

# Les bases de données

## - Le modèle relationnel -

J. Boucher

Lycée Pierre-Paul RIQUET, Terminale NSI

7 novembre 2024

# Plan

- 1 I. Le modèle relationnel
  - 1. Définition
  - 2. Contraintes d'intégrité

# 1. Définition

## Définition

- Le **modèle relationnel** est un modèle dans lequel les données sont représentées par des ensembles de  $n$ -uplets appelés **relation**.
- Un élément d'une relation est appelé **entité** et possède des propriétés appelées des **attributs**.
- On spécifie une relation en donnant son **schéma** :

$$\text{Nom}(a_0 D_0, a_1 D_1, \dots, a_{n-1} D_{n-1})$$

Les attributs  $a_0, \dots, a_{n-1}$  peuvent prendre leur valeur respectivement parmi les ensembles de valeurs  $D_0, \dots, D_{n-1}$  appelés **domaines**.

- Un **base de données** est un ensemble de relations et le **schéma d'une base** est l'ensemble des schémas des relations qui la compose.

# 1. Définition

## Modélisation d'un livre

- Pour représenter un livre dans le modèle relationnel, on peut définir le schéma suivant :

*Livre*(*titre* String, *editeur* String, *annee* Int, *isbn* String)

permettant de spécifier la relation *Livre*.

- Les attributs *titre*, *editeur* et *isbn* sont des chaînes de caractères et *annee* un entier.
- L'ensemble des livres d'une bibliothèque peut alors être représenté par un ensemble de *n*-uplets :

```
Livre = {
    ('Harry Potter à l'école des sorciers', 'Gallimard', 1998, '2070518426'),
    ('Docteur Slump, Tome 01', 'Glénat', 2009, '2723472272'),
    ('The C++ Programming Language', 'Addison Wesley', 2013, '0321563840'),
    ... }
```

## 2. Contraintes d'intégrité

### Définitions

Les propriétés logiques qui assurent la **cohérence des données** au sein d'une base sont appelées **contraintes d'intégrité**. On distingue :

- les **contraintes de relation** qui garantissent que chaque entité d'une relation est **unique**. Pour cela un ensemble d'attributs est défini comme **clé primaire** dans le schéma de la relation, dont l'unicité garantit ces contraintes.
- les **contraintes de référence** qui créent des associations entre deux relations, en garantissant qu'une entité d'une relation mentionne une entité existante dans une autre relation.  
Pour garantir la satisfaction de ces contraintes, on définit un attribut de la première relation comme une **clé étrangère**, c'est à dire un ensemble d'attributs d'une table qui sont une clé primaire dans l'autre table.
- les **contraintes de domaine** qui restreignent les valeurs d'un attribut à celle du domaine et évitent que l'on puisse donner à un attribut une valeur illégale.

## 2. Contraintes d'intégrité

### Illustration de la contrainte de relation

Dans le schéma de la relation *Livre*,

*Livre*(*titre* String, *editeur* String, *annee* Int, *isbn* String)

l'attribut *isbn* est défini comme *clé primaire*. Il en résulte que deux entités ne pourront pas avoir la même valeur pour cet attribut et qu'une entité (un livre) est identifiable uniquement à partir de cette valeur.

## 2. Contraintes d'intégrité

### Illustration de la contrainte de référence

Dans le schéma de la relation *Emprunt*,

$$\textit{Emprunt}(\#code\_barre \textit{String}, \underline{\#isbn \textit{String}}, retour \textit{Date})$$

définit les attributs *code\_barre* et *isbn* comme des *clés étrangères*, indiquant ainsi que ces attributs doivent obligatoirement figurer dans les relations *Usager* et *Livre* comme clé primaire.

## 2. Contraintes d'intégrité

### Illustration de la contrainte de domaine

Dans le schéma de la relation *Usager*,

*Usager*(*nom* String, *prenom* String, *adresse* String, *cp* String,  
*ville* String, *courriel* String, *code\_barre* String)

l'attribut *cp* peut être défini comme une chaîne de caractère de longueur fixée, et non comme un entier, ce qui permet de garantir que la valeur possédera exactement cinq caractères.