

# Introduction au modèle relationnel

# Table des matières

<b>I - Contexte</b>	<b>3</b>
<b>II - Définition du modèle relationnel</b>	<b>4</b>
<b>III - Exercice</b>	<b>6</b>
<b>IV - Domaine</b>	<b>7</b>
<b>V - Exercice</b>	<b>8</b>
<b>VI - Relation</b>	<b>9</b>
<b>VII - Exercice</b>	<b>11</b>
<b>VIII - Clé</b>	<b>12</b>
<b>IX - Exercice</b>	<b>14</b>
<b>X - Clé primaire, clés candidates, clé artificielle</b>	<b>15</b>
<b>XI - Exercice</b>	<b>17</b>
<b>XII - Clés étrangères</b>	<b>18</b>
<b>XIII - Exercice</b>	<b>20</b>
<b>XIV - Schéma relationnel</b>	<b>21</b>
<b>XV - Exercice</b>	<b>24</b>
<b>XVI - Essentiel</b>	<b>25</b>
<b>XVII - Quiz</b>	<b>26</b>
<b>Glossaire</b>	<b>28</b>
<b>Abréviations</b>	<b>29</b>
<b>Bibliographie</b>	<b>30</b>
<b>Index</b>	<b>31</b>
<b>Crédits des ressources</b>	<b>32</b>

# Contexte



**Durée :** 3h

**Environnement de travail :** Aucun

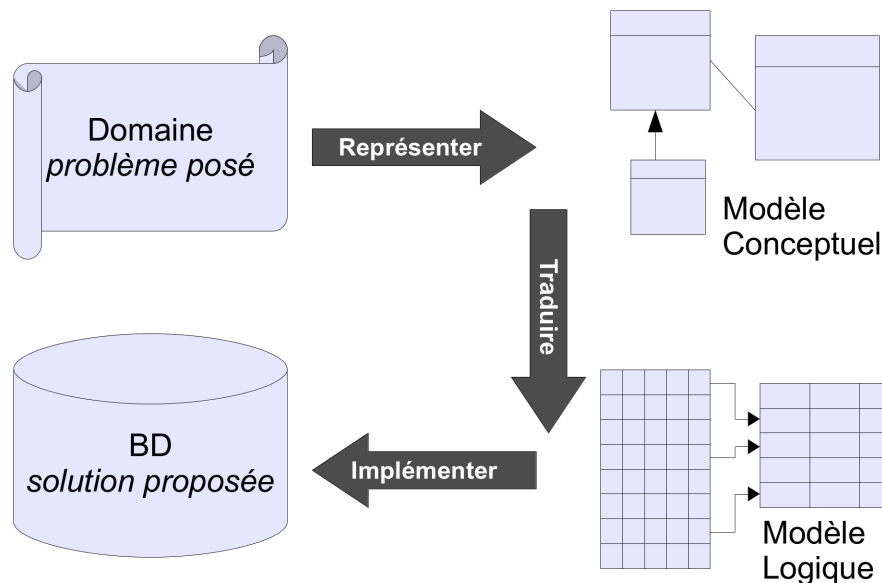
**Pré-requis :** Aucun

[cf. OudkOMr]

Le niveau logique est le lien entre le niveau conceptuel et l'implémentation effective de l'application. Le modèle conceptuel étant un modèle formel, le modèle logique a pour vocation d'être également un modèle formel, mais spécifiant non plus la réalité existante ou recherchée comme le modèle conceptuel, mais les données telles qu'elles vont exister dans l'application informatique.

Pour assumer cette fonction, le modèle relationnel s'est imposé en réaction aux insuffisances des modèles antérieurs, les modèles hiérarchique et réseau, et de part la puissance de ses fondements mathématiques.

Encore aujourd'hui dominant le modèle relationnel est un fondement indispensable à la conception de bases de données.



*Processus de conception d'une base de données*

Le modèle relationnel est aux fondements des SGBDR\*. Il a été, et continue d'être, le modèle théorique dominant pour la représentation logique des bases de données. Le modèle relationnel permet de reformuler le modèle conceptuel dans un formalisme - le tableau - beaucoup plus proche de l'implémentation informatique, bien qu'encore indépendant d'une solution technologique particulière.

# Définition du modèle relationnel

---



[cf. B0cmdPDe]

## Objectif

- Comprendre les objectifs et la logique du modèle relationnel.

## Mise en situation

Comment décrire aux systèmes de gestion de base de données la structure des données ?

Si la réponse immédiate est « *grâce au langage SQL* », il ne faut pas oublier que SQL doit bien se baser sur une certaine logique.

