR INNER JOIN RECOLTE ON PRODUCTEUR.NProd = RECOLTE.Nprod
      WHERE Region='Beaujolais'
      AND Quantite > =300
      ORDER BY Nom, Prenom;
```

Question 3 : Donner la liste des numéros de vin qui ont un degré supérieur à 12 ou qui ont été produits par le producteur numéro 24.

```
SELECT DISTINCT RECOLTE.NVin
      FROM VIN INNER JOIN RECOLTE ON VIN.NVin = RECOLTE.NVin
      WHERE NProd = 24 OR Degre >= 12
```

Question 4 : Quelle est la quantité totale de vin numéro 12 produite?

```
SELECT SUM(Quantite) AS QuantiteTotale
      FROM RECOLTE
      WHERE NVin=12
```

Question 5 : Donner la liste des crus triée par ordre alphabétique. Pour chaque cru, on veut son nom et la quantité de vin produite par ce cru.

```
SELECT Cru, SUM(Quantite) AS QuantiteParCru
      FROM VIN INNER JOIN RECOLTE ON VIN.NVin = RECOLTE.NVin
      GROUP Cru
      ORDER BY Cru
```

La variante :

```
SELECT Cru, SUM(Quantite) AS QuantiteParCru
      FROM VIN INNER JOIN RECOLTE ON VIN.NVin = RECOLTE.NVin
      GROUP BY VIN.Nvin,Cru
      ORDER BY Cru
```

Question 6 : Quels sont les producteurs qui produisent des vins issus de trois crus différents, au moins ? Pour chaque producteur sélectionné, on veut le nom et le prénom.

```
SELECT Nom, Prenom
      FROM VIN INNER JOIN RECOLTE ON VIN.NVin = RECOLTE.NVin
      INNER JOIN PRODUCTEUR ON PRODUCTEUR.NProd = RECOLTE.NProd
      GROUP BY PRODUCTEUR.NProd, Nom, Prenom
      HAVING COUNT(DISTINCT Cru) >= 3
```

Question 7 : Quels sont les couples de producteurs qui habitent la même région?

- les t-uplets du résultat seront de la forme - Nom1, Prenom1, Nom2, Prenom2, Région -, avec NProd1 différent de NProd2
- la présence d'un t-uplet avec NProd1 suivi de NProd2 interdit la présence d'un t-uplet avec NProd2 suivi de NProd1


```
SELECT P1.Nom, P1.Prenom, P2.Nom, P2.Prenom, P1.Region
FROM PRODUCTEUR P1 INNER JOIN PRODUCTEUR P2 ON P1.Region = P2.Region
WHERE P1.NProd < P2.Nprod
```

Question 8 : Quels sont les producteurs qui ont produit tous les vins ? Ecrire la requête en utilisant la fonction COUNT.

```
SELECT NProd
FROM RECOLTE
GROUP BY NProd
HAVING COUNT(NVin) = (SELECT COUNT(*) FROM VIN)
```

Question 9 : Quels sont les producteurs qui ont produit tous les vins (Nom et Prenom)? Ecrire la requête en utilisant le quantificateur EXISTS et l'opérateur logique NOT.

```
SELECT Nom, Prenom
FROM PRODUCTEUR
WHERE NOT EXISTS
  (SELECT NVin
   FROM VIN
   WHERE NOT EXISTS
    ( SELECT RECOLTE.NProd, RECOLTE.Nvin
      FROM RECOLTE
      WHERE RECOLTE.NProd=PRODUCTEUR.NProd
        AND RECOLTE.NVin = VIN.NVin))
```

Question 10 : Quels sont les producteurs qui produisent au moins tous les vins produits par le producteur 2? Ecrire la requête en utilisant la fonction COUNT.

```
SELECT R1.NProd
FROM RECOLTE R1, RECOLTE R2
WHERE R1.NVin = R2.NVin
AND R2.NProd = 2
GROUP BY R1.NProd
HAVING COUNT(R1.NVin )=
  ( SELECT COUNT(*) FROM RECOLTE WHERE NProd=2 )
```

Question 11 : Quels sont les producteurs qui produisent au moins tous les vins produits par le producteur 2? Ecrire la requête en utilisant le quantificateur EXISTS et l'opérateur logique NOT.

```
SELECT NProd
FROM PRODUCTEUR
WHERE NOT EXISTS
  (SELECT R1.NVin
   FROM RECOLTE R1
   WHERE R1.NProd = 2
   AND NOT EXISTS (SELECT R2.NVin
                    FROM RECOLTE R2
                    WHERE R2.NVin = R1.NVin
                    AND R2.Nprod = PRODUCTEUR.Nprod))
```

5.4 Exercice 5 - Partie 3 : Bars et buveurs de bières

5.4.1 Les indications

Question 1 : Requête simple avec des jointures et une sélection.

NOMBAR
Lulu
les loups

Figure 5.43 : Exo 4 - Question 1

Question 2 : Le résultat attendu est obtenu en réalisant simplement une jointure interne des tables `BUVEUR`, `FREQUENTE`, `AIME` et `SERT`. Si pour un buveur donné il reste au moins un t-uplet dans cette jointure celle-ci correspondra à une bière que le buveur aime et qui est servie dans un bar qu'il fréquente.

NOMBUVEUR
Dupont
Durand

Figure 5.44 : Exo 4 - Question 2

Question 3 : Pour trouver les buveurs fréquentant tous les bars il suffit de réaliser une division de la table `FREQUENTE`, par une table contenant tous les numéros de Bars (`Cbar`). La division proposée ici est réalisée en utilisant `NOT EXISTS`.

NOMBUVEUR
Maire
Dupont

Figure 5.45 : Exo 4 - Question 3

Question 4 : Un tel buveur ne peut pas ne pas fréquenter un bar servant au moins une bière qu'il aime.

Question 5 : Ces buveurs ne fréquentent aucun bar qui ne sert pas de bière qu'ils aiment.

Question 6 : Ces buveurs ne sont pas dans la liste des buveurs de bière fréquentant au moins un bar proposant une bière qu'ils aiment.

CBUVEUR
1
3

Figure 5.46 : Exo 4 - Question 4

NOMBUVEUR
Maire
Dupont

Figure 5.47 : Exo 4 - Question 5

Question 7 : Requête de regroupement.

Question 8 : Requête de sélection sur un regroupement.

Question 9 : On a besoin d'utiliser deux fois la table **FREQUENTE**. La jointure entre ces deux tables se fera sur le code du bar.

CBUVEUR
2

Figure 5.48 : Exo 4 - Question 6

CBUVEUR	COUNT(DISTINCTCBAR)
1	2
2	1
3	2

Figure 5.49 : Exo 4 - Question 7

CBUVEUR
1
3

Figure 5.50 : Exo 4 - Question 8

5.4.2 Corrigés

Question 1 : Les noms des bars qui servent une bière appréciée par *Dupont*.

```
SELECT DISTINCT NomBar
  FROM AIME INNER JOIN BUVEUR ON BUVEUR.Cbuveur = AIME.CBuveur
        INNER JOIN SERT ON SERT.CBiere = AIME.CBiere
        inner join BAR on SERT.CBar = BAR.CBar
 WHERE BUVEUR.NomBuveur='Dupont'
```

Question 2 : Le noms des buveurs qui fréquentent au moins un bar où l'on sert une bière qu'ils aiment.

```
SELECT DISTINCT NomBuveur
  FROM AIME INNER JOIN BUVEUR ON BUVEUR.Cbuveur = AIME.CBuveur
        INNER JOIN SERT ON SERT.CBiere = AIME.CBiere
        INNER JOIN FREQUENTE ON BUVEUR.Cbuveur = FREQUENTE.Cbuveur
 WHERE FREQUENTE.CBar = SERT.CBar
```

Question 3 : Les noms des buveurs qui fréquentent tous les bars.

```
SELECT BUVEUR.NomBuveur
  FROM BUVEUR
 WHERE NOT EXISTS
   ( SELECT CBar
     FROM BAR
     WHERE NOT EXISTS
      ( SELECT CBar, CBuveur
        FROM FREQUENTE
        WHERE BAR.CBar = FREQUENTE.CBar
          AND BUVEUR.CBuveur = FREQUENTE.CBuveur))
```

Question 4 : Les numéros des buveurs qui fréquentent tous les bars qui servent au moins une bière qu'ils aiment.

CBUVEUR
1
2
3

Figure 5.51 : Exo 4 - Question 9

```
SELECT DISTINCT CBUveur
FROM BUVEUR
WHERE CBUveur NOT IN
(SELECT Cbuveur
FROM SERT, AIME
WHERE SERT.CBiere=AIME.CBiere
AND Cbuveur NOT IN
(SELECT Cbuveur
FROM FREQUENTE
WHERE CBar = SERT.Cbar))
```

Question 5 : Les noms des buveurs qui ne fréquentent que les bars qui servent une bière qu'ils aiment.

```
SELECT BUVEUR.NomBuveur
FROM BUVEUR
WHERE BUVEUR.Cbuveur Not In
(SELECT CBUveur
FROM FREQUENTE
WHERE CBUveur NOT IN
( SELECT CBUveur
FROM SERT, AIME
WHERE SERT.CBiere = AIME.CBiere
AND SERT.CBar = FREQUENTE.CBar));
```

Question 6 : Les buveurs qui ne fréquentent aucun bar où l'on sert une bière qu'ils aiment.

