

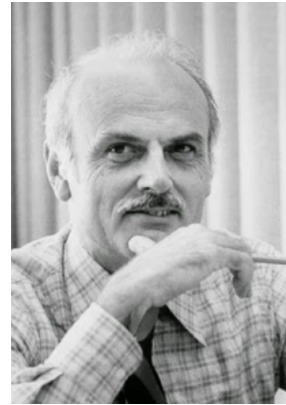
## 1.2

## Les bases de données relationnelles

NSI TERMINALE - JB DUTHOIT

### Histoire

Les bases de données relationnelles ont été mises au point en 1970 par Edgar Franck Codd, informaticien britannique (1923-2003). Ces bases de données sont basées sur la théorie mathématique des ensembles.



En savoir plus sur Edgar Franck Codd

La notion de relation est au coeur des bases de données relationnelles.

#### 1.2.1 Les relations

Une **relation** peut être vue comme un tableau à 2 dimensions, composé d'un en-tête (en rouge dans le tableau) et d'un corps (en bleu dans le tableau).

Le corps est lui-même composé de p-uplets (lignes) et d'attributs (colonnes).

L'en-tête contient les intitulés des **attributs**, le corps contient les données en elles-même.

num__ secu	nom__ auteur	prenom__ auteur	annee__ naiss	pays__ naiss
1730359006088	Beaudelaire	Charles	1821	France
1680359006089	Camus	Albert	1913	France
1560359006089	Hugo	Victor	1802	France
1450359011089	Flaubert	Gustave	1821	France
1030399006089	Rostand	Edmond	1868	France

#### Remarque

| On emploie aussi le terme "table" à la place de "relation".

La structure d'une relation peut se représenter sous cette forme :

AUTEUR
num__ secu
nom__ auteur
prenom__ auteur
annee__ naiss
pays__ naiss

### 1.2.2 Domaine

Pour chaque attribut d'une relation, il est nécessaire de définir un **domaine**.

Le domaine d'un attribut donné correspond à un ensemble fini ou infini de valeurs admissibles.

Par exemple, le domaine de l'attribut "num\_\_ secu" correspond à l'ensemble des entiers. Le domaine de l'attribut "nom\_\_ auteur" et "prenom\_\_ auteur" correspondent à l'ensemble des chaînes de caractères .

Le domaine de l'attribut "annee\_\_ naissance" correspond à l'ensemble des entiers positifs.

Afin d'être certain de ne pas avoir deux p-uplets identiques, on utilisera une clé primaire.

### 1.2.3 Clé primaire

#### Définition

Une **clef primaire** est un attribut dont la valeur permet d'identifier de manière **unique** un p-uplet de la relation.

Autrement dit, si un attribut est considéré comme clef primaire, on ne doit pas trouver dans toute la relation 2 fois la même valeur pour cet attribut.

#### Exercice 1.2

On considère les relations ci-dessous. Pour chaque relation, donner la liste des attributs et son domaine. Proposer ensuite une clé primaire pour chaque relation, en expliquant.

☛ Pour les domaines, on utilisera **TXT** pour les chaînes de caractères, **INT** pour les entiers, **REAL** pour les flottants, **DATE** pour les dates(jour/mois/année).

PAYS
nom__ pays
population
surface

AUTEUR
num__ secu
nom__ auteur
prenom__ auteur
annee__ naiss
pays__ naiss

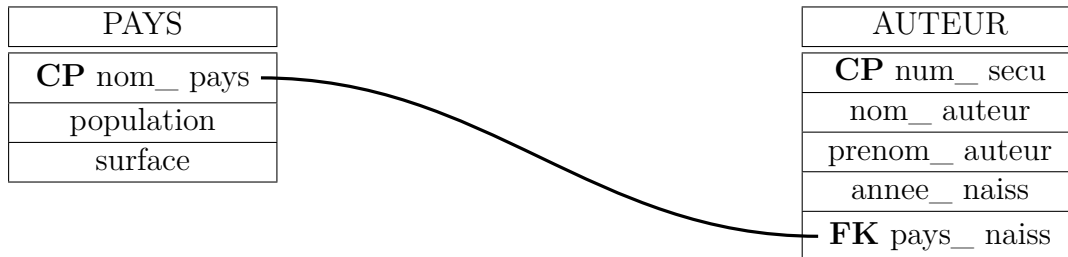
LIVRES
num__ isbn
titre
annee

### 1.2.4 Clé étrangère

#### Définition

les **clés étrangères**, qui permettent de gérer des relations entre plusieurs tables, et garantissent la cohérence des données.

#### Lien entre la relation PAYS et la relation AUTEUR



Dans l'exemple ci-dessus, l'attribut "pays\_\_naiss" de la relation AUTEUR permet bien d'établir un lien entre la relation AUTEUR et la relation PAYS.

