

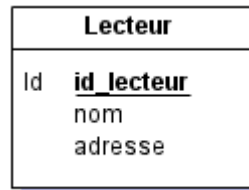
Base de données d'une bibliothèque

Modèle Conceptuel des Données

Entités

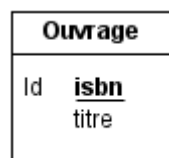
- **Lecteur**

- **attributs** : id_lecteur, nom, adresse
- **identifiant** : id_lecteur



- **Ouvrage**

- **attributs** : isbn, titre
- **identifiant** : isbn



Associations

- **Emprunt**

Association entre les **entités** Lecteur et Ouvrage

Attribut :

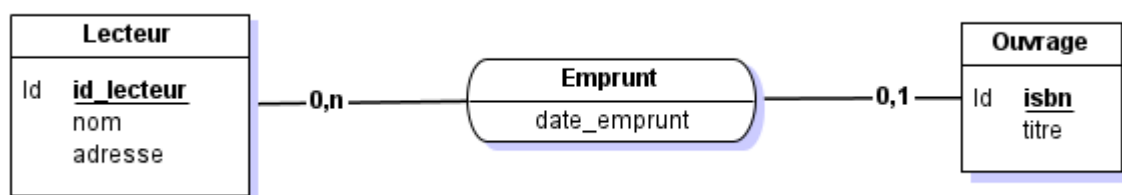
date_emprunt : permet de faire des rappels au lecteur (si date_emprunt n'est pas NULL, on peut réserver le livre).

Cardinalité avec Lecteur : (0, n)

Un lecteur peut ne pas emprunter ou emprunter plusieurs ouvrages.

Cardinalité avec Ouvrage : (0, 1)

Un ouvrage peut ne pas être emprunté ou être emprunté au plus une fois (chaque ouvrage n'est présent qu'en un exemplaire).



- **Reservation**

Association entre les **entités** Lecteur et Ouvrage

Attribut : instant_reservation

L'attribut permet de savoir à quel instant a été faite la réservation, pour qu'une priorité soit établie entre les lecteurs.

Cardinalité avec Lecteur : (0, n)

Un lecteur peut ne pas faire de réservation ou faire plusieurs réservations.

Cardinalité avec Ouvrage : (0, n)

Un ouvrage peut ne pas être réservé ou être réservé par plusieurs personnes.

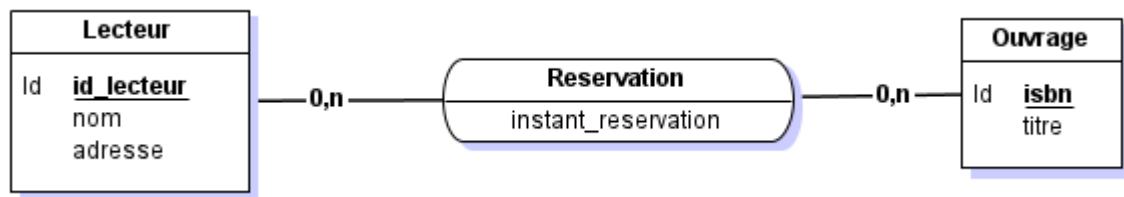
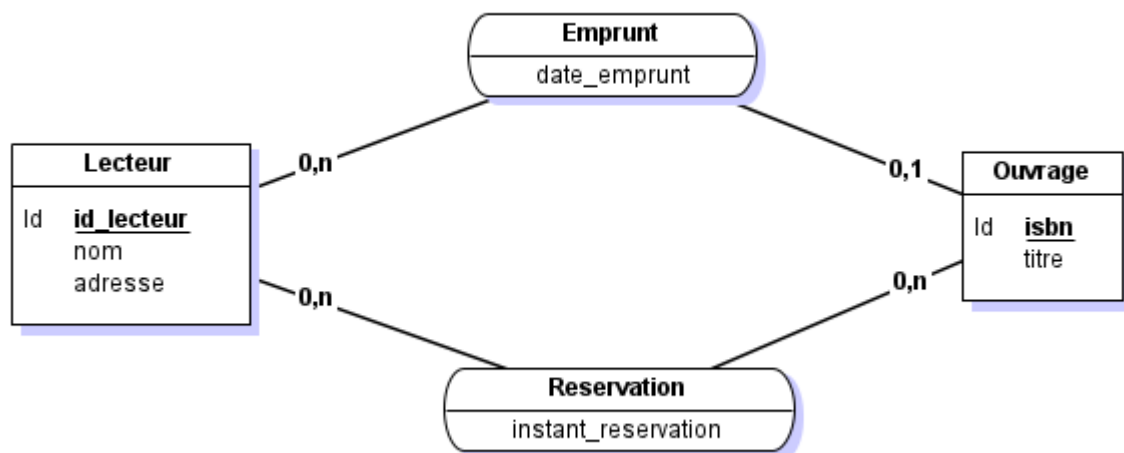