```
SELECT CBUveur
FROM BUVEUR
WHERE CBUveur NOT IN
( SELECT FREQUENTE.CBUveur
FROM SERT, AIME, FREQUENTE
WHERE SERT.CBiere = AIME.CBiere
AND SERT.CBar = FREQUENTE.CBar
AND FREQUENTE.CBUveur = AIME.CBUveur)
```

Question 7 : Donner pour chaque buveur, leur numéro et le nombre de bars servant une bière qu'ils aiment.

```
SELECT CBUveur, COUNT( DISTINCT CBar)
FROM SERT, AIME
WHERE SERT.Cbiere = AIME.CBiere
GROUP BY CBUveur
```

Question 8 : Les numéros des buveurs qui fréquentent au moins deux bars où l'on sert une bière qu'ils aiment.

```
SELECT AIME.CBUveur
```

```
FROM SERT INNER JOIN AIME ON SERT.CBiere = AIME.CBiere
  INNER JOIN FREQUENTE ON SERT.CBar = FREQUENTE.CBar
GROUP BY AIME.CBuveur
HAVING COUNT(DISTINCT SERT.CBar) >= 2
```

Question 9 : Les numéros des buveurs qui vont dans au moins un bar fréquenté par *Dupont*.

```
SELECT DISTINCT F1.CBuveur
FROM FREQUENTE AS F1, FREQUENTE AS F2, BUVEURS
WHERE F1.Cbar = F2.CBar
AND F2.CBuveur = BUVEURS.CBuveur
AND BUVEURS.NomBuveur= 'Dupont'
```

Sans Dupont on a la requête :

```
SELECT DISTINCT F1.CBuveur
FROM FREQUENTE F1 INNER JOIN FREQUENTE F2 ON F1.Cbar = F2.CBar
  INNER JOIN BUVEUR ON F2.CBuveur = BUVEUR.CBuveur
WHERE BUVEUR.NomBuveur= 'Dupont'
AND F1.CBuveur <> BUVEUR.CBuveur
```

Chapter 6

Exercices de la partie 4 : Normalisation

6.1 Exercice 1 - Partie 4 : Scolarité

les indications

Question 1 : Pour savoir si l'on a à faire à une dépendance fonctionnelle, il faut regarder si l'on a une relation plusieurs vers 1. Est-ce qu'une valeur d'un groupe d'attributs définit de façon unique la valeur d'un autre groupe d'attributs ?

Question 2 : Pour définir la clé d'une relation on peut :

1. commencer par représenter les graphes de toutes les dépendances
2. et repérer quels sont les attributs à partir desquels on retrouve les autres attributs.
3. Ensuite on utilise l'algorithme qui permet de vérifier que le groupe d'attributs est une super-clé de la relation.
4. Enfin on vérifie que la super-clé est irréductible.

le corrigé

Question 1

- R1 : NumUE \rightarrow NbreHeureEnseignementUE
- R2 : NumUE \rightarrow NumSSEnseignant
- R3 : NumSSEnseignant \rightarrow NomEnseignant, PrenomEnseignant, Grade
- R4 : NumEtudiant \rightarrow NomEtudiant, PrenomEtudiant
- R5 : Pas de dépendance fonctionnelle
- R6 : NumSSEnseignant \rightarrow UFRRattachement
- R7 : NumEtudiant, NumUE \rightarrow ResultatUE
- R8 : Grade \rightarrow Indice
- R9 : UFRRattachement \rightarrow NbreEnseignantUFR

Question 2

La représentation graphique des DFs est la suivante :

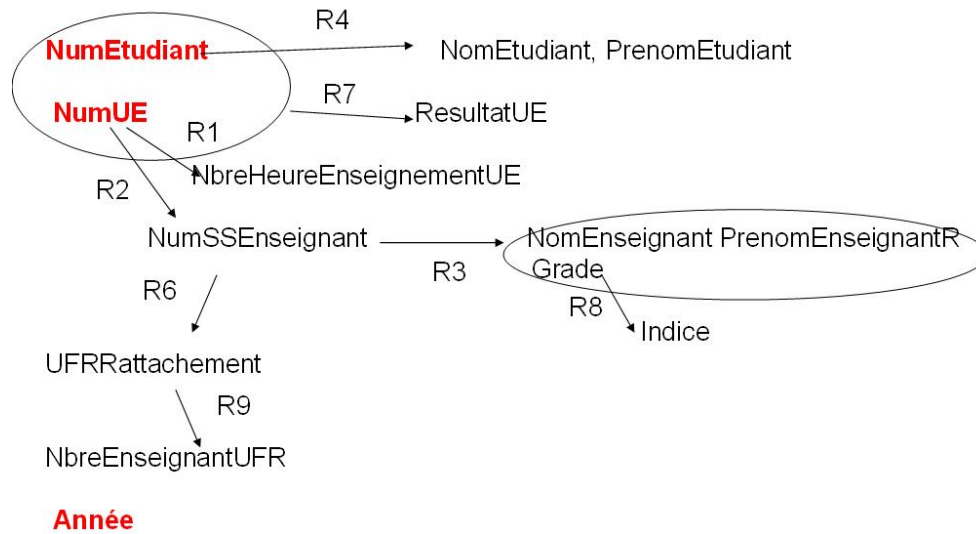


Figure 6.1 : Les dépendances fonctionnelles

Nous allons montrer que $(\text{NumEtudiant}, \text{NumUE}, \text{Annee})$ est une clé candidate de la relation SCOLARITE.

$K^+ = \text{NumEtudiant}, \text{NumUE}, \text{Annee}$

R1 : $\text{NumUE} \in K^+$ donc $K^+ = K^+ \cup \text{NbreHeureEnseignementUE}$

R2 : $\text{NumUE} \in K^+$ donc $K^+ = K^+ \cup \text{NumSSEnseignant}$

R3 : $\text{NumSSEnseignant} \in K^+$ donc $K^+ = K^+ \cup \text{NomEnseignant}, \text{PrenomEnseignant}, \text{Grade}$

R4 : $\text{NumEtudiant} \in K^+$ donc $K^+ = K^+ \cup \text{NomEtudiant}, \text{PrenomEtudiant}$

R6 : $\text{NumSSEnseignant} \in K^+$ donc $K^+ = K^+ \cup \text{UFRRattachement}$

R7 : $\text{NumEtudiant}, \text{NumUE} \in K^+$ donc $K^+ = K^+ \cup \text{ResultatUE}$

R8 : $\text{Grade} \in K^+$ donc $K^+ = K^+ \cup \text{Indice}$

R9 : $\text{UFRRattachement} \in K^+$ donc $K^+ = K^+ \cup \text{NbreEnseignantUFR}$

Nous avons donc :

$K^+ = (\text{NumEtudiant}, \text{NumUE}, \text{NbreHeureEnseignementUE}, \text{NumSSEnseignant}, \text{NomEnseignant}, \text{PrenomEnseignant}, \text{Grade}, \text{Indice}, \text{Annee}, \text{UFRRattachement}, \text{NbreEnseignantUFR}, \text{ResultatUE}, \text{NomEtudiant}, \text{PrenomEtudiant})$

Donc l'ensemble d'attributs : $(\text{NumEtudiant}, \text{NumUE}, \text{Annee})$ est une **super-clé** de la relation SCOLARITE. Par ailleurs il est immédiat que si l'on retire de cet ensemble un de ses attributs nous ne retrouverons pas tous les attributs de SCOLARITE. La super-clé est donc **irréductible**, nous avons une **clé candidate**.

6.2 Exercice 2 - Partie 4 : Père Mère et Enfants

Question 1

La représentation des DFs donne :

Montrons que $(\text{NomE}, \text{PrenomE})$ est une clé candidate de PME. Pour cela nous allons utiliser l'algorithme qui permet à partir d'un ensemble d'attributs, de calculer l'ensemble des attributs qu'ils définissent fonctionnellement par rapport à un ensemble de dépendances fonctionnelles de référence.

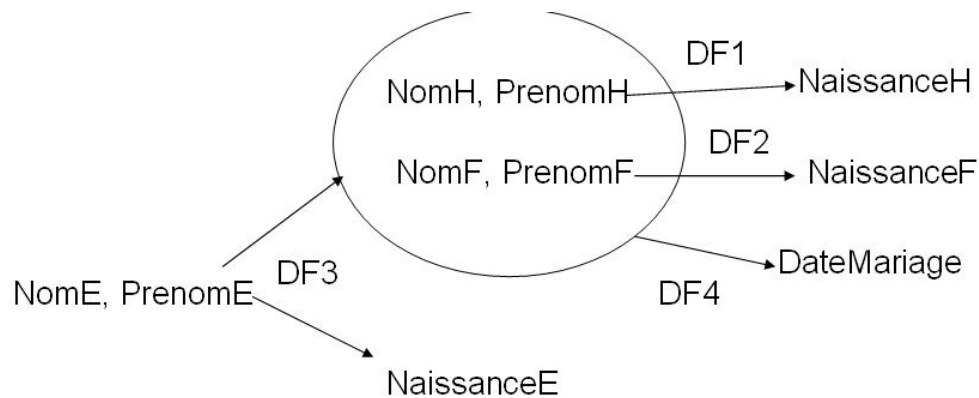


Figure 6.2 : Les DFs de PME

$A^+ = \text{NomE, PrenomE}$

Première Itération :

DF1 : $(\text{NomH, PrenomH}) \notin A^+$. Pas de changement.

DF2 : $(\text{NomF, PrenomF}) \notin A^+$. Pas de changement.

DF3 : $(\text{NomE, PrenomE}) \in A^+$. On a donc :

$A^+ = A^+ \cup (\text{NaissanceE, NomH, PrenomH, NomF, PrenomF})$

$A^+ = (\text{NomE, PrenomE, NaissanceE, NomH, PrenomH, NomF, PrenomF})$

DF4 : $(\text{NomF, PrenomF, NomH, PrenomH}) \in A^+$. On a donc :

$A^+ = A^+ \cup \text{DateMariage}$

$A^+ = (\text{NomE, PrenomE, NaissanceE, NomH, PrenomH, NomF, PrenomF, DateMariage})$

Deuxième Itération :

DF1 : $(\text{NomH, PrenomH}) \in A^+$. On a donc :

$A^+ = A^+ \cup \text{NaissanceH}$ $A^+ = (\text{NomE, PrenomE, NaissanceE, NomH, PrenomH, NomF, PrenomF, DateMariage, NaissanceH})$ DF2 : $(\text{NomF, PrenomF}) \in A^+$. On a donc :

$A^+ = A^+ \cup \text{NaissanceF}$

$A^+ = (\text{NomE, PrenomE, NaissanceE, NomH, PrenomH, NomF, PrenomF, DateMariage, NaissanceH, NaissanceF})$

Nous pouvons nous arrêter ici car A^+ est égal à l'ensemble des attributs de la relation PME. (NomE, PrenomE) est donc une super-clé de cette relation. D'autre part nous voyons bien que cette clé n'est pas réductible. (NomE, PrenomE) est une clé candidate de PME.

Question 2

Pour que PME soit 2NF, il faut qu'il n'existe aucune dépendance fonctionnelle ayant une partie de la clé comme déterminant. Ici la clé est (NomE, PrenomE) , il n'y a aucune dépendance posant problème. PME est donc 2NF.

Question 3

Pour que PME soit 3NF, il faut qu'il n'existe aucune dépendance fonctionnelle ayant des attributs non clés comme déterminant. Ici les dépendances DF1, DF2 et DF4 posent problème. PME n'est

donc pas 3NF.

Transformation de PME en forme 3NF :

DF1 : La relation PME peut être décomposée en deux relations MARI qui contient les attributs de DF1, et R1 qui contient les attributs de la relation PME moins ceux qui apparaissent en partie droite de la dépendance DF1.

- MARI(*NomH, *PrenomH, NaissanceH)
- PME1(NomH, PrenomH, NomF, PrenomF, NaissanceF, *NomE, *PrenomE, NaissanceE, DateMariage)

DF2 : la relation PME1 peut être décomposée en deux relations EPOUSE qui contient les attributs de DF2, et PME2 qui contient les attributs de la relation PME1 moins ceux qui apparaissent en partie droite de la dépendance DF2. Soit :

- EPOUSE(*NomF, *PrenomF, NaissanceF)
- PME2(NomH, PrenomH, NomF, PrenomF, *NomE, *PrenomE, NaissanceE, DateMariage)

DF4 : la relation PME2 peut être décomposée en deux relations MARIAGE qui contient les attributs de DF4, et PME3 qui contient les attributs de la relation PME2 moins ceux qui apparaissent en partie droite de la dépendance DF4. Soit :

- MARIAGE(*NomH, *PrenomH,*NomF, *PrenomF, DateMariage)
- PME3(NomH, PrenomH, NomF, PrenomF, *NomE, *PrenomE, NaissanceE)

Nous avons donc défini l'ensemble de relations suivant qui est 3NF.