Et en effet, cela fait des dizaines d'années que les SGBD relationnels utilisent une structure de tableau pour stocker les données. Ce n'est pas un hasard, car ils se basent tous sur le modèle relationnel, qui indique comment formaliser des données et leurs interactions sous forme de tableau.

Dans ce module, vous découvrirez la philosophie du modèle relationnel. Le modèle relationnel est indépendant de tout SGBD. Il est fondamental car, bien utilisé, il permet d'assurer la cohérence des données.

## Introduction

Le modèle relationnel a été introduit par *Codd\**, en 1970 au laboratoire de recherche d'IBM de San José. Il s'agit d'un modèle simple et puissant à la base de la majorité des bases de données, encore aujourd'hui.

Les objectifs du modèle relationnel, formulés par Codd, sont les suivants :

- Assurer l'indépendance des applications et de la représentation interne des données.
- Gérer les problèmes de cohérence et de redondance des données.
- Utiliser des langages de données basés sur des théories solides.

## Modèle relationnel



### Définition

On appelle modèle relationnel un ensemble de concepts permettant de formaliser logiquement la description d'articles de fichiers plats, indépendamment de la façon dont ils sont physiquement stockés dans une mémoire numérique.

Le modèle relationnel inclut des concepts pour la **description** de données, ainsi que des concepts pour la **manipulation** de données.

## Représenter le monde en tables



Le modèle relationnel permet de représenter les données que l'on va gérer à l'aide d'un très petit nombre de concepts très simples :

- Les relations ou tables : des lignes et des colonnes.
- Les domaines de valeurs : chaque case d'une table prend une unique valeur dans un domaine pré-défini.
- Les clés : il existe des cases dont les valeurs doivent être uniques et non nulles.
- Les clés étrangères : il existe des cases qui doivent prendre une valeur existante dans les cases d'une autre table.

[cf. iikerCSq]

# Exercice

---



Enquête Wikipédia : pour réaliser cet exercice, on consultera l'article Wikipédia sur des bases de données<sup>1</sup>.

## Exercice

---

Après avoir parcouru la partie Terminologie, parmi les propositions ci-dessous, quels sont les différents types de modèle de données présentés ?

- modèle de données relationnel
- Modèle de données entité-association
- Modèle de données hiérarchique
- Modèle de données réseau

## Exercice

---

Dans la section « Histoire », on en apprend plus sur les travaux de Edgar F. Codd.

Quels sont les principaux concepts abordés dans sa thèse ?

- Algèbre relationnelle
- Système d'exploitation
- SQL
- Produit cartésien
- Programmation orientée objet
- Relation

## Exercice

---

À quel modèle de données se rattache les travaux de E. F. Codd ?

- Modèle de données entité-association
- Modèle de données hiérarchique
- Modèle de données relationnel
- Modèle de données réseau

---

<sup>1</sup>[https://fr.wikipedia.org/wiki/Base\\_de\\_donn%C3%A9es#Histoire](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es#Histoire)

# Domaine



[cf. PwZexDdV]

## Objectif

- Savoir décrire un ensemble de valeurs possibles grâce aux domaines.

## Mise en situation

Imaginez que vous souhaitiez modéliser une gamme d'ordinateurs portables, qui ont chacun une couleur, une masse, une marque, etc.

Les valeurs que peuvent prendre ces différents éléments ne sont pas libres : par exemple, les ordinateurs de notre gamme ne peuvent être que gris anthracite, noirs, ou blancs. Aussi, la masse ne peut pas être négative, et la marque ne peut pas dépasser trente caractères.

Comment formaliser ces contraintes sur les valeurs possibles de ces caractéristiques ? C'est justement l'objectif des domaines, que vous découvrirez dans ce module.

## Domaine



Ensemble, caractérisé par un nom, dans lequel des données peuvent prendre leurs valeurs.



Un domaine peut-être défini en *intension*\* (c'est à dire en définissant les propriétés caractéristiques des valeurs du domaine, on parle aussi de compréhension) ou en *extension*\* (c'est à dire en énumérant toutes les valeurs du domaine)

## Domaines définis en intension



- Tous les entiers.
- Les réels inférieurs à 5.
- Toutes les chaînes de 1 à 255 caractères.
- Les valeurs monétaires, définies comme des décimaux avec deux chiffres après la virgule.
- Les dates, définies comme des chaînes de 10 caractères comprenant des chiffres et des tirets selon le patron : « 00-00-0000 ».
- Les salaires, définis comme des valeurs monétaires compris entre 15.000 et 100.000.

