

# Rappel : Les bases de données relationnelles

- Base de données / SQL

# La bibliothèque

- Un lecteur peut emprunter zéro ou plusieurs livres
- Un livre est écrit par un auteur ou plusieurs auteurs
- Un auteur peut avoir écrit un ou plusieurs livres

# La bibliothèque

- Un lecteur peut emprunter zéro ou plusieurs livres
- Un livre est écrit par un auteur ou plusieurs auteurs
- Un auteur peut avoir écrit un ou plusieurs livres

# Fichier à plat

Titre Livre	Nom auteur1	Info auteur1	Nom auteur2	Info auteur 2
Livre1	Auteur 1	Info auteur 1	...	
Livre2	Auteur 2	Info auteur 2		
Livre3	Auteur 1	Info auteur 1		
Livre4	Auteur 1	Info auteur 1		
...	...	...		

- Problèmes rencontrés :
  - Duplication des données
  - Espace disque occupé
  - Performance

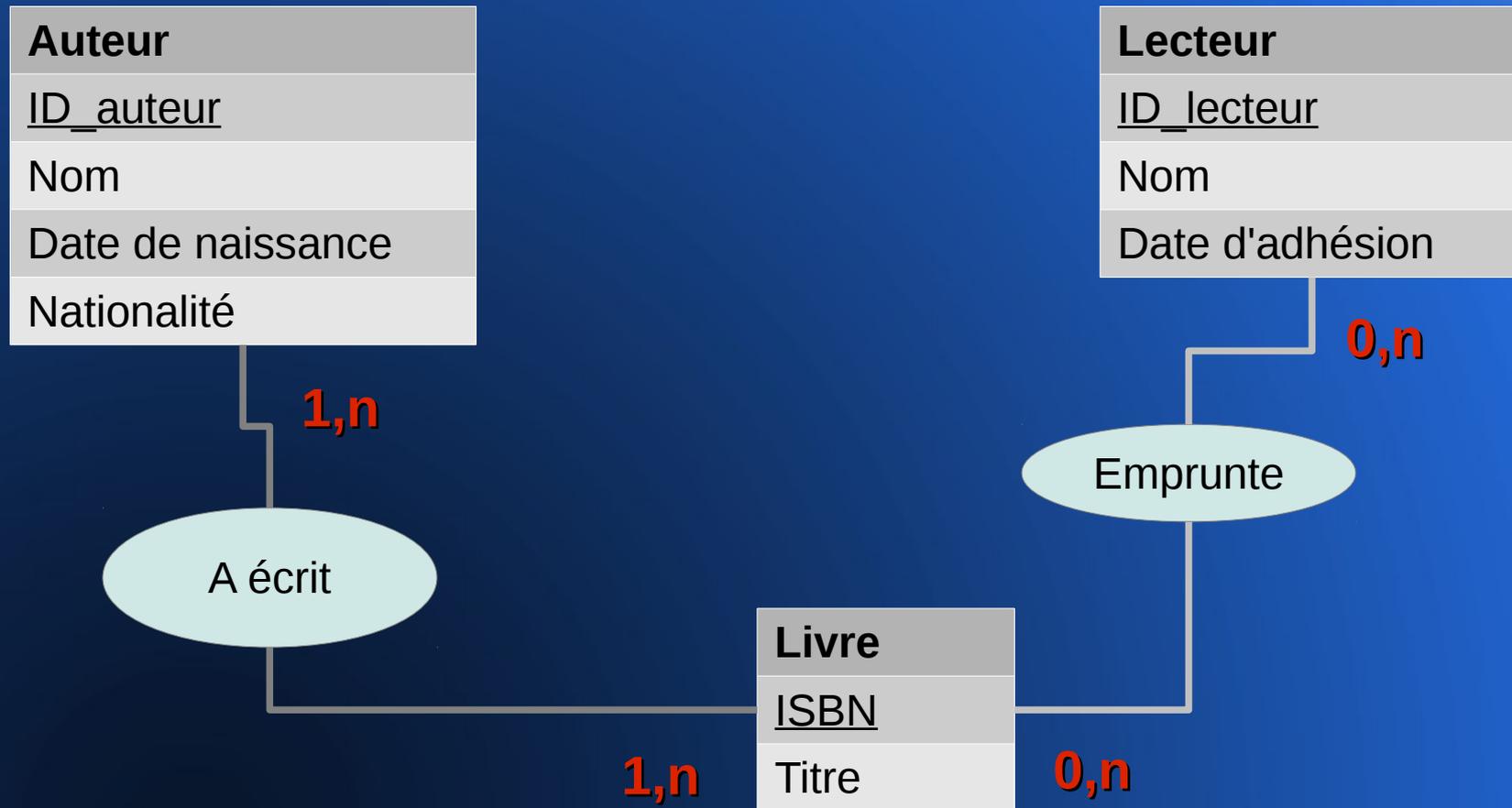
# Modèle relationnel

- Objectifs :
  - Exprimer les relations entre les objets
  - Éviter les duplications d'information
  - Intégrité
- Basé sur l'algèbre relationnel.
- SGBDR : Système de Gestion de Base de Données Relationnelle