```
MARI(*NomH, *PrenomH, NaissanceH)
EPOUSE(*NomF, *PrenomF, NaissanceF)
MARIAGE(*NomH, *PrenomH,*NomF, *PrenomF, DateMariage)
PME3(NomH, PrenomH, NomF, PrenomF, *NomE, *PrenomE, NaissanceE)
```

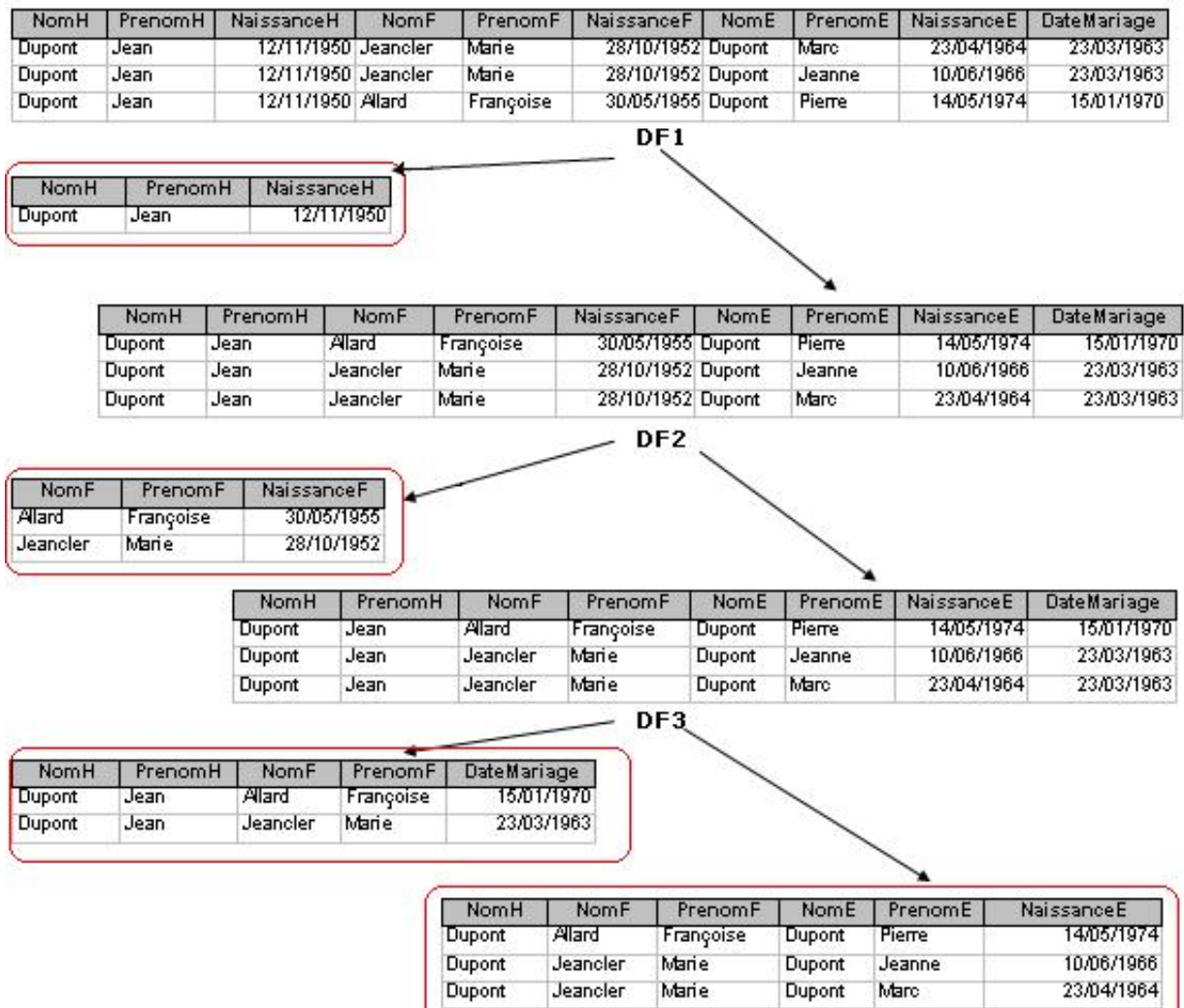


Figure 6.3 : Décomposition de PME en 3NF

6.3 Exercice 3 - Partie 4 : Professeurs - Cours et Matières

Question 1

Nous avons les dépendances suivantes :

DF1 : Classe, Matière \rightarrow Prof

DF2 : Prof \rightarrow Matière

Ces deux dépendances fonctionnelles représentent *l'ensemble noyau* de dépendances fonctionnelles.

Celui-ci peut être étendu par augmentation. Nous obtenons alors la dépendance :

DF3 : Prof, Classe \rightarrow Classe, Matière

Question 2

Nous avons les deux clés candidates suivantes :

- (Classe, Matière) en effet DF1 montre que ces deux attributs définissent fonctionnellement tous les attributs de la relation COURS.
- (Classe, Prof) en effet d'après DF3, ces deux attributs définissent fonctionnellement tous les autres attributs de la relation COURS.

Question 3

La dépendance DF2, nous montre que la relation COURS avec comme clé primaire (Classe, Prof), n'est pas 2NF, car une partie de la clé définit fonctionnellement un attribut non clé. Nous proposons donc de décomposer la relation COURS en deux relations : Une relation PROFESSEUR, qui contient les attributs impliqués dans DF2, et une relation CLASSEPROF, qui contient les attributs de la relation COURS, moins ceux apparaissant en partie droite de DF2. Ce qui nous donne :

- PROFESSEUR(*Prof, Matière)
- CLASSEPROF(*Classe, *Prof)



Une partie de la clé définit un attribut non clé → Relation non 2NF

Matiere	*Prof	*Classe
Maths	Monigadon	6B
Maths	Quere	5A
Dessin	Maire	5A
Dessin	Maire	6B

DF2

*Prof	*Classe
Monigadon	6B
Quere	5A
Maire	5A
Maire	6B

Matiere	*Prof
Maths	Monigadon
Maths	Quere
Dessin	Maire

Question 4

Prenons (Classe, Matière) comme clé primaire. Montrons que la relation COURS n'est pas BCNF. La dépendance DF2, nous montre qu'un attribut non clé définit fonctionnellement une partie de la clé primaire. La relation COURS n'est donc pas BCNF. Nous la décomposons en utilisant le théorème de décomposition, qui propose ici de créer les deux relations suivantes : Une relation PROFESSEUR, qui contient les attributs impliqués dans la DF : DF2, et une relation CLASSEPROF, qui contient les attributs de la relation COURS, moins ceux apparaissant en partie droite de DF2, de plus ici la clé primaire de la relation est modifiée l'attribut Prof est substitué à l'attribut Matière. Ce qui nous donne :

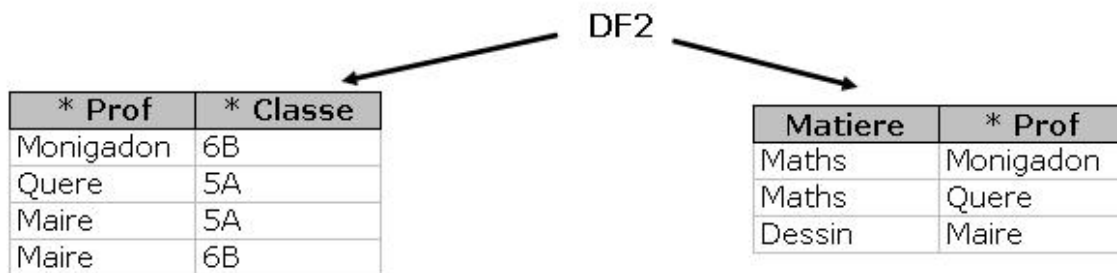
- PROFESSEUR(*Prof, Matière)
- CLASSEPROF(*Classe, *Prof)

Ce qui nous donne le même résultat qu'à la question précédente.



Un attribut non clé définit une partie de la clé → Relation non BCNF

* Matiere	Prof	* Classe
Maths	Monigadon	6B
Maths	Quere	5A
Dessin	Maire	5A
Dessin	Maire	6B



6.4 Exercice 4 - Partie 4 : Enfants et contrats

Question 1

Nous avons les dépendances suivantes :

DF1 : $N_Membre \rightarrow NomMembre, PrenomMembre$ DF2 : $N_Contrat \rightarrow CodeContrat$ DF3 : $CodeContrat \rightarrow TypeContrat$

Il est immédiat que la clé de la relation MEMBRE est : $(N_Membre, N_Enfant, N_Contrat)$. En effet ces trois attributs définissent tous les attributs de la relation MEMBRE, et ce de façon irréductible.

Question 2

La relation MEMBRE n'est pas 2NF, car les dépendances DF1, et DF2 ont comme déterminant une partie de la clé primaire de MEMBRE.

DF1 : La relation MEMBRE est décomposée en les deux relations suivantes :

- LEMEMBRE(*N_Membre, NomMembre, PrenomMembre)
- MEMBRE1(*N_Membre, * N_Enfant, *N_Contrat, CodeContrat, TypeContrat)

DF2 : Nous avons la dépendance :

DF2 : $N_Contrat \rightarrow CodeContrat$

mais aussi la dépendance