## Domaines définis en extension



- Couleur : {Bleu, Vert, Rouge, Jaune, Blanc, Noir}
- SGBD : {Hiérarchique, Réseau, Relationnel, Objet, Relationnel-Objet}

[cf. zN206vOH]

# Exercice

---



Indiquer quelle définition et quel exemple correspondent respectivement aux mots **intension** et **extension**.

Le domaine des couleurs

Énonciation exhaustive de l'ensemble des objets du domaine

{bleu, rouge, vert}

Explicitation d'un domaine par la description de ses caractéristiques (en vue de sa compréhension abstraite, générale)

Intension	Extension



# Relation



[cf. 2DJdGD2M]

## Objectif

- Savoir regrouper des attributs au sein d'une relation.

## Mise en situation

Imaginez que vous souhaitez modéliser des livres. Chaque livre a un nom, un auteur, un nombre de pages, etc. Et chacune de ses caractéristiques prend ses valeurs dans un domaine, comme le domaine des nombres entiers positifs ou le domaine des prénoms.

Mais individuellement, chaque caractéristique ne décrit qu'une petite partie du livre. Comment regrouper ces caractéristiques dans une seule structure, pour former un tout ?

C'est tout l'objet des relations, que vous allez apprendre à construire dans ce module.

## Relation



Une relation  $R$  sur les domaines  $D_1, \dots, D_n$  est une table comportant une colonne pour chaque domaine et une ligne pour chaque tuple de la relation.

$D_1$	...	$D_n$
$V_1$	...	$V_n$
...	...	...
$V_1$	...	$V_n$

*Relation  $R$*

**Synonymes** : Table, tableau

## Attribut



On appelle attribut d'une relation, une colonne de cette relation. Un attribut est caractérisé par un nom et un domaine dans lequel il prend ses valeurs.

**Synonymes** : Champs, Propriété, Colonne

## Enregistrement



On appelle enregistrement d'une relation, une ligne de cette relation. Un enregistrement prend une valeur pour chaque attribut de la relation.

**Synonymes** : Tuple, N-uplet, Vecteur, Ligne

A	B
1	1
1	2
2	2

Relation R

La relation R comporte les deux attributs A et B et les trois enregistrements  $\langle 1,1 \rangle$ ,  $\langle 1,2 \rangle$  et  $\langle 2,2 \rangle$ .

### Attribut, domaine, ordre



Un attribut se distingue d'un domaine car il peut ne comporter que certaines valeurs de ce domaine.

Les colonnes de la relation ne sont pas ordonnées et elles ne sont donc repérées que par le nom de l'attribut.

### Valeur nulle



Un enregistrement peut ne pas avoir de valeur pour certains attributs de la relation, parce que cette valeur est inconnue ou inapplicable, sa valeur est alors « *null* ».

[cf. WE1rEL3]

# Exercice

---



On se donne un extrait d'une relation d'une base de données de vols :

numéro	compagnie	avion	départ	arrivée	date
AF3245	Air France	747	Paris	Oulan Bator	01-08-2002
AF6767	Air France	A320	Paris	Toulouse	30-07-2002
KLM234	KML	727	Paris	Amsterdam	31-07-2002

*Relation Vol*

On a de plus des informations supplémentaires sur les vols :

- le vol de Air France du 01-08-2002 en provenance de Paris est annoncé à 12:03 ;
- le vol KLM234 est annoncée à 06:32 ;
- le vol en direction de Toulouse le 30 juillet 2002 est annoncé à 16:04.

## Question

Complétez la relation et indiquez les attributs associés à celle-ci.



[cf. 5sSYuzNV]

## Objectif

- Savoir identifier les enregistrements d'une relation de manière unique.

## Mise en situation

Tous les enregistrements d'une relation ne sont identifiés que par leur contenu : en d'autres termes, il n'y a pas de notion de numéro d'enregistrement ou d'ordre.

Supposez maintenant que vous deviez gérer une relation avec des personnes : nom, prénom, date de naissance, etc. Si on vous demande de pointer du doigt, dans cette relation, la personne du nom de Saskia Larsen, comment feriez-vous ?

Vous ne pouvez pas répondre « à la ligne 12 », puisque la notion de numéro de ligne n'existe pas. Et si jamais il y a deux Saskia Larsen, le nom et le prénom ne suffisent pas non plus : il y a ambiguïté. Et si jamais elles sont nées le même jour..., etc.

Pour résoudre ce genre de situations, on utilise des clés, et vous allez apprendre comment faire dans ce module.

## Clé



### Définition

Une clé est un groupe d'attributs minimum qui permet d'identifier de façon univoque un tuple dans une relation.



