

# Systemes de Gestion de Base de données

Guilaine Talens

[Guilaine.talens@univ-lyon3.fr](mailto:Guilaine.talens@univ-lyon3.fr)

IAE - Université Lyon3

# PLAN

**Un modèle, une méthode pour élaborer une base de données :**

- Modélisation des données avec UML
- Concepts de bases de données
- Langage de manipulation de données

# I - UML

En 1996, l'OMG lance un appel d'offres pour un langage de modélisation devant la menace d'un standard privé.

UML (Unified Modeling Language) est un standard sous le contrôle de l'OMG.

UML est un ensemble de notations graphiques (modèles) qui s'appuient sur une syntaxe (métamodèle).

Le langage a évolué par versions successives, octobre 2004 UML 2.0, novembre 2007 UML 2.1

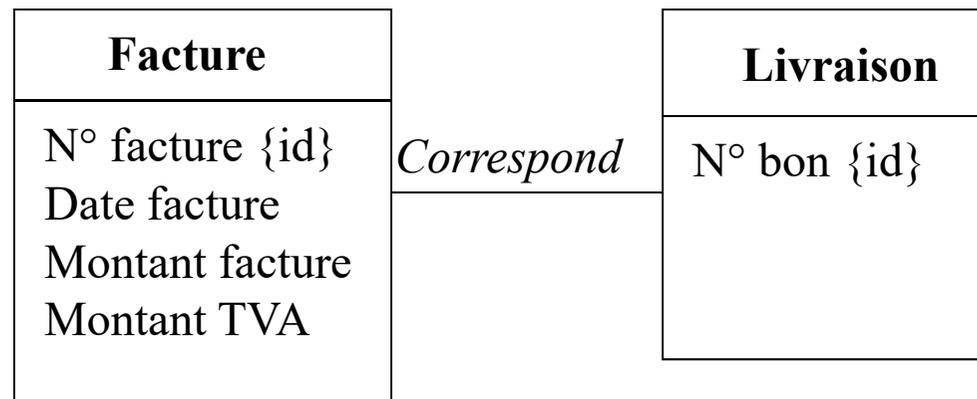
Un diagramme est défini comme la représentation graphique d'un ensemble sélectionné de constituants UML. Il en existe treize.

# Diagramme de classes

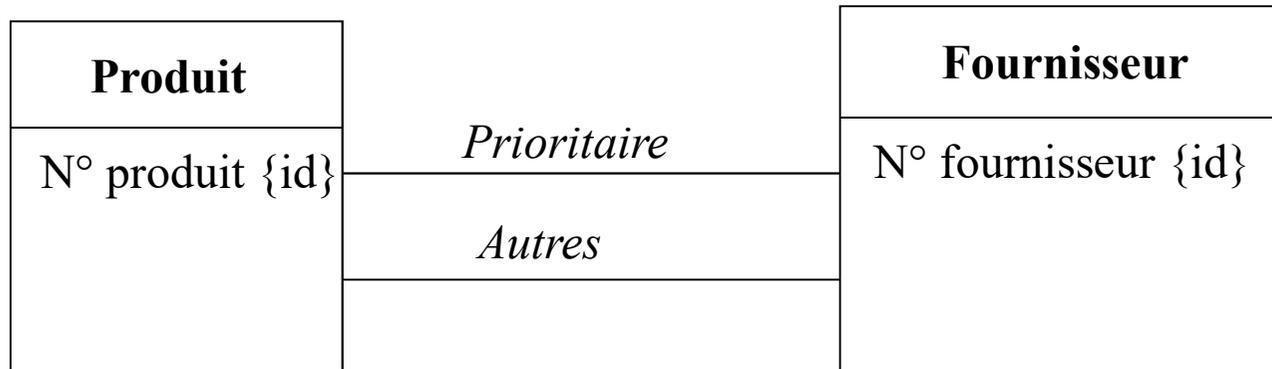
Un diagramme de classes représente la structure statique d'un système. Il contient des classes ainsi que leurs associations.

Une **classe** correspond à un concept global d'information et se décompose d'un ensemble d'informations élémentaires, appelées **attributs** de la classe.