DF3 : $CodeContrat \rightarrow TypeContrat$

Par transitivité nous pouvons donc écrire : DF2' : $N_Contrat \rightarrow CodeContrat, TypeContrat$

Nous utiliserons cette dépendance pour la décomposition, en effet il ne faut pas séparer des informations "liées" lors d'une décomposition. Si nous n'utilisons ici que les attributs $N_Contrat$ et $CodeContrat$ pour notre décomposition, l'attribut $TypeContrat$ sera séparé de l'attribut $CodeContrat$ dont il dépend directement, ce qui n'est pas logique.

MEMBRE1 est donc décomposé comme suit :

- CONTRAT(*N_Contrat, CodeContrat, TypeContrat)
- MEMBRE2(*N_Membre, * N_Enfant, *N_Contrat)

Nous obtenons donc les relations suivantes qui sont 2NF.

- LEMEMBRE(*N_Membre, NomMembre, PrenomMembre)
- CONTRAT(*N_Contrat, CodeContrat, TypeContrat)
- MEMBRE2(*N_Membre, * N_Enfant, *N_Contrat)

avec les données :

N_Membre	NomMembre	PrenomMembre
1	Dupont	Jean
2	Dupont	Marie

Table 6.1 : Relation LEMEMBRE

N_Membre	N_Enfant	N_Contrat
1	3	011
1	3	012
1	3	123
1	4	011
1	4	012
1	4	123
2	3	124
2	4	124

Table 6.2 : Relation MEMBRE2

N_Contrat	CodeContrat	TypeContrat
011	AssV1	AssurVoiture
123	AssVi	AssuranceVie
012	AssV2	AssurVoiture
124	AssVi	AssuranceVie

Table 6.3 : relation CONTRAT

Question 3

La relation CONTRAT n'est pas 3NF car nous avons une dépendance entre deux attributs non clés : DF3. Nous proposons la décomposition suivante :

CONTRAT(*N_Contrat, CodeContrat, TypeContrat)

devient :

- CONTRAT(*N_Contrat, CodeContrat)
- CODECONTRAT(*CodeContrat, TypeContrat)

Les deux relations sont 3NF.

Nous obtenons donc les relations suivantes qui sont 3NF.

- LEMEMBRE(*N_Membre, NomMembre, PrenomMembre)

- CONTRAT(*N_Contrat, CodeContrat)
- CODECONTRAT(*CodeContrat, TypeContrat)
- MEMBRE2(*N_Membre, * N_Enfant, *N_Contrat)

De plus nous pouvons immédiatement préciser que ces relations sont BCNF, car chacune ne possède qu'une seule clé candidate.

N_Contrat	CodeContrat
011	AssV1
123	AssVi
012	AssV2
124	AssVi

Table 6.4 : relation CONTRAT

CodeContrat	TypeContrat
AssV1	AssurVoiture
AssVi	AssuranceVie
AssV2	AssurVoiture

Table 6.5 : relation CODECONTRAT

Question 4

Pour une personne donnée nous pouvons avoir plusieurs enfants et plusieurs contrats. De plus les enfants et les contrats sont complètement indépendants. Nous avons donc une dépendance multivaluée de la forme :

$N_Membre \twoheadrightarrow N_Enfant / N_Contrat$

Le théorème de FAGIN permet de décomposer cette relation en deux relations de la façon suivante :

- MEMBREENFANT(*N_Membre, *N_Enfant)
- MEMBRECONTRAT(*N_Membre, *N_Contrat)

Nous obtenons pour finir l'ensemble de relations suivant qui est 4NF.

- LEMEMBRE(*N_Membre, NomMembre, PrenomMembre)
- CONTRAT(*N_Contrat, CodeContrat)
- CODECONTRAT(*CodeContrat, TypeContrat)
- MEMBREENFANT(*N_Membre, *N_Enfant)
- MEMBRECONTRAT(*N_Membre, *N_Contrat)

N_Membre	N_Enfant
1	3
1	4
2	3
2	4

Table 6.6 : Relation MEMBREENFANT

N_Membre	N_Contrat
1	011
1	123
1	012
2	124

Table 6.7 : Relation MEMBRECONTRAT

6.5 Exercice 5 - Partie 4 : Maîtresse de maison

Question 1

Etant donné qu'il n'y a aucune dépendance fonctionnelle dans la relation LESREPAS, un t-uplet est caractérisé par **tous** ses attributs. La clé de cette relation est donc l'ensemble de ses attributs :

LESREPAS(*Date,* Plat, *Invite,* Preference)

Question 2

La relation LESREPAS est 2NF, car tous les attributs font partie de la clé. Il n'y a donc pas de dépendance entre une partie de la clé, et des attributs non clés.

Question 3

Comme précédemment, cette relation ne contient aucun attribut non clé. Elle est donc 3NF.

Question 4

De la même façon, nous n'avons pas de problème de dépendance d'attributs non clé vers un attribut clé, étant donné que tous les attributs font partie de la clé.

Question 5