"nom\_\_pays" correspond bien à la clef primaire de la relation PAYS, conclusion : "pays\_\_naiss" est une clef étrangère pour la relation AUTEUR.

On note aussi :

PAYS(nom\_\_pays, population, surface)

AUTEUR(num\_\_secu, # pays\_\_naiss, nom\_\_auteur, prenom\_\_auteur, annee\_\_naissance)

#### Remarque

- le (ou les ) attribut(s) soulignés sont les clés primaires.
- le # signifie que l'on est en présence d'une clé étrangère

#### Exercice 1.3

On souhaite modéliser un annuaire téléphonique simple dans lequel chaque personne (identifiée par son nom et son prénom) est associée à son numéro de téléphone.

On considère donc la relation suivante :

*ANNUAIRE*((nom TEXT, prenom TEXT), telephone TEXT)

⚠ On considère donc ici qu'il n'existe pas deux personnes ayant le même nom et le même prénom. ⚠ On pourra aussi s'inspirer de la question suivante, qui permettra de choisir judicieusement les attributs !

- Pourquoi avoir choisi TEXT comme domaine pour l'attribut telephone ?
- Dire si chacun des enregistrements suivants est valide pour la relation *Annuaire*. On considère que la table *Annuaire* est vide à chaque début d'enregistrement.
  - {}
  - {('titi', 'toto', '0123456789')}
  - {('titi', 'toto', '0123456789'), ('Doe', 'John', '0123456789')}
  - {('titi', 'toto', '0123456789'), ('titi', 'toto', '9876543210')}
  - {('titi', 'toto', '0123456789'), ('titi', 'tata', '9876543210')}
  - {('titi', 'toto', '0123456789'), ('Doe', 'John')}
  - {('titi', 'toto', '012345789')}

⚠ Expliquez!!!

3. Quelle solution peut-on apporter si l'on veut pouvoir considérer deux personnes ayant le même nom et le même prénom ? Donner la relation.

#### Exercice 1.4

- Donner la modélisation relationnelle d'un bulletin scolaire. Il faut créer 3 relations, et il conviendra aussi de s'aider de la question 2 :
  - La relations **ELEVE** des élèves, possédant un numéro d'étudiant alphanumérique unique
  - La relation **MATIERE** pour ajouter les différentes matières
  - La relation **NOTE** pour rentrer les notes, avec au plus une note sur 20 par élève et par matière. Chaque note sera identifiée par un numéro alphanumérique.
- Dire si chacun des ensembles ci-dessous est une relation valide pour le schéma de la base de données du bulletin de notes.
  - ELEVE** = {}
    - MATIERE** = {}
    - NOTE** = {}
  - ELEVE** = {( 'Titi', 'Toto', 'AB56789' )}
    - MATIERE** = {( 'NSI', 0), ( 'Sport', 1 )}
    - NOTE** = {( 'AB56789', 1, 17 )}
  - ELEVE** = {( 'Titi', 'Toto', 'AB56789' )}
    - MATIERE** = {( 'NSI', 0 )}
    - NOTE** = {( 'AB56789', 1, 17 )}
  - ELEVE** = {( 'Titi', 'Toto', 'AB56789' )}
    - MATIERE** = {( 'NSI', 0 )}
    - NOTE** = {( 'AB56789', 0, 17), ( 'AB56789', 0, 18 )}
  - ELEVE** = {( 'Titi', 'Toto', 'AB56789' )}
    - MATIERE** = {( 'NSI', 0), ( 'Sport', 1 )}
    - NOTE** = {( 'AB56789', 0, 17), ( 'AB56789', 1, 17 )}

#### Exercice 1.5

Un commerçant utilise plusieurs fichiers pour gérer ses produits. On considère un fichier destiné à gérer des produits frais.

Le tableau présenté est un extrait du contenu de ce fichier.

Les quatre colonnes contiennent respectivement un identifiant numérique, le nom d'un produit, son prix et la marque qui le commercialise.

Les mêmes noms de marques peuvent apparaître plusieurs fois dans la colonne marque mais aussi dans les fichiers correspondant à d'autres types de produits.

id	nom	prix	marque
17	Yaourt6	2.52	Yopnone
21	Yaourt12	4.93	Dalait
25	Beurre250	2.27	Croisement
28	Crème50	2.74	Dalait
31	Crème70	3.79	Yopnone

A partir de ce fichier construire :

- une relation **Frais**, (pour les produits frais),
- une relation **Marque**

un modèle relationnel permettant d'éviter la redondance d'informations. Indiquer une clé primaire pour chacune des deux tables et préciser un champ jouant le rôle d'une clé étrangère.  
\*\*\*

### Exercice 1.6

Un institut a constitué un tableau contenant des données statistiques sur une épidémie qui s'est répandue dans tous les pays.