### Fondamental

Toute relation doit comporter au moins une clé, ce qui implique qu'une relation ne peut pas contenir deux tuples identiques.

## Attributs de clés unique et non null



### Attention

Afin d'être déterminants pour l'identification d'un enregistrement, tous les attributs d'une clé doivent être « *valués* », c'est-à-dire qu'aucun ne peut avoir de valeur *null*.

Dire qu'un groupe d'attributs est une clé implique qu'il est unique et non *null*.

## Numéro d'étudiant



### Exemple

- Le numéro d'étudiant d'une relation *Etudiant* est une bonne clé car il y aura systématiquement une valeur non nulle.
- Le groupe d'attributs (nom, prénom) d'une relation *Etudiant* est en général une mauvaise clé, car les homonymes existent.

## Détermination d'une clé

Définir un groupe d'attributs comme étant une clé nécessite une réflexion sémantique sur les données composant ces attributs, afin de s'assurer de leur unicité.



### Fondamental

La définition des clés est un acte de **modélisation**, elle ne renvoie pas donc pas à une vérité intangible, mais à la réalité telle qu'elle est représentée dans le modèle que l'on élabore.



### Exemple

L'attribut numéro de sécurité sociale d'une relation personne peut paraître une bonne clé a priori car son unicité est assurée. Mais tout le monde n'en dispose pas forcément (les enfants, des étrangers), donc ce n'est une clé que si l'on considère des personnes affiliées à la sécurité sociale.

[cf. dHbqeZfG]

# Exercice



Soit les données d'un aéroport. On sait que le numéro de vol identifie de manière unique un vol dans le monde.

De même, on sait qu'il n'y a qu'un seul avion d'une compagnie qui part dans un créneau horaire d'une heure chaque jour.

Une compagnie de vol dispose potentiellement de plusieurs avions du même modèle.

numéro	compagnie	avion	départ	arrivée	date	heure
AF3245	Air France	747	Paris	Oulan Bator	01-08-2002	12:03:00
AF6767	Air France	A320	Paris	Toulouse	30-07-2002	16:04:00
KLM234	KML	727	Paris	Amsterdam	31-07-2002	06:32:00

*Relation Vol*

Parmi les propositions suivantes, lesquelles sont des clés pour cette relation.

- numéro
- (compagnie,avion)
- (compagnie,date,heure)
- (compagnie,départ,arrivée,date)
- (numéro,compagnie,avion,départ,arrivée,date,heure)

# Clé primaire, clés candidates, clé artificielle



[cf. CGouvM8]

## Objectifs

- Connaître les différents types de clé ;
- Savoir choisir la clé à utiliser dans une relation.

## Mise en situation

Chaque enregistrement d'une relation doit être identifié de manière unique par une clé.

Supposez que vous gérez des citoyens qui possèdent tous un numéro de sécurité sociale unique, ainsi qu'un numéro de passeport unique. Comment choisir la clé à utiliser ?

Autre situation : supposez que vous gérez des morceaux de musique. Des artistes peuvent avoir le même nom, les albums aussi, les morceaux aussi, avec pourquoi pas la même durée... Quelle stratégie adopter pour identifier chaque morceau de manière unique ?

Dans ce module, vous allez découvrir les notions de clés primaire, candidate et artificielle pour répondre à ces questions.

## Clé primaire



Si plusieurs clés existent dans une relation, on en choisit une parmi celles-ci. Cette clé est appelée **clé primaire**.

La clé primaire est généralement choisie :

- de façon à être **immuable**\*, c'est à dire à ne jamais changer une fois « *valuée* » pour la première fois ;
- de façon à ce qu'elle soit la plus **simple**, c'est à dire portant sur le moins d'attributs et sur les attributs de domaine les plus basiques (entiers ou chaînes courtes typiquement).

## Clés candidates



On appelle **clés candidates** l'ensemble des clés d'une relation qui n'ont pas été choisies comme clé primaire (elles étaient candidates à cette fonction).



S'il est impossible de trouver une clé primaire, ou que les clés candidates ne conviennent pas (non immuables, trop complexes...), il est possible de faire appel à une **clé artificielle**.

### **Clé artificielle**



Une clé artificielle est un attribut supplémentaire ajouté au schéma de la relation, qui n'est lié à aucune signification, et qui sert uniquement à identifier de façon unique les enregistrements et/ou à simplifier les références de clés étrangères à l'intérieur de la base de données.

### **Clé signifiante**



Une clé est signifiante si elle n'est pas artificielle.