Nous pouvons remarquer que s'il n'y a aucune dépendance fonctionnelle entre les différents attributs de la relation LESREPAS, il y a cependant des "dépendances" entre certains. Par exemple, on se rend bien compte qu'une date définit plusieurs invités, et de façon indépendante plusieurs plats. De même un invité définit plusieurs préférences indépendamment du reste.

Nous avons une première dépendance multi-valuée qui est :

Date ->> Plat / (Invite, Preference)

Nous proposons donc la décomposition suivante :

- MENU(*Date, * Plat)
- LESINVITES(*Date, *Invite, *Preference)

Maintenant si nous tenons observons la relation LESINVITES, nous voyons que nous avons la dépendance multi-valuée suivante :

Invite ->> Date / Preference

D'où la décomposition de la relation LESINVITES en :

- REPAS(*Date, *Invite)
- PREFERENCES(*Invite, *Preference)

Ce qui nous donne les tables :

- MENU(*Date, * Plat)

plat	personne	date
sorbet	alice	03/10/2009
dinde	jean	03/10/2009
soupe	jean	03/10/2009
dinde	jean	26/12/2009
soupe	jean	26/12/2009
lotte à l'américaine	marie	03/10/2009
foie gras	pierre	03/10/2009
foie gras	pierre	26/12/2009

Table 6.8 : Relation LESINVITES

Invite	Date
alice	03/10/2009
jean	03/10/2009
jean	26/12/2009
marie	03/10/2009
pierre	03/10/2009
pierre	26/12/2009

Table 6.9 : Relation REPAS

- REPAS(*Date, *Invite)
- PREFERENCES(*Invite, *Preference)

date	plat
03/10/2009	gigot
03/10/2009	salade niçoise
03/10/2009	sorbet
26/12/2009	dinde
26/12/2009	foie gras
26/12/2009	sorbet

Table 6.10 : Relation MENU

Personne	Plat
alice	sorbet
henri	sorbet
jean	dinde
jean	soupe
marie	lotte à l'américaine
pierre	foie gras
sophie	poulet

Table 6.11 : relation PREFERENCE

6.6 Exercice 6 - Partie 4 : Entreprise

Question 1

Montrons que le groupe d'attributs (N_{Employe} , N_{Enfant} , N_{Tache}) est une clé candidate de la relation UNIV. Pour cela nous devons montrer que ce groupe d'attributs est une super-clé de la relation UNIV et qu'il est irréductible. Pour cela nous allons utiliser l'algorithme qui permet à partir d'un ensemble d'attributs, de calculer l'ensemble des attributs qu'ils définissent fonctionnellement par rapport à un ensemble de dépendances fonctionnelles de référence.

$K^+ = N_{\text{Employe}}, N_{\text{Enfant}}, N_{\text{Tache}}$

Première Itération :

DF1 : ($N_{\text{Entreprise}}$) $\notin K^+$. Pas de changement.

DF2 : (N_{Employe}) $\in K^+$. On a donc :

$K^+ = K^+ \cup (N_{\text{Employe}}, N_{\text{Entreprise}})$

DF3 : ($N_{\text{Responsable}}$) $\notin K^+$. Pas de changement.

Deuxième Itération :

DF1 : ($N_{\text{Entreprise}}$) $\in K^+$. On a donc :

$\text{texttt}K^+ = K^+ \cup (N_{\text{Entreprise}}, \text{RueEntreprise}, \text{VilleEntreprise}, \text{CPEntreprise}, N_{\text{Responsable}})$

DF3 : ($N_{\text{Responsable}}$) $\in K^+$. On a donc :

$K^+ = K^+ \cup N_{\text{Responsable}}$

K^+ contient tous les attributs de la relation UNIV, nous avons donc une **super-clé**. Cette super-clé est **irréductible**, si on retire N_{Enfant} ou N_{Tache} , ces deux attributs n'apparaissant dans aucune dépendance fonctionnelle nous ne pourrions pas les retrouver. Si on retire N_{Employe} on ne pourra pas non plus retrouver l'employé, et son entreprise.

($N_{\text{Employe}}, N_{\text{Enfant}}, N_{\text{Tache}}$) est donc une **clé candidate** de la relation UNIV.

Question 2

La relation UNIV n'est pas 2NF à cause de DF2. La décomposition de la relation UNIV donne les deux relations suivantes :

- EMPLOYE($N_{\text{Entreprise}}, N_{\text{NomEntreprise}}, \text{RueEntreprise}, \text{VilleEntreprise}, \text{CPEntreprise}, *N_{\text{Employe}}, N_{\text{NomEmploye}}, N_{\text{Responsable}}, N_{\text{NomResponsable}}$)
- UNIV1($*N_{\text{Employe}}, *N_{\text{Tache}}, *N_{\text{Enfant}}$)

Question 3