Ce tableau est constitué de quatre colonnes représentant :

- le nom d'un pays,
- le numéro d'un jour (de 1 à 365),
- le nombre de cas confirmés,
- le nombre de décès.

Voici quatre lignes extraites du tableau :

pays	jour	cas	décès
France	83	1195	186
Allemagne	87	966	53
Suisse	95	228	17
France	108	2866	441

Expliquer quelle peut être une clé primaire.

### Exercice 1.7

Un particulier a un grand nombre de chansons stockées sur son ordinateur.

Il tient à jour un fichier qui contient toutes les chansons enregistrées par des groupes.

Ce fichier contient quatre colonnes où sont notés respectivement le titre de la chanson, le groupe qui l'a enregistrée, les membres du groupe et la date d'enregistrement.

Voici une ligne de ce fichier :

Titre	Groupe	Membres	Année
Roxane	The Police	Sting, Summers, Copeland	1978

Constituer un modèle relationnel normalisé l'aide de trois tables nommées :

- Chanson,
- Groupe,
- Artiste.

### Exercice 1.8

Modéliser des informations sur les départements français. Pour chaque département, on veut pouvoir stocker son nom, son code, son chef-lieu et la liste des départements voisins.

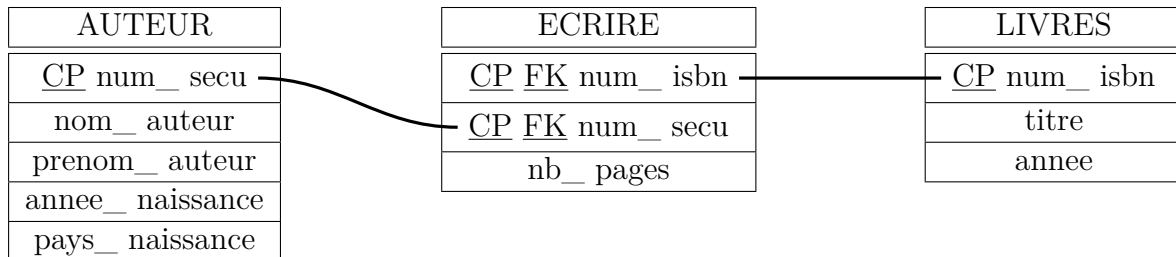
⚠ Ne pas oublier la corse :-)

Donner une contrainte utilisateur qui permettra d'éviter la redondance des informations dans la liste des voisins

## Lien entre AUTEUR et LIVRES

Ici, le lien est un peu plus difficile, car un auteur peut réaliser plusieurs livres et un livre peut être réalisé par plusieurs auteurs.

Il va donc être nécessaire de passer par une relation ECRIRE :



⚠ Dans ce cas, il y a donc deux clés étrangères. Et ce couple de clés étrangères forment la clé primaire.

### 1.2.5 Contraintes d'intégrité

Les **SGBD** modernes permettent de garder les données conformes au modèle relationnel. Les **contraintes d'intégrité** sont des règles que doivent respecter les champs de la base de données afin de **garantir** leur **cohérence**, leur **pertinence** et leur **validité**.

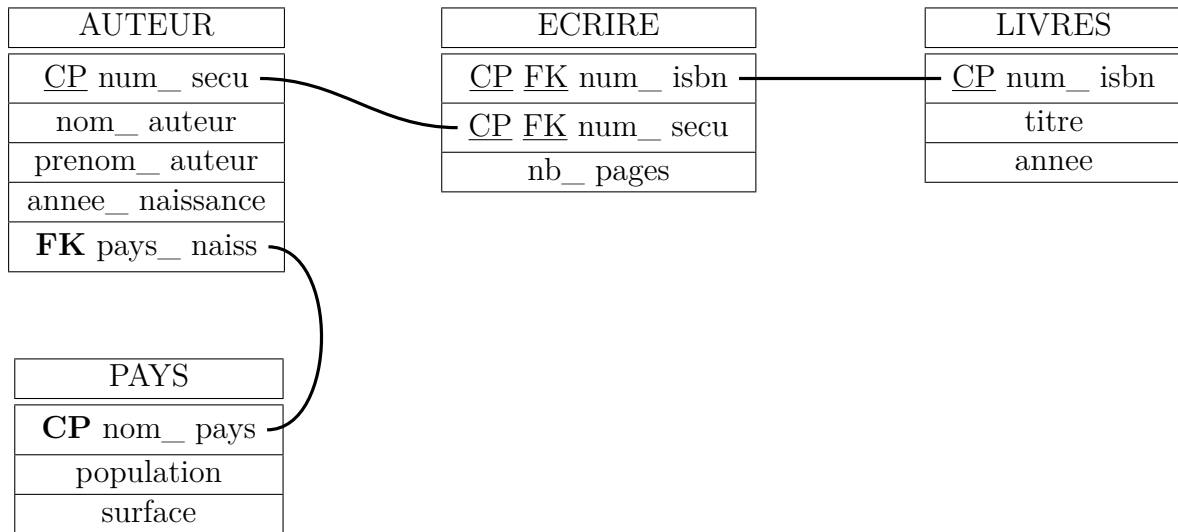
Trois types de contraintes d'intégrité font partie intégrante du modèle de données relationnel :

