

## 1. Introduction

Le terme base de données est apparu au début des années 60. C'est l'apparition des disques durs à la fin des années 50 qui a permis d'utiliser les ordinateurs pour stocker et manipuler des données. Avec l'apparition du Web, la quantité de données à stocker a littéralement explosé. Aujourd'hui, la plupart des sites internet (du petit site personnel au grand site d'e-commerce) utilisent au moins une base de données. Les bases de données jouent un rôle fondamental dans notre monde devenu numérique où il est extrêmement facile de dupliquer l'information.

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.

## 2. Modèle relationnel

Le modèle de données relationnel représente une base de données comme un ensemble structuré d'informations sous forme de tables (tableau à deux dimensions) appelées **relations**.

Une relation est composée :

- d'un **en-tête** ;
- d'un **corps**.

Cette table permet de mettre en relation :

- des lignes (ou enregistrements) nommées **t-uplets** ou tuples ;
- des colonnes nommées **attributs** ou champs.

id	titre	auteur	ann_publi	note	en-tête
1	1984	Orwell	1949	10	un t-uplet
2	Dune	Herbert	1965	8	
3	Fondation	Azimov	1951	9	
4	Le meilleur des mondes	Huxley	1931	9	
5	Fahrenheit 451	Bradbury	1953	7	
6	Ubik	K.Dick	1969	6	
7	Chroniques martiennes	Bradbury	1950	7	
8	La nuit des temps	Barjavel	1968	7	
9	Blade Runner	K.Dick	1968	9	
10	Les Robots	Azimov	1950	10	
11	La Planète des singes	Boule	1963	9	
12	Ravage	Barjavel	1943	11	
13	Le Maître du Haut Château	K.Dick	1962	7	
14	La fin de l'éternité	Azimov	1955	9	

attribut : titre      corps

### 2.1. Domaine

Pour chaque attribut d'une relation, il est nécessaire de définir un **domaine**. Le **domaine** d'un attribut est un ensemble, fini ou infini, de valeurs que peut prendre une donnée (entiers, dates, chaînes de caractères...).

Le **domaine** d'un attribut est renseigné au moment de la création de la relation. Le Système de Gestion de Base de Données (SGBD) s'assure par la suite qu'un élément ajouté à une relation respecte bien le domaine de l'attribut correspondant.

Le domaine de l'attribut **titre** est une chaîne de caractères.

Le domaine de l'attribut **ann\_publi** est une année.

### 2.2. Clé primaire

Dans une base de données relationnelles, **une relation ne peut pas contenir deux t-uplets identiques**.

Afin de respecter cette contrainte, il faut une clé primaire.

La **clé primaire** est un **attribut dont la valeur permet d'identifier de manière unique un t-uplet de la relation**.

Dans la table ci-dessus, l'attribut **id** est la clé primaire.

Une clé primaire peut aussi être un couple de deux attributs (**titre, ann\_publi**).

Dans la table ci-dessus, l'attribut **titre** pourrait servir de clé primaire mais parfois plusieurs films portent le même nom, donc ce n'est pas judicieux.