La relation UNIV1 est 3NF (Il n'y a pas d'attributs non clés donc aucune dépendance entre attributs non clés). Par contre la relation EMPLOYE n'est pas 3NF à cause de DF1 et DF3.

Une première décomposition de EMPLOYE par rapport à DF1 donne les deux relations :

- ENTREPRISE($*N_{\text{Entreprise}}, N_{\text{NomEntreprise}}, \text{RueEntreprise}, \text{VilleEntreprise}, \text{CPEntreprise}, N_{\text{Responsable}}, N_{\text{NomResponsable}}$)
- EMPLOYE1($N_{\text{Entreprise}}, *N_{\text{Employe}}, N_{\text{NomEmploye}}$)

* N° Employe	Nom Employe	N° Entreprise	Nom Entreprise	Rue Entreprise	Ville Entreprise	CP Entreprise	N° Responsable	Nom responsable	* N° Enfant	* N° Tache
E1	DUPONT	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE1	T1
E1	DUPONT	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE1	T2
E2	DURAND	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE2	T1
E2	DURAND	ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT	EE3	T1
E3	DUBOIS	ENT2	OLSA	Grande Rue	PONTARLIER	25500	R4	ANDREY	EE4	T1
E3	DUBOIS	ENT2	OLSA	Grande Rue	PONTARLIER	25500	R4	ANDREY	EE4	T3

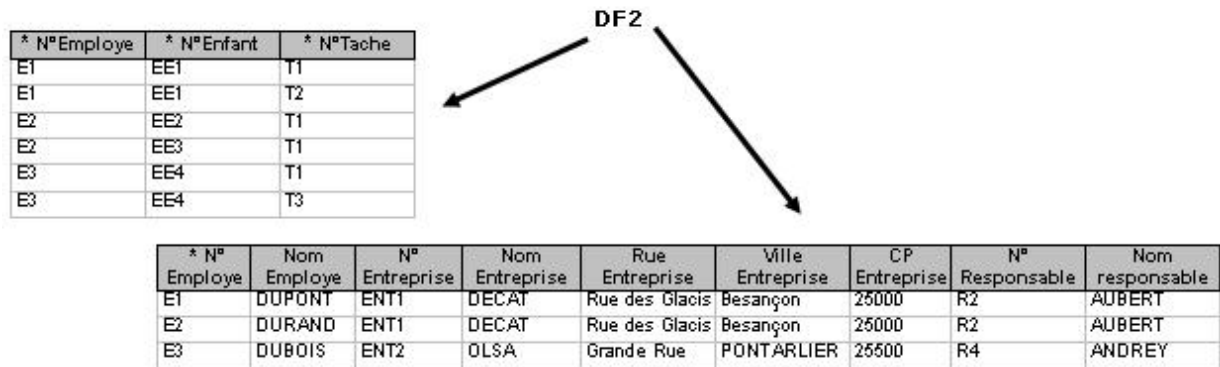


Figure 6.4 : ENTREPRISE : Décomposition NF

La relation EMPLOYE1 est 3NF par contre ENTREPRISE ne l'est toujours pas à cause de DF3. La décomposition de ENTREPRISE par rapport à DF3 donne les deux relations :

- ENTREPRISE1(* NEntreprise, NomEntreprise, RueEntreprise, VilleEntreprise, CPEntreprise, NResponsable)
- RESPONSABLE(*NResponsable, NomResponsable)

* N° Entreprise	Nom Entreprise	Rue Entreprise	Ville Entreprise	CP Entreprise	N° Responsable	Nom responsable
ENT1	DECAT	Rue des Glacis	Besançon	25000	R2	AUBERT
ENT2	OLSA	Grande Rue	PONTARLIER	25500	R4	ANDREY

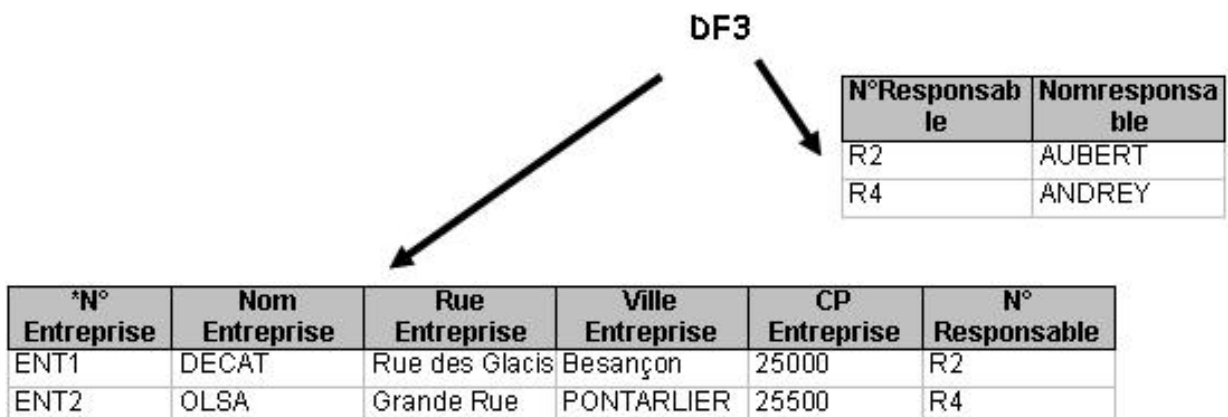


Figure 6.5 : ENTREPRISE Décomposition NF

Question 4

Nous n'avons pas de pb de forme BCNF, en effet toutes les relations n'ont qu'une seule clé candidate et il n'y a pas de dépendance de type attribut non clé vers attributs clé.

Question 5

Pour savoir si les relations sont 4NF, il faut déjà regarder si nous avons dans nos relations une ou plusieurs **dépendances multi-valuées**. Nous avons effectivement la dépendance multi-valuée : $N\text{Employe} \twoheadrightarrow N\text{Enfant} / N\text{Tache}$

En effet un employé peut avoir plusieurs enfants et être affecté à différentes tâches et ceci de façon totalement indépendante. Le déterminant de cette dépendance n'est pas une **super-clé** de UNIV. Cette relation n'est donc pas 4NF. Le théorème de FAGIN nous donne la décomposition suivante :

- EMPLOYEENFANT(* NEmploye, *NEnfant)
- TACHE EMPLOYE(* NEmploye, *NTache)

Ce qui nous donne à la finale l'ensemble de relations :

- ENTREPRISE1(* NEntreprise, NomEntreprise, RueEntreprise, VilleEntreprise, CPEntreprise, NResponsable)
- RESPONSABLE(*NResponsable, NomResponsable)
- EMPLOYE1(NEntreprise, * NEmploye, NomEmploye)
- EMPLOYEENFANT(* NEmploye, *NEnfant)
- TACHE EMPLOYE(* NEmploye, *NTache)

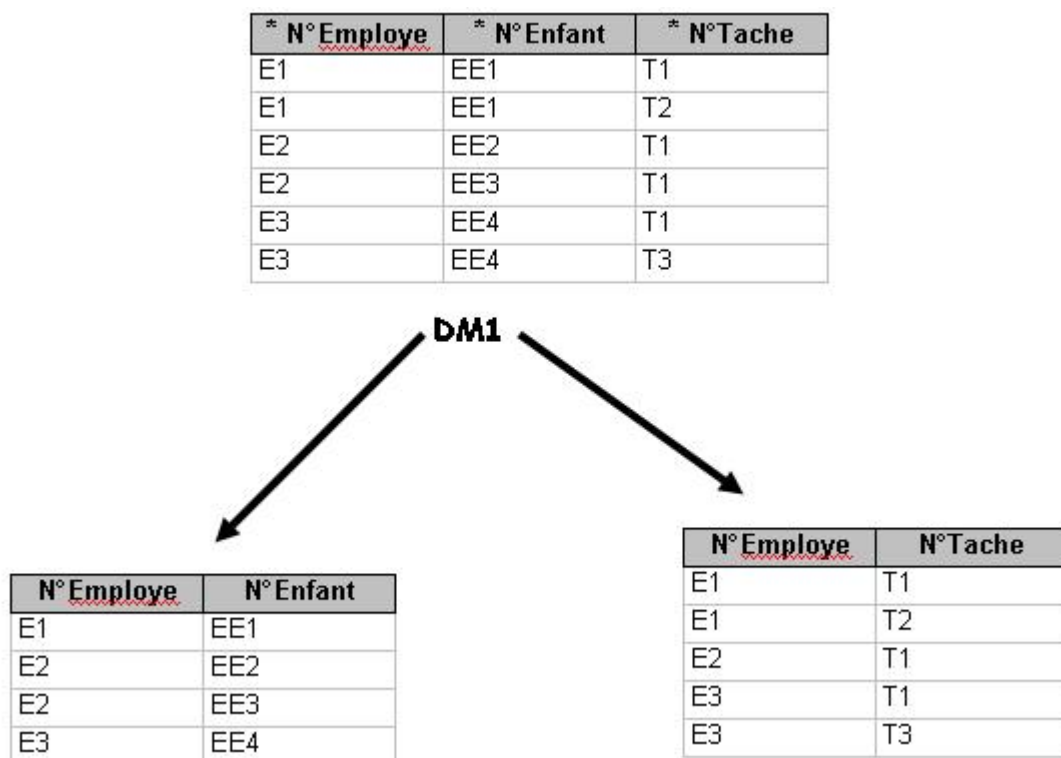


Figure 6.6 : ENTREPRISE : Décomposition 4NF

6.7 Exercice 7 - Partie 4 : Fournisseurs, Pièces et projets

Question 1

Montrons que le groupe d'attributs (**Fournisseur**, **Piece**, **Projet**) est une clé candidate de la relation UNIV. Pour cela nous devons montrer que ce groupe d'attributs est une **super-clé** de la relation UNIV et qu'il est **irréductible**. Pour cela nous allons utiliser l'algorithme qui permet à partir d'un ensemble d'attributs, de calculer l'ensemble des attributs qu'ils définissent fonctionnellement par rapport à un ensemble de dépendances fonctionnelles de référence.

$K^+ = \text{Fournisseur, Piece, Projet}$

Première Itération :

DF1 : (**Fournisseur**) $\in K^+$. On a donc :

$K^+ = K^+ \cup (\text{Nom, Rue, CP})$

DF2 : (**Piece**) $\in K^+$. On a donc :