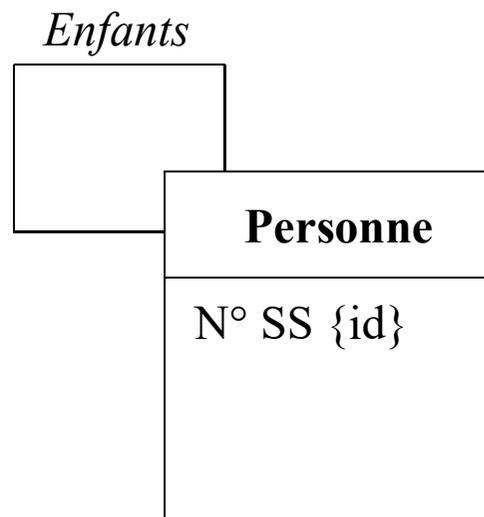
L'identifiant de la classe est l'attribut qui permet de référencer de façon unique chaque objet de la classe.



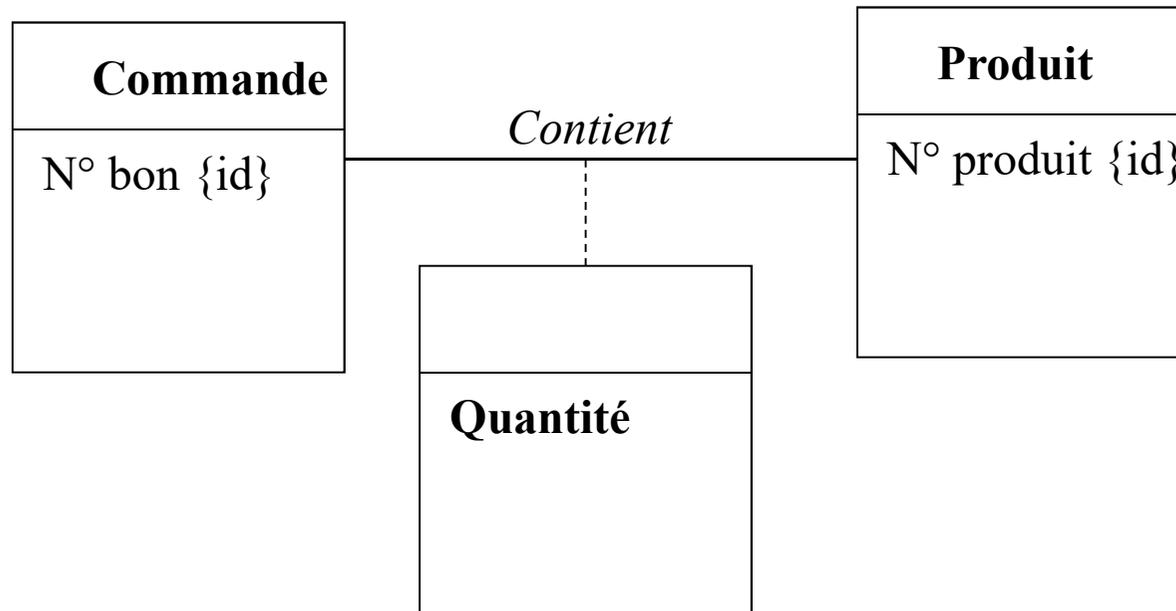
Une **association** a une signification. Entre deux classes on peut établir plusieurs associations. Chaque association porte un nom. On utilise le format italique.



Une association est dite réflexive si elle met en relation une classe avec elle-même.



Une association porteuse d'attributs est appelée **classe-association**.



On peut également définir une classe.

On aurait une classe Ligne commande qui contiendrait l'attribut quantité. Elle serait associée par une première association avec la classe Commande et par une deuxième avec la classe Produit.

On appelle **arité** d'une association le nombre de classes qui participent à l'association. La plupart sont binaires.

On caractérise une association par une information, portée à chacune de ses extrémités, qui indique le nombre d'instances : la **multiplicité**. En général, on fait apparaître deux nombres, représentant le minimum obligatoire et le maximum autorisé. Parfois, ces deux nombres sont égaux.