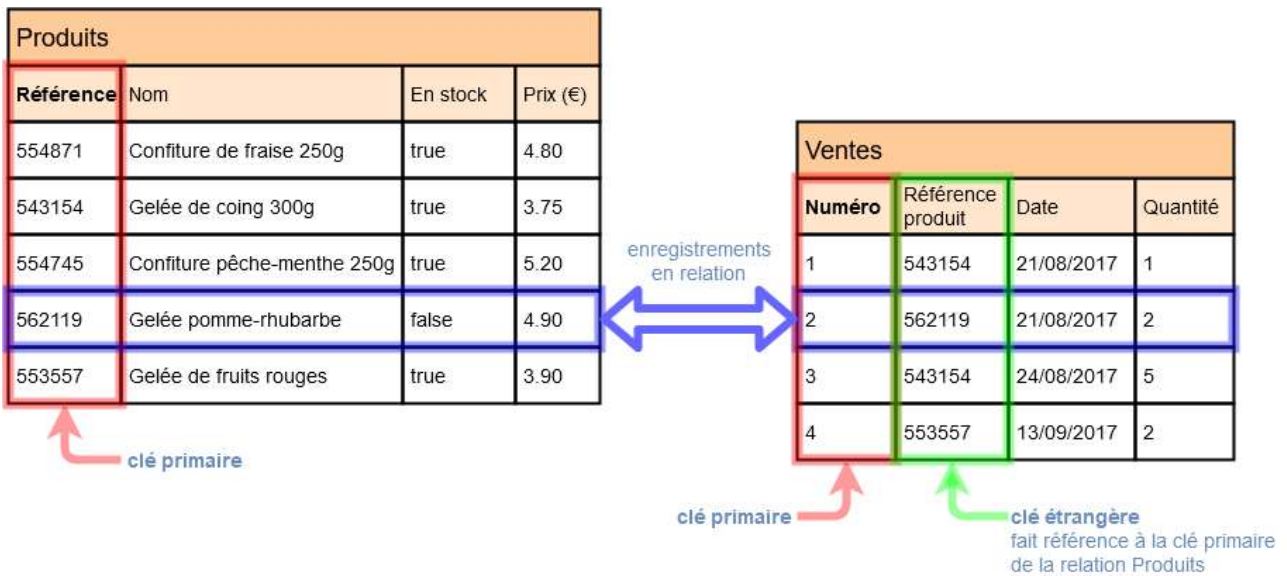
## 2.3. Relations et clés

Les données sont donc organisées en table (relation) dans des lignes (t-uplets) et des colonnes (attributs). Une base de données relationnelle est constituée de relations connectées permettant d'enrichir chaque table.

Par exemple les relations Produits et Ventes sont connectées puisque l'objet de la base est de gérer la vente de produits.

La relation entre les tables peut ensuite être définie à l'aide de clés étrangères. Une **clé étrangère** est un champ d'une table dont les valeurs sont des références à une clé primaire d'une autre table (il peut y en avoir plusieurs dans une même relation).

**Une clé étrangère est donc une clé primaire d'une autre table.**



La relation **Produits** indique que le pot de gelée pomme-rhubarbe coûte 4,90 € et qu'il n'y en a plus en stock. La relation **Ventes** nous informe en plus que pots ont été vendus le 21/08/2017.

## 2.4. Créer des relations

Dans la relation LIVRES\_AUTEURS ci-contre, il y a beaucoup d'informations dupliquées. Par exemple l'auteur K. Dick et ses caractéristiques sont présents trois fois. Cela n'est pas efficace car s'il faut apporter une modification sur ces données, il faudra la faire plusieurs fois.

Relation LIVRES\_AUTEURS

id	titre	nom_auteur	prenom_auteur	date_nai_auteur	langue_ecriture_auteur	ann_public	note
1	1984	Orwell	George	1903	anglais	1949	10
2	Dune	Herbert	Frank	1920	anglais	1965	8
3	Fondation	Asimov	Isaac	1920	anglais	1951	9
4	Le meilleur des mondes	Huxley	Aldous	1894	anglais	1931	7
5	Fahrenheit 451	Bradbury	Ray	1920	anglais	1953	7
6	Ubik	K.Dick	Philip	1928	anglais	1969	9
7	Chroniques martiennes	Bradbury	Ray	1920	anglais	1950	8
8	La nuit des temps	Barjavel	René	1911	français	1968	7
9	Blade Runner	K.Dick	Philip	1928	anglais	1968	8
10	Les Robots	Asimov	Isaac	1920	anglais	1950	9
11	La Planète des singes	Boulle	Pierre	1912	français	1963	8
12	Ravage	Barjavel	René	1911	français	1943	8
13	Le Maître du Haut Château	K.Dick	Philip	1928	anglais	1962	8
14	Le monde des Â	Van Vogt	Alfred Elton	1912	anglais	1945	7
15	La Fin de l'éternité	Asimov	Isaac	1920	anglais	1955	8
16	De la Terre à la Lune	Verne	Jules	1828	français	1865	10