$K^+ = K^+ \cup (\text{LibPiece})$

DF3 : (**Projet**) $\in K^+$. On a donc :

$K^+ = K^+ \cup (\text{LibProjet, Responsable})$

DF3 : (**CP**) $\in K^+$. On a donc :

$K^+ = K^+ \cup (\text{Ville})$

K^+ contient tous les attributs de la relation UNIV, nous avons donc une **super-clé**. Cette super-clé est **irréductible**, si on retire l'attribut **Fournisseur** on ne pourra pas retrouver les informations concernant le fournisseur (**nom, rue, CP**), même chose pour les attributs **Piece** et **Projet**.

(**Fournisseur, Piece, Projet**) est donc une **clé candidate** de la relation UNIV.

Question 2

La relation UNIV n'est pas 2NF à cause des DF : 1, 2 et 3, qui ont comme déterminant une partie de la clé primaire de UNIV.

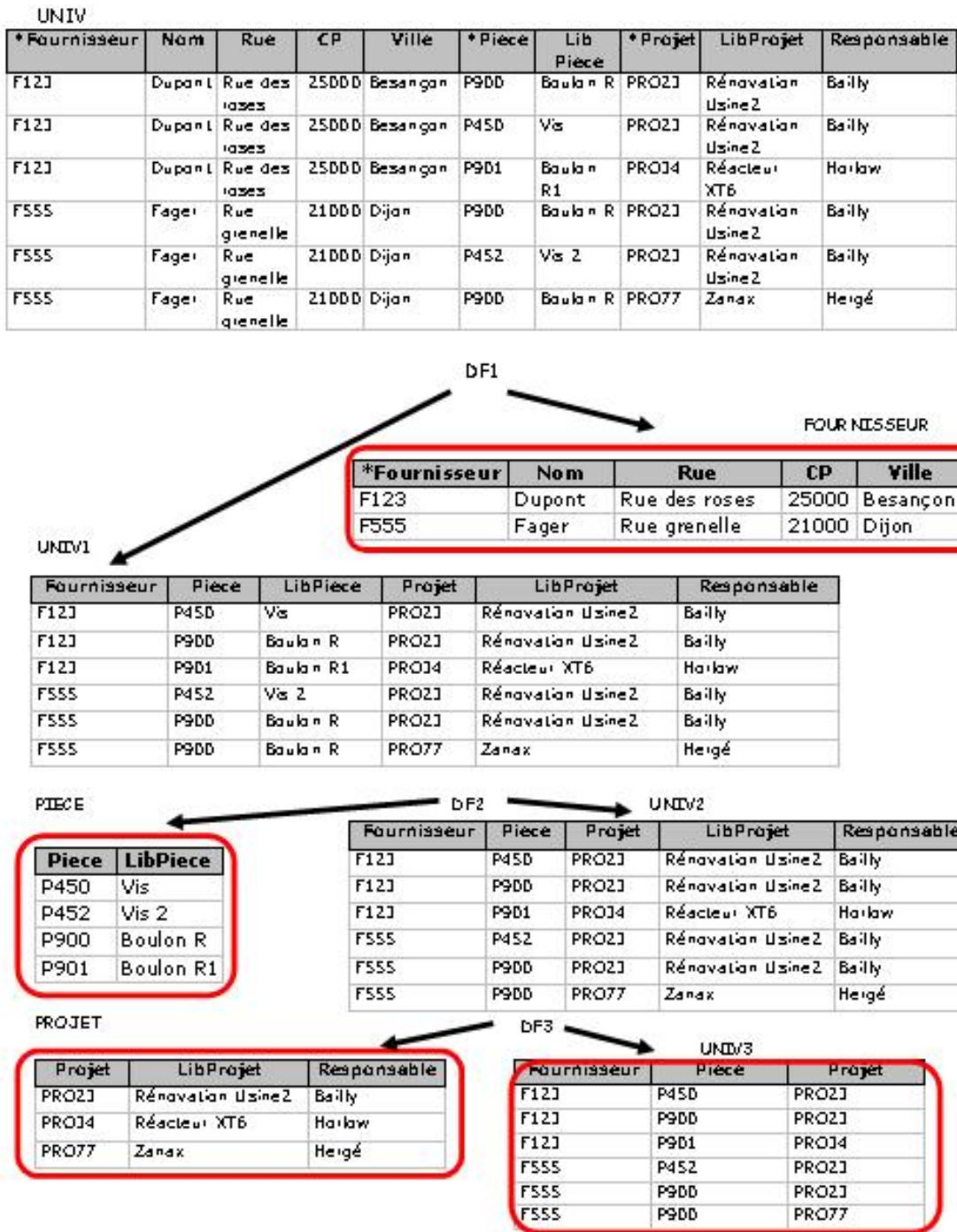


Figure 6.7 : FOURNISSEUR : Décomposition 2NF

DF1 : La relation UNIV est décomposée en les deux relations suivantes :

- FOURNISSEUR(*Fournisseur, Nom, Rue, CP, Ville)
- UNIV1(*Fournisseur, *Piece, LibPiece, *Projet, LibProjet, Responsable)

Attention l'attribut Ville doit être placé dans la relation FOURNISSEUR, car par transitivité cet attribut dépend de l'attribut Fournisseur.

La relation UNIV1 n'est pas 2NF à cause de DF2. sa décomposition donne les 2 relations :

- PIECE(*Piece, LibPiece)
- UNIV2(*Fournisseur, *Piece, *Projet, LibProjet, Responsable)

La relation UNIV2 n'est pas 2NF à cause de DF3. sa décomposition donne les 2 relations :

- PROJET(*Projet, LibProjet, Responsable)
- UNIV3(*Fournisseur, *Piece, *Projet)

Nous obtenons donc les relations 2NF :

- FOURNISSEUR(*Fournisseur, Nom, Rue, CP, Ville)
- PIECE(*Piece, LibPiece)
- PROJET(*Projet, LibProjet, Responsable)
- UNIV3(*Fournisseur, *Piece, *Projet)

Question 3

La relation FOURNISSEUR n'est pas 3NF car nous avons la dépendance DF4 entre attributs non clés. La décomposition de FOURNISSEUR donne les 2 relations :

- FOURNISSEUR1(*Fournisseur, Nom, Rue, CP)
- VILLE(*CP, Ville)

FOURNISSEUR

*Fournisseur	Nom	Rue	CP	Ville
F123	Dupont	Rue des roses	25000	Besançon
F555	Fager	Rue grenelle	21000	Dijon

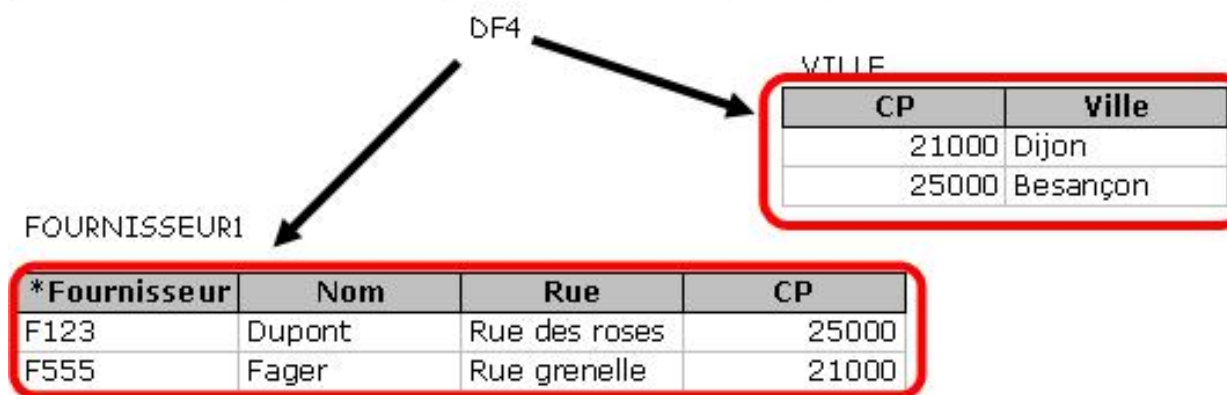


Figure 6.8 : FOURNISSEUR : Décomposition 3NF

Question 4

Nous n'avons pas de pb de forme BCNF, en effet toutes les relations n'ont qu'une seule clé candidate et il n'y a pas de dépendance de type attribut non clé vers attributs clé.

Question 5

Pour savoir si les relations sont 4NF, il faut déjà regarder si nous avons dans nos relations une ou plusieurs **dépendances multi-valuées**.

Les relations FOURNISSEUR1, PIECE, PROJET et VILLE ne posent pas de problème car nous avons pour chacune une DF qui lie la clé et les attributs non clés. Par contre la relation UNIV3 contient 3 attributs qui ne sont liés par aucune DF. Regardons si dans cette relation nous avons ou non une dépendance multi-valuée.

Pour avoir une dépendance multi-valuée de la forme $A1 \twoheadrightarrow A2/A3$, dans une relation nous devons tout d'abord pouvoir partitionner l'ensemble des attributs de la relation en les 3 groupes d'attributs $A1, A2, A3$. Ces groupes d'attributs sont tels que pour une valeur de $A1$ on a plusieurs valeurs de $A2$ et de $A3$. Et les valeurs de $A2$ et de $A3$ sont indépendantes. Ce qui s'exprime par la propriété vue en cours si les t-uplets $(a1, a2, a3)$ et $(a1, a2', a3')$ existent alors les t-uplets $(a1, a2', a3)$ et $(a1, a2, a3')$ existent.

Montrons que nous n'avons pas la DM : Fournisseur \twoheadrightarrow Piece / Projet.

En effet nous avons les t-uplets $(F123, P450, PRO23)$ et $(F123, P901, PRO34)$, mais pas le

t-uplet $(F123, P901, PRO23)$. Nous montrons de la même manière que nous n'avons pas les DM : $Piece \twoheadrightarrow Fournisseur / Projet$ ou $Projet \twoheadrightarrow Fournisseur / Piece$.