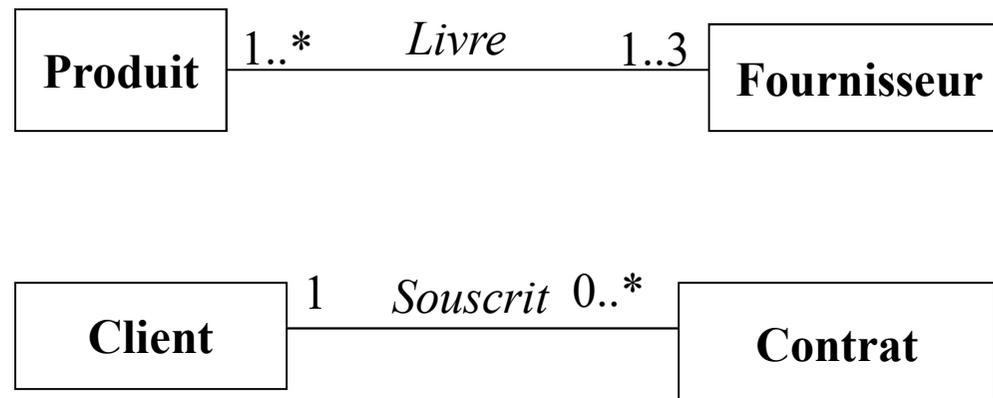
On utilise généralement les valeurs : 0, 1 et \*.

Parfois, on n'utilise qu'un seul nombre, le second étant implicite :

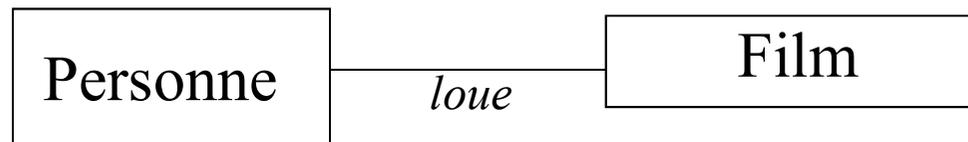
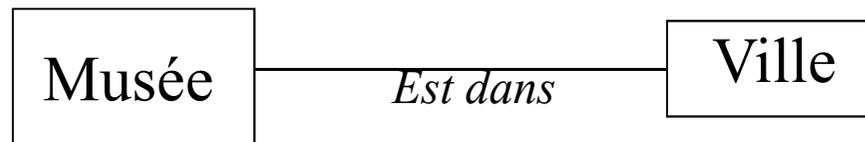
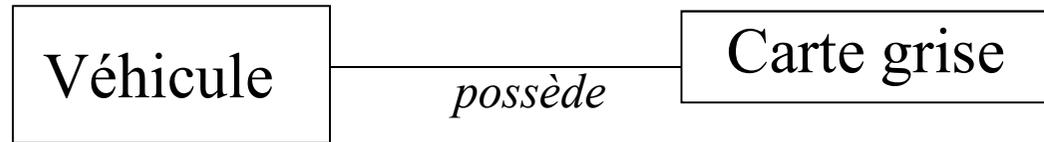
1 = 1..1

\* = 0..\*

q1 = q1..q1



Exemple :



# Exemple

- Des étudiants ayant un nom et un prénom sont inscrits dans un seul diplôme. Un diplôme a un nom et une date de création.
- Les étudiants suivent des cours. Un cours a un intitulé et un coefficient. Les étudiants ont une seule note pour chaque cours.

## Exemple : Gestion de bons de commandes

<b>N° bon</b> .....	<b>Date</b> .....			
<b>Nom client</b> .....				
<b>Adresse</b> .....				
.....				
<b>Nom représentant</b> .....				
<b>REF</b>	<b>DESIGN</b>	<b>QTE</b>	<b>PU</b>	<b>MONTANT</b>
.....	.....	.....	.....	.....
.....	.....	.....	.....	.....
<b>TOTAL</b>				.....

Un client peut passer une ou plusieurs commandes ou aucune commande.