- L'**intégrité de clé primaire** ou **intégrité de relation** : cette règle stipule que chaque table (ou relation) doit avoir une clé primaire et que la ou les colonnes choisies pour être la clé primaire doivent être uniques et non nulles. Chaque tuple (ou enregistrement) est identifié par la clé primaire.
- L'**intégrité référentielle** est une situation dans laquelle pour chaque information d'une table A qui fait référence à une information d'une table B, l'information référencée existe dans la table B.
- L'**intégrité du domaine** spécifie que toutes les colonnes d'une base de données relationnelle doivent être déclarées sur un domaine défini. Chaque attribut doit donc prendre une valeur dans le domaine de valeur.

Une fois la contrainte déclarée, le SGBD refusera toute modification du contenu de la base de données qui violerait une de ces règles et casserait l'intégrité.

### 1.2.6 Schéma relationnel

on appelle schéma relationnel l'ensemble des relations présentes dans une base de données.



#### Exercice 1.9

On souhaite gérer des réservations dans une compagnie d'hôtels. Chaque hôtel possède un identifiant alphanumérique et chaque chambre de l'hôtel a un numéro de chambre qui tient compte de l'hôtel dans lequel est située la chambre. Par exemple, dans l'hôtel ayant pour identifiant **HOTEL1**, une chambre pourra avoir comme identifiant **HOTEL1\_023**.

On considère alors les relations suivantes :

RESERVATION
numero
numero__client
numero__chambre
date__resa

CHAMBRE
numero__chambre
id__hotel
prix

HOTEL
id__hotel
nom__hotel
adresse__hotel

CLIENT
numero__client
nom
prenom

1. Pour chaque relation, donner la liste des attributs et son domaine.
2. Ajoutez les associations et compléter avec les clés primaires et étrangères
3. A l'aide de ce modèle, répondez aux questions suivantes \*\*\* :
  - a) Peut-on avoir des clients homonymes ?
  - b) Un client peut-il réserver plusieurs chambre à une date donnée ?
  - c) Est-il possible de réserver une chambre sur plusieurs jours ?
  - d) Peut-on savoir si une chambre est libre a une date donnée ?
  - e) Peut-on réserver plusieurs fois une même chambre à une date donnée ?

#### Exercice 1.10

On reprend le schéma relationnel de l'exercice précédent.  
Compléter le tableau suivant :

Relation	Attribut	Type	Unicité	Domaine évent	Valeur nulle ?	Clé
CHAMBRE	id__hotel					
CHAMBRE	prix					
RESERVATION	date__resa					
CLIENT	numero__client					

Pour la colonne Type, on choisira parmi : entier, réel, texte, date  
 Pour les colonnes Unicité et Valeur nulle , on répondra par oui ou non.  
 Pour la colonne clé, on mettra PK (clé primaire) ou FK(clé étrangère), ou on laissera vide.  
 Pour la colonne "domaine évent", on précisera le domaine possible  
 \*\*\*

### Exercice 1.11

On souhaite gérer des visites dans un centre médical. On considère les relations suivantes :

CONSULTATION
numero
matricule
numero__ss
date__consultation

MEDICAMENT
code
libelle

MEDECIN
matricule
nom__medecin

PATIENT
numero__ss
nom__patient
prenom__patient

PRESCRIRE
numero
code
nb__prises

1. Pour chaque relation, donner la liste des attributs et son domaine.
2. Ajoutez les associations et compléter avec les clés primaires et étrangères
3. A l'aide de ce modèle, répondez aux questions suivantes \*\*\* :
  - a) Un patient peut-il effectuer plusieurs visites ?
  - b) Un médecin peut-il recevoir plusieurs patients dans une même consultation ?
  - c) Peut-on prescrire plusieurs médicaments dans la même consultation ?
  - d) Deux médecins différents peuvent-ils prescrire le même médicament ?

### Exercice 1.12

On donne ci-dessous les occurrences issues de la relation CONSULTATION , issue du schéma relationnel du second exercice.  
 Citez les anomalies ! \*\*\*

numero	matricule	numero__ss	date__consultation
1	123		21/11/2018
2	123	18252454564566	
2	526	'Aspirine'	13/03/2019