

# Chapitre 02 : Bases de données

## Partie 1

---

NSI2

13 septembre 2021

# Introduction

---

Une **base de données** (BDD) est un ensemble structuré de données ainsi que des relations logiques sur ces données.

Une **base de données** (BDD) est un ensemble structuré de données ainsi que des relations logiques sur ces données.

Le **système de gestion de bases de données** (SGBD) peut être vu comme le logiciel qui gère :

Une **base de données** (BDD) est un ensemble structuré de données ainsi que des relations logiques sur ces données.

Le **système de gestion de bases de données** (SGBD) peut être vu comme le logiciel qui gère :

- la structuration des données;

Une **base de données** (BDD) est un ensemble structuré de données ainsi que des relations logiques sur ces données.

Le **système de gestion de bases de données** (SGBD) peut être vu comme le logiciel qui gère :

- la structuration des données;
- le stockage des données;

Une **base de données** (BDD) est un ensemble structuré de données ainsi que des relations logiques sur ces données.

Le **système de gestion de bases de données** (SGBD) peut être vu comme le logiciel qui gère :

- la structuration des données;
- le stockage des données;
- la maintenance des données;

Une **base de données** (BDD) est un ensemble structuré de données ainsi que des relations logiques sur ces données.

Le **système de gestion de bases de données** (SGBD) peut être vu comme le logiciel qui gère :

- la structuration des données;
- le stockage des données;
- la maintenance des données;
- la sécurité des données;



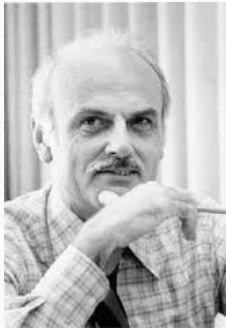
Une **base de données** (BDD) est un ensemble structuré de données ainsi que des relations logiques sur ces données.

Le **système de gestion de bases de données** (SGBD) peut être vu comme le logiciel qui gère :

- la structuration des données;
- le stockage des données;
- la maintenance des données;
- la sécurité des données;
- les accès aux données (lecture et/ou écriture) en temps réel par de multiples intervenants.

# Un peu d'histoire

La majorité de ce que nous allons voir repose sur les travaux d'Edgar F. Codd, c'est lui qui a inventé le modèle relationnel en 1970 alors qu'il travaillait comme informaticien chez IBM.



C'est une tâche essentielle pour assurer le bon fonctionnement des applications qui vont l'utiliser.

C'est une tâche essentielle pour assurer le bon fonctionnement des applications qui vont l'utiliser.

- **niveau conceptuel** : on représente la BDD à l'aide de schémas indépendamment de toute considération informatique;

C'est une tâche essentielle pour assurer le bon fonctionnement des applications qui vont l'utiliser.

- **niveau conceptuel** : on représente la BDD à l'aide de schémas indépendamment de toute considération informatique;
- **niveau logique** on adapte le schéma en tables à deux dimensions;

C'est une tâche essentielle pour assurer le bon fonctionnement des applications qui vont l'utiliser.

- **niveau conceptuel** : on représente la BDD à l'aide de schémas indépendamment de toute considération informatique;
- **niveau logique** on adapte le schéma en tables à deux dimensions;
- **niveau physique** on implémente les tables sur un SGBD.

Niveau conceptuel

Modèle entités-associations

---

On commence par déterminer les types des entités qui interviennent :



On commence par déterminer les types des entités qui interviennent :

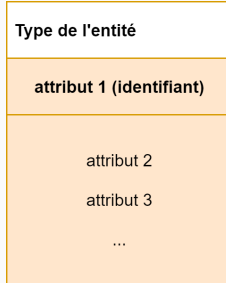
- une **entité** est un objet unique avec un nombre fini d'attributs;

On commence par déterminer les types des entités qui interviennent :

- une **entité** est un objet unique avec un nombre fini d'attributs;
- un (ou plusieurs) attribut(s) permet(tent) d'identifier de manière unique l'entité : on parle d'**identifiant**(s) ou de clé.

On commence par déterminer les types des entités qui interviennent :

- une **entité** est un objet unique avec un nombre fini d'attributs;
- un (ou plusieurs) attribut(s) permet(tent) d'identifier de manière unique l'entité : on parle d'**identifiant**(s) ou de clé.



## Exemple : entité-type Auteur



Une entité-type a en général plusieurs occurrences, appelées entités.

Une entité-type a en général plusieurs occurrences, appelées entités.

id_auteur	nom	prenom	date_naissance
00000001	Dupond	Marie	23/08/1982
12345678	Martin	Luce	13/05/1963
98765432	Leblanc	Jean	18/11/1974

Une entité-type a en général plusieurs occurrences, appelées entités.

id_auteur	nom	prenom	date_naissance
00000001	Dupond	Marie	23/08/1982
12345678	Martin	Luce	13/05/1963
98765432	Leblanc	Jean	18/11/1974

## Remarque

Par souci de simplicité, on parlera d'entité à la fois pour désigner l'entité-type et chacune de ses occurrences.

Voici une autre entité entrant en jeu dans la BDD :

<b>Pays</b>
<b>nom_pays</b>
population
superficie



Elles définissent des **liens sémantiques** (des liens de sens) que les simples entités ne suffisent pas à définir.