Une commande est passée à un représentant qui n'est pas toujours le même pour un client donné.

## Constitution d'un dictionnaire de données

Nom	Signification	Type	Longueur	Nature	Règle de calcul
N°bon	N° bon de commande	texte	4		
date_commande	date de la commande	date	6		
code_client		texte	5		
nom_client		texte	30		
adresse		texte	60		
code_rep		texte	5		
nom_rep		texte	30		
ref		texte	5		
design		texte	30		
qte		N	3		
pu		N	7		
montant		N	8	calculé	pu * qte
total		N	8	calculé	somme des montants

# Questions

Réaliser le diagramme de classe

# **Gestion de l'emprunt et du retour d'un exemplaire dans une bibliothèque par un abonné.**

Un ouvrage est caractérisé par un numéro, un titre, un auteur, un éditeur et par un certain nombre de mots clés qui indiquent les sujets qui y sont traités. La bibliothèque possède un ou plusieurs exemplaires de chaque ouvrage. Un exemplaire est caractérisé par sa position dans les rayonnages et sa date d'achat. Un exemplaire peut être emprunté par un abonné. Les dates d'emprunt et de retour de l'exemplaire par l'abonné doivent être conservées. Un abonné a un nom, un prénom, une adresse et une date d'abonnement.

# Questions

Réaliser le diagramme de classe

# II - Base de données

## 1. Systèmes de Gestion de Bases de Données

Un SGBD est un ensemble de logiciels et de matériels associés permettant de gérer un ensemble de fichiers interdépendants et/ou de modéliser une partie du monde réel.

### 1.1. Fonctions d'un SGBD

Description

Utilisation

Intégrité

Confidentialité

Concurrence d'accès

Sécurité de fonctionnement

## **1.2. Les différents niveaux de représentation d'une base de données**

Niveau interne

Niveau conceptuel

Niveau externe

### **1.3. La mise en œuvre d'une SGBD**

Langage de définition de données

Langage de manipulation de données

## **1.4. Les différents modèles de données**

Modèle hiérarchique

Modèle en réseau

Modèle relationnel

Modèle objet

Modèle NoSQL

## 2. Le modèle relationnel

### 2.1. Présentation

Le modèle relationnel a été introduit par E.F. Codd à IBM San José en 1970.

- DOMAINE

**Domaine** : Ensemble de valeurs caractérisé par un nom.

Exemple :

Domaine D1 = ensemble des chaînes de caractères de longueur  $\leq 20$

Domaine D3 = ensemble des entiers

- **ATTRIBUT**

**Attribut** : Variable prenant ses valeurs dans un domaine.

Exemple :

L'attribut Nom prend ses valeurs dans D1.

L'attribut Num prend ses valeurs dans D3.

- **RELATION**

**Relation** : Sous ensemble du produit cartésien d'une liste de domaines. Soit les attributs  $A_1, A_2, \dots, A_n$  de domaines  $D_1, D_2, \dots, D_n$ . La relation  $r$  est l'ensemble des  $n$ -uplets  $(x_1, x_2, \dots, x_n)$  avec  $x_i$  élément de  $D_i$  ( $1 \leq i \leq n$ ).

Exemple :

Une relation  $r$  sur les attributs Nom, Adr, Num est décrite par un ensemble de triplets.

$r = \{(Dupond, Paris, 2140), (Durand, Orsay, 1128), (Dubois, Orsay, 3213)\}$

Exemple :

<b>Nom</b>	<b>Adr</b>	<b>Num</b>
Dupond	Paris	2140
Durand	Orsay	1128
Dubois	Orsay	3213

- SCHEMA DE RELATION

**Schéma de relation :** Liste des attributs de la relation accompagnés de leurs domaines.

Deux opérations peuvent être appliquées à la relation :

- **Projection** : La projection d'une relation  $r$  de schéma  $R$  sur un sous-ensemble d'attributs  $S$  de  $R$ , notée  $Pr_S(r)$  est la relation de schéma  $S$  obtenue par conservation des colonnes de  $r$  correspondant aux attributs de  $S$  et élimination des  $n$ -uplets dupliqués.