Nous n'avons donc aucune dépendance multi-valuée dans UNIV3, nos relations sont donc 4NF. L'image ci-dessous montre pourquoi par exemple on a pas la DM : $Fournisseur \twoheadrightarrow Piece / Projet$. Les lignes en rouge qui apparaissent après la recomposition de la relation par jointure sont incorrectes.

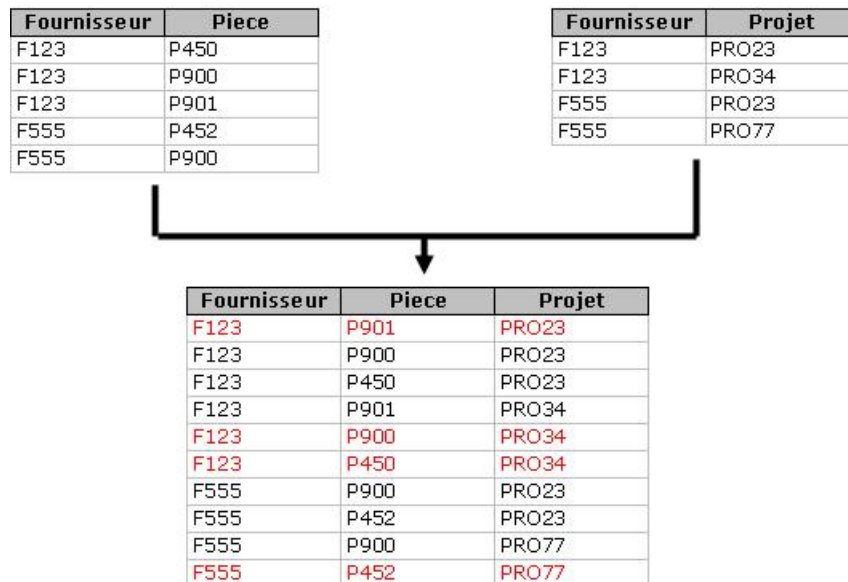


Figure 6.9 : Décomposition 4NF ?

Question 6

Regardons l'ensemble des relations obtenues. Comme nous l'avons dit à la question précédente, les relations FOURNISSEUR1, PIECE, PROJET et VILLE ne posent pas de problème car nous avons pour chacune une DF qui lie la clé et les attributs non clés. Par contre la relation UNIV3 contient 3 attributs qui ne sont liés par aucune DF. Regardons si dans cette relation nous avons ou non une **dépendance de jointure**.

Nous avons vu à la question précédente que la jointure des relations composées des attributs $(*Fournisseur, *Piece)$ et $(*Fournisseur, *Projet)$ donnait une relation UNIV4 qui contient des t-uplets supplémentaires par rapport à UNIV3. Considérons maintenant la table constituée des attributs $(*Piece, *Projet)$ et réalisons la jointure avec la table UNIV4. Nous obtenons UNIV3. Nous avons donc la dépendance de jointure suivante $*((*Fournisseur, *Piece), (*Fournisseur, *Projet), (*Piece, *Projet))$.

6.8 Exercice 8 - Partie 4 : Les magasins, leurs produits et leurs fournisseurs

Question 1

La relation UNIV est elle en 2NF ? Si non, la transformer en utilisant le théorème de décomposition, en un ensemble de relations 2NF. Les dépendances DF1 (et DF2 par transitivité), DF3 et

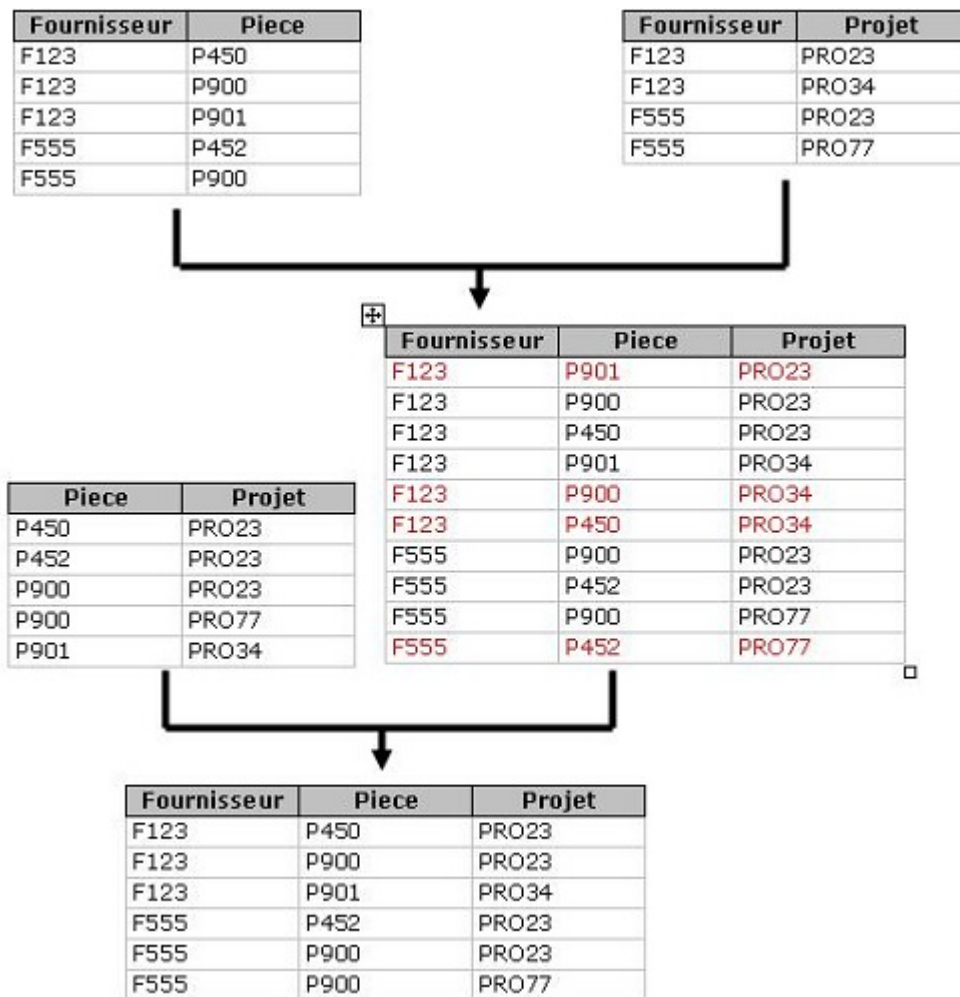


Figure 6.10 : FOURNISSEUR : Décomposition 5NF

DF4 posent problème pour la forme 2NF. Voici un schéma qui résume les décompositions réalisées :

Question 2

Les relations obtenues sont-elles en 3NF ? Si non, les transformer en utilisant le théorème de décomposition. Ici la relation PRODUIT n'est pas 3NF à cause de DF2. ce qui nous donne :

Question 3

Les relations obtenues sont-elles en BCNF ? Si non, les transformer en utilisant le théorème de décomposition. Il n'y a pas de problème toutes les relations n'ont qu'une seule clé candidate.

Question 4

Décrire en termes de dépendances multivaluées ou de dépendances de jointure les règles de gestion suivantes :

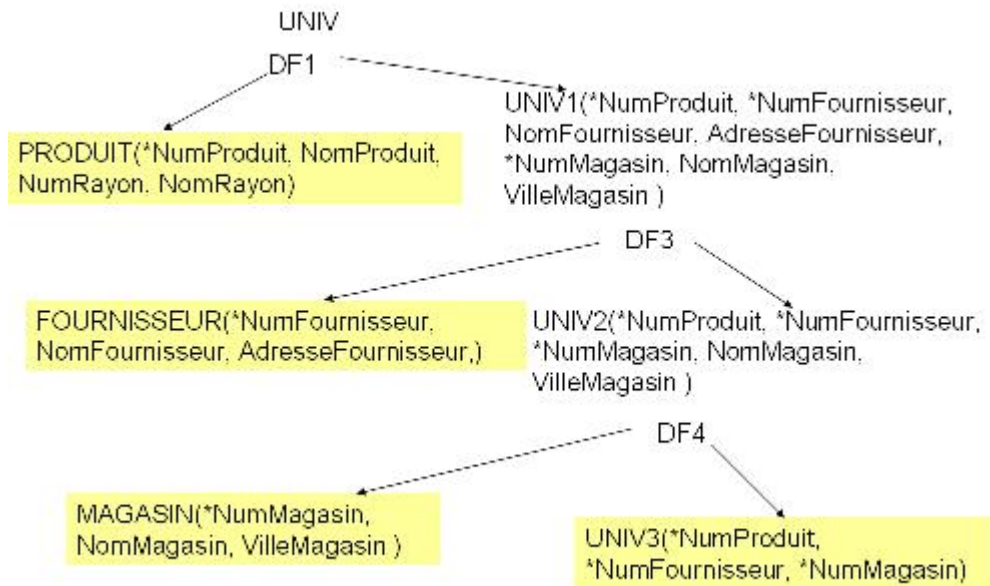


Figure 6.11 : Décomposition 2NF

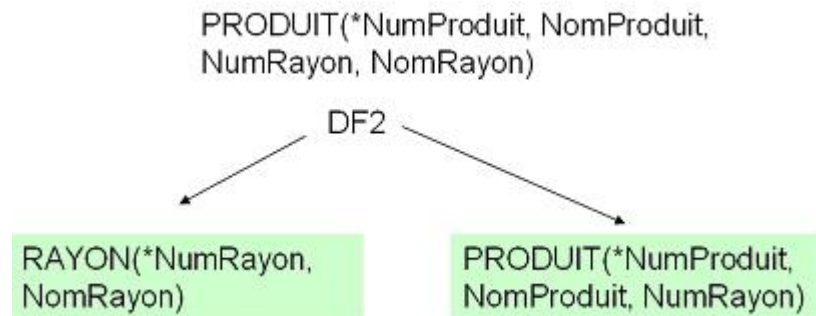


Figure 6.12 : Décomposition 3NF

Scénario 1

- Un produit peut être réalisé par plusieurs fournisseurs
- Un produit peut être à la vente dans plusieurs magasins

On a la dépendance multi-valuée :

DM : NumProduit $\rightarrow\rightarrow$ NumFournisseur, NumMagasin

Et le déterminant de cette DM n'est pas une super-clé de PRODUIT. cette relation n'est donc pas 4NF, d'où la décomposition :

PRODFOUR(*NumProduit, *NumFour)

PRODMAG(*NumProduit, *NumMag)

Scénario 2

- Un produit peut être réalisé par plusieurs fournisseurs
- Un produit peut être à la vente dans plusieurs magasins

- Un magasin a un ensemble de fournisseurs spécifiques

Ici on a une dépendance de jointure qui n'est pas triviale :

DJ $*$ ((NumProd, NumFour), (NumProd, NumMag), (NumMag, NumFour)) La relation PRODUIT n'est donc pas 5NF. D'où la décomposition :

PRODFOUR(*NumProduit, *NumFour)

PRODMAG(*NumProduit, *NumMag)

MAGFOUR(*NumMag, * NumFour)

Scénario 3

- Un produit peut être réalisé par plusieurs fournisseurs
- Un produit peut être à la vente dans plusieurs magasins
- Un magasin peut choisir pour chaque produit ses fournisseurs (dans la liste des fournisseurs de ce produit évidemment).

Ici on a une dépendance de jointure aussi :

DJ : DJ $*$ ((NumProd, NumFour), (NumProd, NumMag), (NumMag, NumFour, NumProd))

Cette DJ est triviale , donc la relation PRODUIT est 5NF.