Synonyme : Clé naturelle

[cf. Id2savHA]



# Exercice

---



On se donne un extrait d'une relation d'une base de données d'une université reprenant les informations sur les étudiants.

prénom	nom	date_de_naissance	adresse_mail
Alice	Duprès	02/12/96	alice.dupres@univ.fr
Thomas	Foucault	08/03/97	thomas.foucault@univ.fr
Alexandre	Germain	10/26/97	alexandre.germain@univ.fr

Chaque étudiant est identifié de manière identique par un prénom, un nom et sa date de naissance. De même, chaque étudiant a une adresse mail qui lui est propre.

## Question 1

Quelles sont les différentes clés de cette table ?

## Question 2

Cette université organise maintenant un autre événement où sont conviés tous les étudiants. Elle produit la liste relativement simple suivante pour attester de la présence de tous les étudiants.

Prénom	Nom
Alice	Duprès
Thomas	Foucault
Alexandre	Germain

Peut-on être sûr d'identifier tous les étudiants ?

# Clés étrangères



[cf. dKhoLhc8]

## Objectifs

- Savoir représenter des liens entre des relations ;
- Comprendre le lien entre clé primaire et clé étrangère.

## Mise en situation

En relationnel, les seuls éléments que vous avez à votre disposition pour décrire vos données sont les relations.

Or, si vous modélisez des réserves naturelles, vous aurez probablement une relation décrivant les lieux de ces réserves, et une relation décrivant les animaux qui s'y trouvent.

On peut se représenter ces deux relations comme deux tableaux : mais comment faire le lien entre ces tableaux, par exemple pour indiquer qu'un animal est présent dans une réserve particulière ?

Comme il n'est pas possible de « tracer de trait » entre les deux tableaux, il faut trouver une autre solution, et c'est tout l'objet des clés étrangères, que vous allez découvrir dans ce module.

Le modèle relationnel a pour objectif la structuration de données selon des relations. L'enjeu est de parvenir à traduire un modèle conceptuel en modèle logique relationnel. Or, il n'y a pas de notion d'association en relationnel, donc il faudra pouvoir traduire les associations avec les concepts dont on dispose : relation, clé, clé étrangère.

Afin de représenter des références entre relations dans un modèle relationnel, la seule solution est de stocker l'information dans une relation, et donc que certains attributs d'une relation servent à pointer sur d'autres relations.



**Attention**

Il n'y a pas vraiment de référence ou de lien en relationnel, puisque nous ne disposons que de tables, de clés, de clés étrangère et de valeurs.

On va donc devoir se servir de ces outils pour matérialiser une notion de référence.

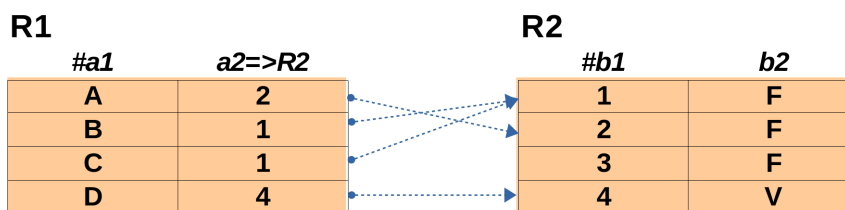
## Référence



**Méthode**

La référence entre deux tuples T1 et T2 de deux relations différentes est exprimable par une valeur identique entre une clé étrangère du tuple T1 et la clé primaire de l'autre tuple T2.

Synonyme : Lien

 Exemple

L'attribut  $a2$  de la relation  $R1$  référence l'attribut  $b1$  de la relation  $R2$  car  $a2$  est une clé étrangère de  $R1$  vers  $R2$  ( $b1$  est la clé primaire de  $R2$ ).

Ici on a donc par exemple les tuples identifiés par  $B$  et  $C$  de  $R1$  qui référencent le tuple identifié par  $1$  dans  $R2$ .

## Clé étrangère



Une clé étrangère est un attribut ou un groupe d'attributs d'une relation  $R1$ , devant apparaître comme clé primaire dans une relation  $R2$ , afin de matérialiser une référence entre les tuples de  $R1$  et les tuples de  $R2$ .

Une clé étrangère d'un tuple référence une clé primaire d'un autre tuple.



Seule une clé primaire peut être référencée par une clé étrangère.

**C'est même la seule fonction de la clé primaire : être la clé qui peut être référencée par les clés étrangères.**

## Contrainte d'intégrité référentielle



Une clé étrangère respecte la contrainte d'intégrité référentielle si sa valeur est effectivement existante dans la clé primaire d'un tuple de la relation référencée, ou si sa valeur est *null*.

Une clé étrangère qui ne respecte pas la contrainte d'intégrité référentielle exprime un lien vers un tuple qui n'existe pas et donc n'est pas cohérente.