Elles définissent des **liens sémantiques** (des liens de sens) que les simples entités ne suffisent pas à définir.

Une association comporte :

Elles définissent des **liens sémantiques** (des liens de sens) que les simples entités ne suffisent pas à définir.

Une association comporte :

- un nom;

Elles définissent des **liens sémantiques** (des liens de sens) que les simples entités ne suffisent pas à définir.

Une association comporte :

- un nom;
- un lien entre 2 relations;

Elles définissent des **liens sémantiques** (des liens de sens) que les simples entités ne suffisent pas à définir.

Une association comporte :

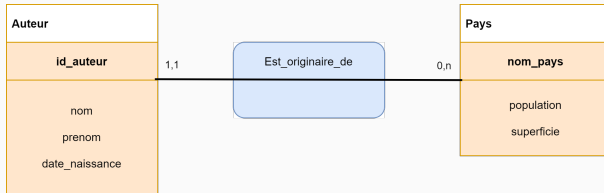
- un nom;
- un lien entre 2 relations;
- deux cardinalités qui sont représentées sur les extrémités du lien;

Elles définissent des **liens sémantiques** (des liens de sens) que les simples entités ne suffisent pas à définir.

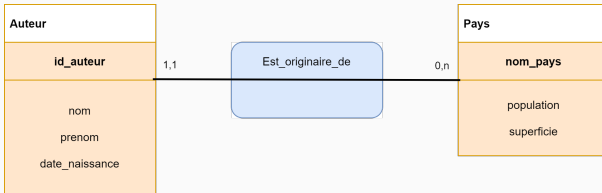
Une association comporte :

- un nom;
- un lien entre 2 relations;
- deux cardinalités qui sont représentées sur les extrémités du lien;
- parfois elle peut comporter un ou des attributs.

# Exemple



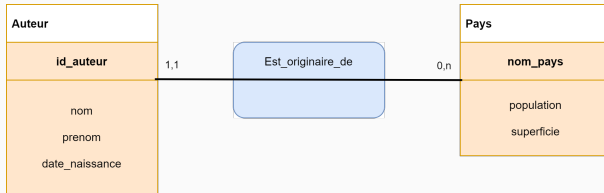
# Exemple



Une cardinalité est un couple d'entiers « de type min, max » :



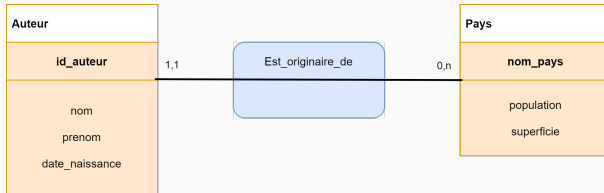
# Exemple



Une cardinalité est un couple d'entiers « de type min, max » :

- la cardinalité 1,1 signifie qu'un auteur peut être lié au minimum à 1 pays, et au maximum à 1 pays (donc à un pays et un seul);

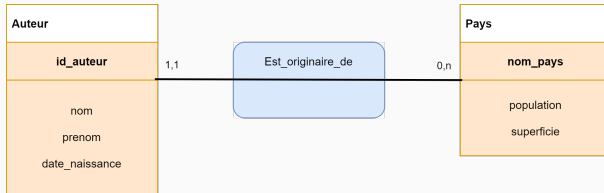
# Exemple



Une cardinalité est un couple d'entiers « de type min, max » :

- la cardinalité 1,1 signifie qu'un auteur peut être lié au minimum à 1 pays, et au maximum à 1 pays (donc à un pays et un seul);
- la cardinalité 0,n signifie qu'un pays peut être lié au minimum à aucun auteur et au maximum plusieurs auteurs.

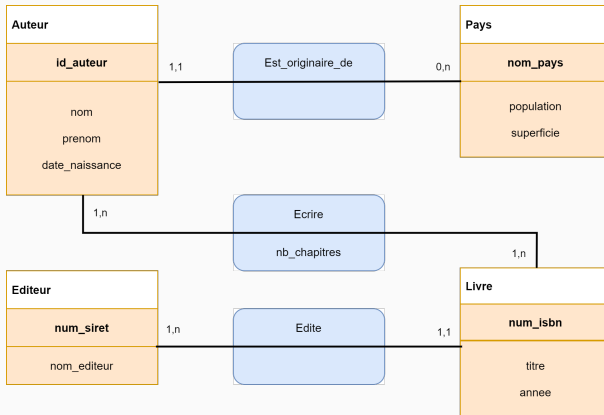
# Exemple



Une cardinalité est un couple d'entiers « de type min, max » :

- la cardinalité 1,1 signifie qu'un auteur peut être lié au minimum à 1 pays, et au maximum à 1 pays (donc à un pays et un seul);
- la cardinalité 0,n signifie qu'un pays peut être lié au minimum à aucun auteur et au maximum plusieurs auteurs.

# Un schéma abouti



Le schéma précédent s'appelle un modèle conceptuel des données (MCD).

Le schéma précédent s'appelle un modèle conceptuel des données (MCD).

On peut modéliser une situation avec plusieurs MCD, chacun d'entre eux ont leurs avantages et leurs inconvénients.

Le schéma précédent s'appelle un modèle conceptuel des données (MCD).

On peut modéliser une situation avec plusieurs MCD, chacun d'entre eux ont leurs avantages et leurs inconvénients.