Schéma Entité-Association



Modèle Logique des Données

Tables

- **Lecteur** (id_lecteur, nom, adresse)

- attributs : id_lecteur, nom, adresse
- clé primaire : id_lecteur

| id_lecteur | nom | adresse |
|------------|-----|---------|
| ... | ... | ... |

- **Ouvrage** (isbn, titre, date_emprunt, #id_lecteur)

- attributs : isbn, titre, date_emprunt, id_lecteur
- clé primaire : isbn
- clé étrangère : #id_lecteur (correspond à la clé primaire id_lecteur de la table **Lecteur**)

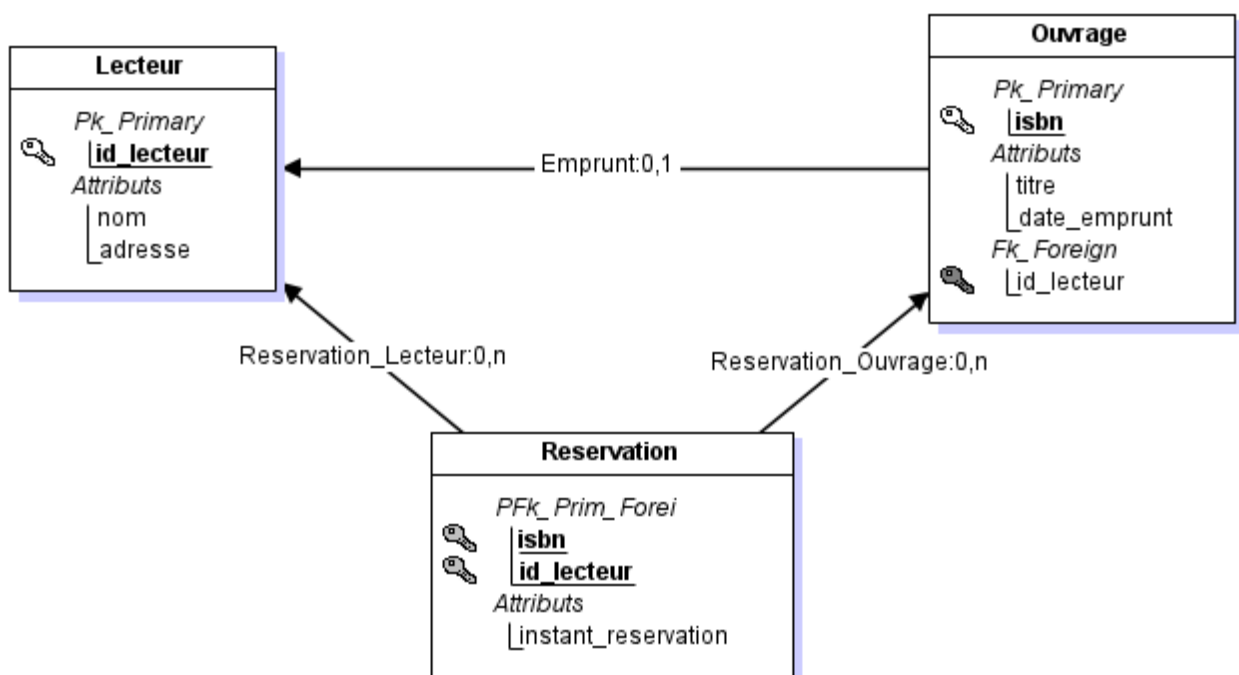
| isbn | titre | date_emprunt | id_lecteur |
|------|-------|--------------|------------|
| ... | ... | ... | ... |

- **Reservation** (#id_lecteur, #isbn, instant_reservation)

- attributs : id_lecteur, isbn, instant_reservation
- clé primaire : id_lecteur, isbn
- clés étrangères : #id_lecteur (correspond à la clé primaire id_lecteur de la table **Lecteur**)
#isbn (correspond à la clé primaire isbn de la table **Ouvrage**)

| id_lecteur | isbn | instant_reservation |
|------------|------|---------------------|
| ... | ... | ... |

Schéma relationnel (JMerise)