Exemple :

Projection de la relation  $r$  sur les attributs Nom et Adr.

$$r1 = Pr_{\text{Nom, Adr}}(r) \implies$$

<b>Nom</b>	<b>Adr</b>
Dupond	Paris
Durand	Orsay
Dubois	Orsay

$$r2 = Pr_{\text{Adr}}(r) \implies$$

<b>Adr</b>
Paris
Orsay

• **Jointure** : La jointure de deux relations  $r$  et  $s$  de schéma respectifs  $R$  et  $S$  qui ont en commun des attributs est une relation notée  $r \text{ Jo } s$ , qui a pour schéma la réunion de l'ensemble des attributs de  $R$  et de  $S$ . Les  $n$ -uplets de  $r \text{ Jo } s$  sont obtenus en concaténant les  $n$ -uplets de  $r$  avec les  $n$ -uplets de  $s$  qui ont la même valeur sur les attributs de même nom.

Exemple

$r =$

Nom	Adr	Num
Dupond	Paris	2140
Durand	Orsay	1128
Dubois	Orsay	3213

$s =$

Num	Code
2140	A1
3213	A2
2555	A2

r Jo s =

Nom	Adr	Num	Code
Dupond	Paris	2140	A1
Dubois	Orsay	3213	A2

- CLE D'UN SCHEMA DE RELATION

**Clé d'un schéma de relation** : La clé d'un schéma de relation appelée aussi **clé de la relation** est un sous ensemble X des attributs d'un schéma de relation R (A1, A2, ..., An) tel que :

-X  $\rightarrow$  A1, A2, ..., An

- il n'existe pas de sous ensemble Y inclus dans X tel que Y  $\rightarrow$  A1, A2, ..., An

Exemple :

R = (Nv, Type, Marque, puissance, Couleur)

Nv  $\rightarrow$  Type, Marque, puissance, Couleur

- FORMES NORMALES

**Première forme normale** : Un schéma de relation est en première forme normale notée FN1 si tout attribut contient une valeur atomique.

Exemple :

Personne (Nom, Prénoms)

sera modifiée en :

Personne (Nom, Prénom1, Prénom2)

OU

Personne1 (Nom, Prénom1)

Personne2 (Nom, Prénom2)

## Deuxième forme normale :

Un schéma de relation est en deuxième forme normale notée FN2 si:

- il est en première forme normale,
- tout attribut n'appartenant pas à la clé ne dépend pas que d'une partie de cette clé

### Exemple :

Fournisseur (Nom, Adresse, Article, Prix)

sera décomposée en deux :

Fournisseur (Nom, Adresse)

ET

Produit(Nom, Article, Prix)

### Troisième forme normale :

Un schéma de relation est en troisième forme normale notée FN3 si:

- il est en deuxième forme normale,
- tout attribut n'appartenant pas à une clé ne dépend pas d'un attribut non clé.

#### Exemple :

$R = (\underline{Nv}, \text{Type}, \text{Marque}, \text{Puissance}, \text{Couleur})$

sera décomposée en deux :

$R1 = (\underline{Nv}, \text{Type}, \text{Couleur})$

$R2 = (\underline{\text{Type}}, \text{Marque}, \text{Puissance})$

# Transformation d'un diagramme de classe en modèle relationnel

**Règle 1** : A chaque classe est associée un schéma de relation composé de l'ensemble des attributs de la classe.

Exemple :