[cf. Z9a2UNRH]

# Exercice



On dispose des deux relations suivantes concernant des avions et des compagnies aériennes.

De plus on sait que :

- l'avion immatriculé FR-1831 appartient à la compagnie AIR France KLM ;
- l'avion immatriculé N361AA appartient à la compagnie Fiji Airways ;
- l'avion immatriculé DQ-424 appartient à la compagnie American Airlines.

L'immatriculation des avions est unique.

<i>Avion</i>		
immatriculation	modele	heures_de_vol
FR-1831	A300	23719
N361AA	787	241
DQ-424	787	

<i>Compagnie Aérienne</i>		
denomination	OACI	siege_social
AIR France KLM	AFR	Paris
Fiji Airways	FJI	Nadi
Amerian Airlines	AAL	Fort Worth
IR Iran Air	IRA	Téhéran

## Question 1

Énoncer les clés de la relation Avion. Choisir une clé primaire.

## Question 2

Énoncer les clés de la relation Compagnie Aérienne. Choisir une clé primaire.

## Question 3

Ajouter la clé étrangère manquante dans la relation Avion et les valeurs associées.

# Schéma relationnel



[cf. mk2bpDXU]

## Objectifs

- Savoir décrire une relation grâce à un schéma relationnel ;
- Savoir représenter les différents types de clé.

## Mise en situation

Le modèle relationnel décrit, comme son nom l'indique, des relations : elles font intervenir plusieurs concepts, comme les attributs, les domaines, les clés primaires, les clés étrangères...

Afin de formaliser les relations, il faut bien écrire quelque part leur "fiche", qui décrit les différents attributs ainsi que leur nature.

On pourrait décrire les relations en langage naturel, mais il y aurait alors des risques d'ambiguïté.

Une solution, c'est d'utiliser un schéma relationnel, qui permette de décrire une relation sans ambiguïté et dans son intégralité : vous allez voir comment dans ce module.

## Schéma d'une relation



Le schéma d'une relation définit cette relation en intension. Il est composé :

- du nom de la relation,
- de la liste de ses attributs avec les domaines respectifs dans lesquels ils prennent leurs valeurs,
- de la clé primaire,
- des clés étrangères,
- des clés candidates.

## Schéma relationnel d'une base de donnée



Le schéma relationnel d'une BD est la définition en intension de cette BD (par opposition à l'instance de la BD qui est une extension de la BD). Il est composé de l'ensemble des schémas de chaque relation de la BD.

## Relation



`1 Relation (attribut1:domaine1, attribut2:domaine2, ... , attributN:domaineN)`

La relation "Relation" contient N attributs chacun défini sur son domaine.

**Clé primaire**

1 Relation (#attribut1:domaine1, ... , #attributM:domaineM, ... , attributN:domaineN)

La clé de la relation « *Relation* » est composée des attributs « *Attribut1* » à « *AttributM* » (attributs précédés du caractère #)

En général on note la clé primaire en premier dans la relation.

**Clé étrangère**

1 Relation1 (... , attributM=>Relation2, ... , attributN=>Relation2)

La relation « *Relation1* » comporte une clé étrangère (composée des attributs « *attributM* » à « *attributN* ») référençant la clé primaire de « *Relation2* ».

En général on note les clés étrangères en dernier dans la relation, sauf pour les clés étrangères qui font partie de la clé primaire (clés identifiantes).

- Bien sûr il peut exister plusieurs clés étrangères vers plusieurs relations distinctes.
- Une clé étrangère et sa clé primaire référencée sont toujours composées du même nombre d'attributs.
- Il n'est pas nécessaire de préciser les domaines des attributs appartenant à la clé étrangère car ce sont forcément les mêmes que ceux de la clé primaire référencée.
- Il n'est pas non plus en général nécessaire de préciser dans le schéma relationnel quels attributs de la clé étrangère référencent quels attributs de la clé primaire (cela est généralement évident) mais il est possible de la faire en notant « *Attribut=>Relation.Attribut* ».

**Clé candidates**

1 Relation1 (... , attributM:DomaineM, ...) avec attributM clé

Les clés candidates doivent être notées sur le schéma relationnel :

- S'il n'y a qu'une ou deux clés candidates, les noter directement après la définition de la relation.
- S'il y a beaucoup de clés, pour ne pas trop alourdir la notation, renvoyer à un tableau à part.