La solution consiste à travailler avec deux relations au lieu d'une seule et créer un lien entre ces deux relations.

id	titre	id_auteur	ann_publi	note
1	1984	1	1949	10
2	Dune	2	1965	8
3	Fondation	3	1951	9
4	Le meilleur des mondes	4	1931	7
5	Fahrenheit 451	5	1953	7
6	Ubik	6	1969	9
7	Chroniques martiennes	5	1950	8
8	La nuit des temps	7	1968	7
9	Blade Runner	6	1968	8
10	Les Robots	3	1950	9
11	La Planète des singes	8	1963	8
12	Ravage	7	1943	8
13	Le Maître du Haut Château	6	1962	8
14	Le monde des Â	9	1945	7
15	La Fin de l'éternité	3	1955	8
16	De la Terre à la Lune	10	1865	10

id	nom	prenom	ann_naissance	langue_ecriture
1	Orwell	George	1903	anglais
2	Herbert	Frank	1920	anglais
3	Asimov	Isaac	1920	anglais
4	Huxley	Aldous	1894	anglais
5	Bradbury	Ray	1920	anglais
6	K.Dick	Philip	1928	anglais
7	Barjavel	René	1911	français
8	Boule	Pierre	1912	français
9	Van Vogt	Alfred Elton	1912	anglais
10	Verne	Jules	1828	français

L'attribut `id_auteur` de la relation LIVRES permet de créer un lien avec la relation AUTEURS.

## 2.5. Intégrité d'une base de données

Une **contrainte d'intégrité** est une règle qui définit la cohérence d'une donnée ou d'un ensemble de données d'une base de données.

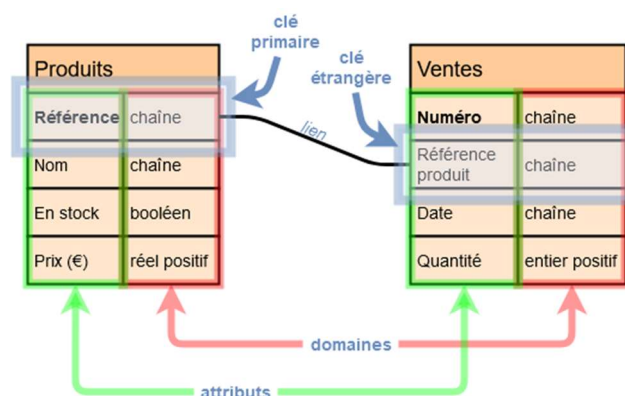
- Pour préserver l'intégrité d'une base de données, il est important de bien vérifier que toutes les valeurs de la clef étrangère correspondent bien à des valeurs présentes dans la clef primaire (nous aurions un problème d'intégrité de la base de données si une valeur de l'attribut `id_auteur` de la relation LIVRES ne correspondait à aucune valeur de la clef primaire de la relation AUTEURS).

## 2.6. Schéma relationnel

Les tables constituent la structure logique du modèle.

Un **schéma de relation** précise le nom de la relation ainsi que la liste des attributs avec leurs domaines.

On peut représenter un schéma de relation sous une forme graphique, par exemple à l'aide d'un diagramme :



On peut aussi le noter plus simplement sous la forme du nom de la relation suivi de la liste de ses attributs.

- **Produits** (Référence, Nom, En stock, Prix).
- **Ventes** (Numéro, #Référence produit, Date, Quantité).

Dans cette situation, les **clés primaires** apparaissent soulignées, les **clés étrangères** précédées d'un #.

On peut aussi également préciser les domaines des différents attributs :

- **Produits** (Référence : entier positif, Nom : str, En stock : booléen, Prix : chiffre à virgule).
- **Ventes**(Numéro : entier positif, #Référence produit : str, Date : date, Quantité : entier positif).