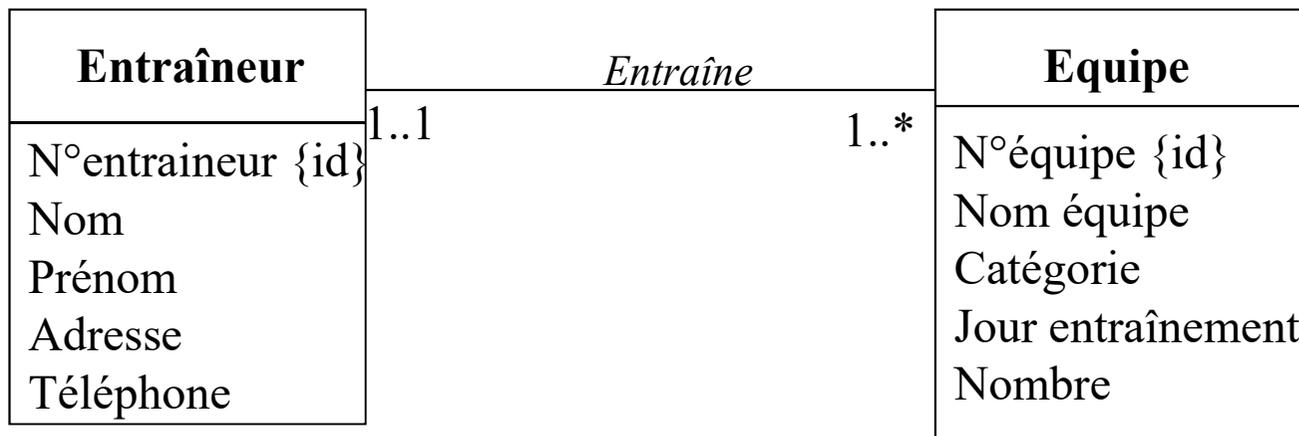
<b>Etudiant</b>
N°étudiant {id}
Nom
Prénom
Date naissance
Adresse

Etudiant (N°étudiant, Nom, Prénom, Date naissance, Adresse)

**Règle 2 :** Si dans une association A, il existe une classe E pour laquelle la cardinalité du couple (E, A) est égale à (0,1) ou (1,1), on ajoute dans le schéma de relation R issu de E une clé de chacune des autres classes participant à l'association ainsi que l'ensemble des attributs de l'association.

Exemple :

Dans un club de sport, un entraîneur suit plusieurs équipes.  
Chaque équipe est entraînée par un seul entraîneur.

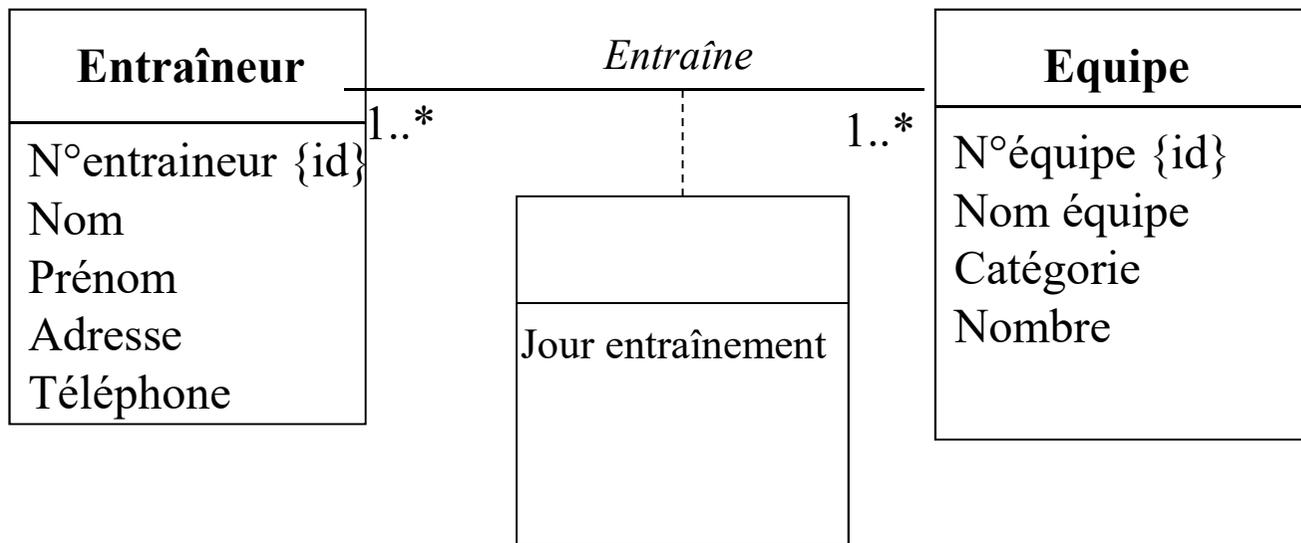


Entraîneur (N°entraîneur, nom, prénom, adresse, téléphone)