**Avion et Compagnie Aérienne**

<i>Avion</i>			
immatriculation	modele	heures_de_vol	OACI
FR-1831	A300	23719	AFR
N361AA	787	241	AAL
DQ-424	787		FJI

<i>Compagnie Aérienne</i>		
denomination	OACI	siege_social
AIR France KLM	AFR	Paris
Fiji Airways	FJI	Nadi
Amerian Airlines	AAL	Fort Worth
IR Iran Air	IRA	Téhéran

On peut exprimer en relationnel ces tables de cette façon :

- 1 Avion(#immatriculation:chaîne, modele:chaîne, heures\_de\_vol:entier, oaci=>CompagnieAerienne)
- 2
- 3 CompagnieAerienne(#oaci:chaîne, denomination:chaîne, siege\_social:chaîne) avec denomination clé

### Clés candidates et clé primaire



**Attention**

La notation  $R(\#a, \#b)$  signifie toujours que  $R$  a comme clé primaire  $(a, b)$ , et non que  $R$  aurait deux clés  $a$  et  $b$  (dont on ne saurait pas laquelle est primaire).

La notation  $R(\#a, b)$  avec  $b$  clé signifie bien que  $a$  et  $b$  sont deux clés de  $R$ , et que  $a$  est primaire.

Il ne faut pas confondre **une clé composée de deux attributs** avec **deux clés**.

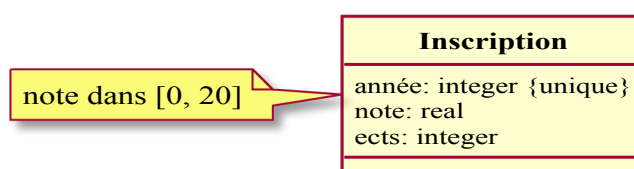
### Traduction des contraintes



**Complément**

Si le modèle conceptuel contient des contraintes spécifiques aux classes, alors on les liste après la relation associée.

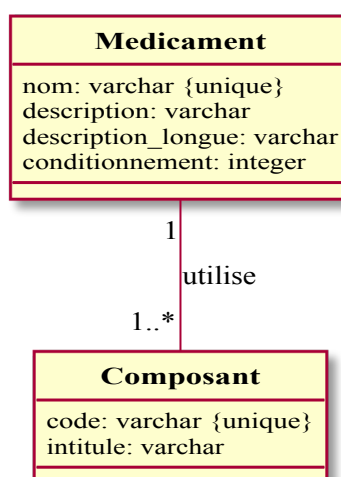
Par exemple, le modèle conceptuel suivant :



Se traduit comme :

- 1 Inscription(#annee:entier, note:réel, ects:entier) avec note dans [0, 20]

Cela vaut aussi pour certaines contraintes dues aux cardinalités des associations. Par exemple, le modèle conceptuel suivant :



Se traduit comme :

- 1 Medicament(#nom:varchar, description:varchar, description\_longue:varchar, conditionnement:entier) avec au moins un Composant référant
- 2
- 3 Composant(#code:varchar, intitule:varchar, medicament=>Medicament) avec medicament non null

[cf. Afwbkxv6]

# Exercice

---



Un laboratoire souhaite gérer les médicaments qu'il conçoit.

- Un médicament est décrit par un nom, qui permet de l'identifier. En effet il n'existe pas deux médicaments avec le même nom. Un médicament comporte une description courte en français, ainsi qu'une description longue en latin. On gère aussi le conditionnement du médicament, c'est à dire le nombre de pilules par boîte (qui est un nombre entier).
- À chaque médicament on associe une liste de contre-indications, généralement plusieurs, parfois aucune. Une contre-indication comporte un code unique qui l'identifie, ainsi qu'une description. Une contre-indication est toujours associée à un et un seul médicament.

## Exemple de données

Afin de matérialiser notre base de données, nous obtenons les descriptions suivantes :

- Le **Chourix** a pour description courte « *Médicament contre la chute des choux* » et pour description longue « *Vivamus fermentum semper porta. Nunc diam velit, adipiscing ut tristique vitae, sagittis vel odio. Maecenas convallis ullamcorper ultricies. Curabitur ornare.* ». Il est conditionné en boîte de 13.

Ses contre-indications sont :

- CI1 : Ne jamais prendre après minuit.
- CI2 : Ne jamais mettre en contact avec de l'eau.

- Le **Tropas** a pour description courte « *Médicament contre les dysfonctionnements intellectuels* » et pour description longue « *Suspendisse lectus leo, consectetur in tempor sit amet, placerat quis neque. Etiam luctus porttitor lorem, sed suscipit est rutrum non.* ». Il est conditionné en boîte de 42.

Ses contre-indications sont :

- CI3 : Garder à l'abri de la lumière du soleil.