# Aperçu de Merise

- Méthode d'analyse et de modélisation
  - Relation entre les entités
  - Description des relations

# Merise



# Merise



# SGBDR

- Avantages :
  - Indépendance du langage
  - Facilité de mise à jour
  - Cohérence des données
  - Performances / Indexation

# Le Langage

- Structured Query Language : SQL
- Créer une nouvelle base :
  - CREATE DATABASE Bibliotheque ;
  - SQLite : sqlite bibliotheque.db

# CREATE

- `sqlite> CREATE TABLE Livre (ISBN CHAR(13) PRIMARY KEY, Titre CHAR(255));`
- `sqlite> CREATE TABLE Auteur (ID_auteur INT PRIMARY KEY, Nom CHAR(255), DateNaissance DATE, Nat TEXT );`
- `sqlite> CREATE TABLE Ecrit_par (ID_Auteur INT, ISBN TEXT, FOREIGN KEY(ID_Auteur) REFERENCES Auteur(ID_auteur), FOREIGN KEY(ISBN) REFERENCES Livre(ISBN));`

# INSERT

- `sqlite> INSERT INTO Auteur VALUES (1, 'Taylor, Allen-G', '19700101', 'US');`
- `sqlite> INSERT INTO Livre VALUES ('9123562344442', 'SQL pour les nuls');`
- `sqlite> INSERT INTO Ecrit_par VALUES (4, '4545546646466');`
  - SQL error: foreign key constraint failed
- `sqlite> INSERT INTO Ecrit_par VALUES (1, '9123562344442');`

# SELECT

- `Ssqlite> select * from Ecrit_par ;`
  - `1|9123562344442`
- `sqlite> SELECT * FROM Livre,Auteur,Ecrit_par ;`
  - `9123562344442|SQL pour les nuls|1|Taylor, Allen-G|19700101|US|1|9123562344442`
  - `9123562344442|SQL pour les nuls|1|Taylor, Allen-G|19700101|US|2|9453562344442`
  - `9123562344442|SQL pour les nuls|2|JC|19740101|FR|1|9123562344442`
  - `9123562344442|SQL pour les nuls|2|JC|19740101|FR|2|9453562344442`
  - `9453562344442|Perl pour les nuls|1|Taylor, Allen-G|19700101|US|1|9123562344442`
  - `9453562344442|Perl pour les nuls|1|Taylor, Allen-G|19700101|US|2|9453562344442`
  - `9453562344442|Perl pour les nuls|2|JC|19740101|FR|1|9123562344442`
  - `9453562344442|Perl pour les nuls|2|JC|19740101|FR|2|9453562344442`

# SELECT

- `sqlite> SELECT * FROM Livre, Auteur, Ecrit_par WHERE Livre.ISBN = Ecrit_par.ISBN AND Auteur.ID_auteur = Ecrit_par.ID_Auteur ;`
  - 9123562344442|SQL pour les nuls|1|Taylor, Allen-G|19700101|US|1|9123562344442
  - 9453562344442|Perl pour les nuls|2|JC|19740101|FR|2|9453562344442
- `sqlite> SELECT Nom, Titre FROM Livre, Auteur, Ecrit_par WHERE Livre.ISBN = Ecrit_par.ISBN AND Auteur.ID_auteur = Ecrit_par.ID_Auteur ;`
  - Taylor, Allen-G|SQL pour les nuls
  - JC|Perl pour les nuls

# SELECT IMBRIQUE

- `sqlite> sqlite> SELECT * FROM Livre ;`
  - 9123562344442|SQL pour les nuls
  - 9453562344442|Perl pour les nuls
- `sqlite> SELECT * FROM Ecrit_par ;`
  - 1|9123562344442
  - 2|9453562344442
- `sqlite> SELECT * FROM Livre WHERE ISBN IN ( SELECT ISBN FROM Ecrit_par WHERE ID_Auteur = 1 );;`
  - 9123562344442|SQL pour les nuls

# UPDATE

- `sqlite> SELECT * FROM Auteur WHERE ID_Auteur = 1;`
  - 1|Taylor, Allen-G|19700101|US
- `sqlite> UPDATE Auteur SET Nom = 'JC2' WHERE Nom LIKE '%Taylor%';`
- `sqlite> SELECT * FROM Auteur WHERE ID_Auteur = 1;`
  - 1|JC2|19700101|US

# DELETE / DROP

- `sqlite> DELETE FROM Auteur WHERE Nom = 'JC' ;`
- `Sqlite> DROP TABLE Emprunt ;`