Equipe (N°équipe, nom\_équipe, catégorie, jour entraînement, nombre, #N°entraîneur)

**Règle 3 :** Si dans une association A il n'existe pas de classe E pour laquelle la cardinalité du couple (E, A) est égale à (0,1) ou (1,1), on crée un schéma de relation complémentaire composé de l'ensemble des clés des différentes classes accompagné des attributs de l'association.

Exemple : Supposons maintenant que dans l'association sportive de l'exemple ci-dessus, une équipe soit entraînée plusieurs fois par semaine par des entraîneurs différents.



Entraîneur (N°entraîneur, nom, prénom, adresse, téléphone)

Equipe (N°équipe, nom\_équipe, catégorie, nombre)

Entraîne (#N°équipe, #N°entraîneur, jour entraînement)

## Questions

Traduire les diagrammes de classe :

- De la gestion de bons de commande
- De l'emprunt et du retour d'un ouvrage

en modèle relationnel (Application des trois règles).

# Consultations médicales et prescriptions

Un patient a un n°ss, un nom, un prénom, une date de naissance.

Un médecin a un nom, un prénom et un numéro.

Une consultation est réalisée à une date et une heure précise pour un patient par un médecin et elle a un motif. Elle donne lieu à zéro ou une prescription de médicaments. Pour chaque médicament contenu dans la prescription la posologie est à noter.

Une prescription à une date et une durée.

Un médicament a un numéro, une désignation, un prix<sup>3</sup>.

## **Questions**

Réaliser le diagramme de classe

Traduire le diagramme de classe en modèle relationnel (Application des trois règles).

# III - Langage de manipulation de données SQL

SQL est un langage normalisé, commercialisé par IBM en 1981.

Il comporte quatre verbes de manipulation des données :

SELECT : recherche

INSERT : création

UPDATE : mise à jour

DELETE : suppression

## 5.1. Recherche

La syntaxe générale d'une instruction SELECT est la suivante:

SELECT	nom(s) des colonne(s) désirée(s) en résultat (OU * si toutes les colonnes)
FROM	nom(s) des relation(s) concernée(s)
WHERE	condition de recherche

ORDER BY	critère d'ordonnancement du résultat
----------	--------------------------------------

GROUP BY	colonne(s) de regroupement
----------	----------------------------

HAVING	condition de recherche sur le regroupement
--------	--

### *Recherche simple avec conditions*

```
SELECT nom_col1 [,nom_col2, ...]  
FROM nom_table1 [, nom_table2, ...]  
WHERE condition ;
```

Pour éviter les redondances : DISTINCT

Si un même nom de colonne apparaît dans deux tables distinctes, il est nécessaire de le faire précéder du nom de la table : nom\_table1.nom\_col1.

1) Quels sont les noms des médecins dont le prénom est Guillaume ?

2) Quels sont les médecins dont le prénom est Guillaume ?

3) Quels sont les numéros des médecins qui ont réalisé une consultation ?  
Quels sont les numéros et les noms des médecins qui ont réalisé une consultation ?

4) Quels sont les numéros et les noms des patients qui ont effectués une consultation et le numéro des consultations ?

L'opérateur IN teste l'appartenance d'une valeur à un ensemble de valeurs :

```
SELECT ...  
FROM ...  
WHERE nom_col IN (val1, val2, ...);
```

5) Quels sont les noms des médecins qui ont comme prénom : Guillaume, Lilia ou Louis ?

Les opérateurs logiques NOT, AND et OR.

L'opérateur LIKE

Les opérateurs utilisant une date : YEAR(date1), MONTH (date1), DAY (date1).

Les dates sont entourées de # afin de conserver le format date, et au format aaaa/mois/jour

6) Quels sont les médecins dont le nom commence par la lettre D ?

7) Quels sont les patients qui ont leurs anniversaires aujourd'hui ?