## Question 1

Dessiner des relations instanciées (en extension donc) remplies avec les données fournies en exemple.

## Question 2

Écrire le schéma relationnel (définition en intension donc) permettant de représenter une base de données relationnelle pour le laboratoire.



# Essentiel

---



[cf. Gpztxlw8]

Les bases de données relationnelles ne savent gérer qu'une seule chose : des tableaux.

Et ça tombe bien, car le modèle relationnel permet de décrire des données et leurs interactions grâce à des relations, assimilables à des tableaux.

Avant de construire une base de données, on construit donc un modèle relationnel.

Chaque relation est constituée de différents attributs, qui puisent leurs valeurs dans un domaine, comme le nom d'une personne ou son âge.

Toutes les lignes d'une relation, ou enregistrements, doivent être identifiables de façon unique : c'est le rôle de la clé primaire, qui permet de lever toutes les ambiguïtés, par exemple entre deux personnes avec le même nom.

Bien entendu, des relations ne seraient rien sans interaction : une personne peut être amie avec une autre personne, ou louer un vélo. Les clés étrangères permettent de faire référence à des enregistrements d'une autre table, tout en conservant la structure de tableau.

Enfin, les schémas relationnels permettent de décrire les relations textuellement, sans ambiguïté et de manière synthétique.

# Quiz

---



## Exercice 1 : Quiz - Culture

### Exercice

---

Quels sont les concepts propres au relationnel ?

- Clé étrangère
- Classe
- Relation
- Association
- Clé primaire
- Requête

### Exercice

---

Pourquoi introduit-on le concept de clé primaire en relationnel ?

- On l'introduit pour simplifier l'usage des clés étrangères.
- On l'introduit pour pouvoir identifier chaque enregistrement de manière unique.

## Exercice 4 : Quiz - Méthode

### Exercice

---

En relationnel, on peut noter qu'un ensemble d'attributs est une clé en :

- Indiquant explicitement cela à côté de sa relation si c'est une clef candidate non primaire.
- Indiquant explicitement cela grâce à la clause `{unique}` pour les attributs concernés.
- Indiquant explicitement cela en préfixant les attributs de la clé du caractère # si c'est une clé primaire.

### Exercice

---

En relationnel, on indique les références :

- En ajoutant pour les champs de la clé étrangère ceux de la clé primaire avec le symbole `=>`.
- En écrivant explicitement ces références à côté de chaque relation.

## Exercice 7 : Quiz - Code

### Exercice

---

Quand et pourquoi introduire des clés artificielles dans les relations ?

- On introduit des clés artificielles s'il n'existe pas de clé naturelle.
- On introduit des clés artificielles systématiquement : cela est plus simple.

### Exercice

---

Pourquoi introduit-on le concept de clé étrangère ?

- Pour gérer le fait qu'une table a en général plusieurs clé candidates.
- Pour garantir la validité des références entre tables.

# Glossaire

---



## Extension

L'extension est l'explicitation d'un domaine par l'énonciation exhaustive de l'ensemble des objets du domaine.

Elle s'oppose à l'intension qui est une description abstraite des caractéristiques du domaine.

- Exemple : {bleu, rouge, vert}
- Contre-exemple : Le domaine des couleurs

## Immuable

Un attribut immuable est un attribut dont la valeur ne change plus jamais une fois fixée.

Synonymes : *frozen*, *immutable* en anglais (ou parfois en français comme anglicisme)

## Intension

L'intension est l'explicitation d'un domaine par la description de ses caractéristiques (en vue de sa compréhension abstraite, générale).

Elle s'oppose à l'extension qui est l'énonciation exhaustive de l'ensemble des objets du domaine.

- Exemple : Le domaine des couleurs
- Contre-exemple : {bleu, rouge, vert}

# Abréviations

---



**SGBDR** : Système de Gestion de Bases de Données Relationnelles

# Bibliographie

---



Codd EF, *A relational model for large shared data banks*, Communications de l'ACM, juin 1970.

# Index

---



Association .....	18
Attribut .....	12
Clé .....	12
Clé candidate.....	15
Clé primaire .....	15
Codd.....	4
Domaine .....	7, 9
Lien .....	18
Logique .....	4
Modèle .....	4, 21
Produit.....	9
Référence.....	18
Relation.....	9, 12, 18, 21
Relationnel.....	4, 9, 21
Relationnel-objet .....	4
Schéma .....	21

## Crédits des ressources

---



p. 23

*<http://creativecommons.org/licenses/by-sa/3.0/fr/>, Julien Jerphanion*

p. 23

*<http://creativecommons.org/licenses/by-sa/3.0/fr/>, Julien Jerphanion*