## *Imbrication de requêtes*

Le résultat de la requête imbriquée servira de condition de recherche à la requête de niveau supérieur. Plusieurs imbrications sont possibles.

*Sous requête utilisant l'opérateur IN ou NOT IN*

```
SELECT ...  
FROM ...  
WHERE nom_col IN (SELECT ...);
```

8) Quels sont les motifs des consultations des patients nés le 01/01/1920 ?

9) Quels sont les médicaments qui n'ont jamais été prescrit ?

*Sous requête utilisant les opérateurs de comparaison classiques*

```
SELECT ...  
FROM ...  
WHERE nom_col opérateur_comparaison (SELECT ...);
```

*Utilisation des fonctions standards de calcul*

Les fonctions suivantes sont à ajouter dans la clause SELECT : AVG, SUM, COUNT, MAX, MIN

*Utilisation de démonstratifs*

```
SELECT ...  
FROM nom_table démonstratif1, nom_table démonstratif2  
WHERE ...;
```

10) Quels sont les médicaments qui le même prix que le médicament de numéro «123 » ?

11) Quels sont les médicaments qui ont un prix supérieur au prix moyen ?

12) Quel est le nombre de prescriptions faites en 2019 ?

## *Tris*

```
SELECT ...  
FROM ...  
ORDER BY nom_col [ASC ou DESC], ... ;
```

## *Regroupements*

```
SELECT ...  
FROM ...  
GROUP BY nom_col ;
```

Il est possible d'imposer une condition aux groupes formés par la clause GROUP BY.  
Le critère de sélection d'un sous ensemble est introduit par la clause HAVING.

Quand une requête combine les clauses WHERE et HAVING, la clause WHERE est d'abord appliquée.

13) Afficher les numéros de patient et les numéros des consultations qu'ils ont effectuées (trier sur le numéro du patient).

14) Afficher les numéros de patient et les numéros des consultations qu'ils ont effectuées (trier en premier sur le numéro du patient et en deuxième sur le numéro de consultation).

15) Restituer pour chaque médecin, le nombre de consultations qu'il a effectué.

16) Restituer pour chaque médecin, le nombre de consultations qu'il a effectué seulement si leur nombre est supérieur à 5.

## 5.2. Manipulation

### Insertion

#### Exemple :

-Valorisation d'un tuple d'une relation. Création des caractéristiques d'un médecin.

```
INSERT INTO Medecin(n°medecin, nom, prenom, adresse) :  
<'221', 'azerty', 'jacques', '22 rue du lac 69008 Lyon'>
```

On peut insérer le résultat d'une sélection dans une relation.

#### Exemple :

- Soit la relation R, on peut valoriser cette relation avec les patients qui sont nés avant 1980

#### \* Création

```
SELECT n°ss, nom, prenom INTO R  
FROM patient  
WHERE year(datenaissance) <1980;
```

#### \* Ajout

```
INSERT INTO R  
SELECT n°ss, nom, prenom  
FROM patient  
WHERE year(datenaissance) <1980;
```

## **Suppression**

### Exemple :

-Effacer le contenu de la prescription n°34.

```
DELETE *  
FROM Contient  
WHERE #n°prescription = '34';
```

## **Mise à jour**

Pour une mise à jour, les nouvelles valeurs font intervenir la clause SET.

### Exemple :

- Augmenter de 5% les prix des médicaments.

```
UPDATE Medicament  
SET prix = prix * 1.05;
```

## **Gestion de formations dans un institut**

Chaque session est suivie par un certain nombre de participants. Une session est caractérisée par une date de début et un prix. Un cours peut faire l'objet dans l'année de plusieurs sessions identiques. Un cours a un libellé et un nombre d'heures. Une session peut être assurée par plusieurs animateurs et est placée sous la responsabilité d'un animateur principal. Un animateur peut intervenir dans plusieurs sessions. Le nombre d'heures effectué par chaque animateur dans une session doit être mémorisé.

- Établir un schéma conceptuel des données à l'aide d'un diagramme de classes UML
- Traduire le diagramme de classes en modèle relationnel (Application des trois règles).