Définition des données

Un exemple de code SQL pour générer ces tables :

```
CREATE TABLE Lecteur(  
    id_lecteur Int NOT NULL ,  
    nom Varchar (50) NOT NULL ,  
    adresse Varchar (50) NOT NULL ,  
    CONSTRAINT Lecteur_PK PRIMARY KEY (id_lecteur)  
);  
  
CREATE TABLE Ouvrage(  
    isbn Int NOT NULL ,  
    titre Varchar (50) NOT NULL ,  
    date_emprunt Date ,  
    id_lecteur Int ,  
    CONSTRAINT Ouvrage_PK PRIMARY KEY (isbn) ,  
  
    CONSTRAINT Ouvrage_Lecteur_FK FOREIGN KEY (id_lecteur) REFERENCES  
Lecteur(id_lecteur)  
);  
  
CREATE TABLE Reservation(  
    isbn Int NOT NULL ,  
    id_lecteur Int NOT NULL ,  
    instant_reservation Datetime default CURRENT_TIMESTAMP ,  
    CONSTRAINT Reservation_PK PRIMARY KEY (isbn,id_lecteur) ,  
  
    CONSTRAINT Reservation_Ouvrage_FK FOREIGN KEY (isbn) REFERENCES  
Ouvrage(isbn) ,  
    CONSTRAINT Reservation_Lecteur0_FK FOREIGN KEY (id_lecteur)  
REFERENCES Lecteur(id_lecteur)  
);
```

Tables (3)

| Nom | Type | Schéma |
|---------------------|-------------|---|
| Lecteur | | CREATE TABLE Lecteur(id_lecteur Int NOT NULL , nom Varchar (50) NOT NULL , adresse Varchar (50) NOT NULL , CONSTRAINT Lecteur_PK PRIMARY KEY (id_lecteur)) |
| id_lecteur | Int | "id_lecteur" Int NOT NULL |
| nom | Varchar(50) | "nom" Varchar(50) NOT NULL |
| adresse | Varchar(50) | "adresse" Varchar(50) NOT NULL |
| Ouvrage | | CREATE TABLE Ouvrage(isbn Int NOT NULL , titre Varchar (50) NOT NULL , date_emprunt Date , id_lecteur Int , CONSTRAINT Ouvrage_PK PRIMARY KEY (isbn) , CONSTRAINT Ouvrage_Lecteur_FK FOREIGN KEY (id_lecteur) REFERENCES Lecteur(id_lecteur)) |
| isbn | Int | "isbn" Int NOT NULL |
| titre | Varchar(50) | "titre" Varchar(50) NOT NULL |
| date_emprunt | Date | "date_emprunt" Date |
| id_lecteur | Int | "id_lecteur" Int |
| Reservation | | CREATE TABLE Reservation(isbn Int NOT NULL , id_lecteur Int NOT NULL , instant_reservation Datetime default CURRENT_TIMESTAMP , CONSTRAINT Reservation_PK PRIMARY KEY (isbn,id_lecteur) , CONSTRAINT Reservation_Ouvrage_FK FOREIGN KEY (isbn) REFERENCES Ouvrage(isbn) , CONSTRAINT Reservation_Lecteur0_FK FOREIGN KEY (id_lecteur) REFERENCES Lecteur(id_lecteur)) |
| isbn | Int | "isbn" Int NOT NULL |
| id_lecteur | Int | "id_lecteur" Int NOT NULL |
| instant_reservation | Datetime | "instant_reservation" Datetime DEFAULT CURRENT_TIMESTAMP |

Manipulation des données

Quelques exemples de manipulation de données sur la base de données :

```
-- ajouts Lecteur
INSERT INTO Lecteur
VALUES( 1 , 'Edgar Cobb' , 'Exeter College, Oxford' );

INSERT INTO Lecteur
VALUES( 2 , 'James Gray' , 'University of California, Berkeley' );

INSERT INTO Lecteur
VALUES( 3 , 'Serge Abiteboul' , 'University of Southern California' );

-- ajouts Ouvrage
INSERT INTO Ouvrage(isbn, titre)
VALUES( 9782711786459 , 'Fondements des bases de données' );

INSERT INTO Ouvrage(isbn, titre)
VALUES( 9781558601901 , 'Transaction Processing' );

INSERT INTO Ouvrage(isbn, titre)
VALUES( 9780201099553 , 'Relational Databases and Knowledge Bases' );

INSERT INTO Ouvrage(isbn, titre)
VALUES( 9780201141924 , 'The Relational Model for Database Management' );

-- emprunt ouvrage
UPDATE Ouvrage
SET date_emprunt = date('now'), id_lecteur = 1
WHERE isbn = 9782711786459;

-- reservation ouvrage
INSERT INTO Reservation(isbn, id_lecteur)
VALUES( 9782711786459 , 2 );

-- reservation ouvrage
INSERT INTO Reservation(isbn, id_lecteur)
VALUES( 9782711786459 , 3 );

-- retour ouvrage
UPDATE Ouvrage
SET date_emprunt = NULL, id_lecteur = NULL
WHERE isbn = 9782711786459;

-- emprunt ouvrage quand reservation
UPDATE Ouvrage
SET date_emprunt = date('now'), id_lecteur = 2
WHERE isbn = 9782711786459;

DELETE FROM Reservation
WHERE isbn = 9782711786459 AND id_lecteur = 2 ;

-- emprunt ouvrage
UPDATE Ouvrage
SET date_emprunt = date('now'), id_lecteur = 2
WHERE isbn = 9780201141924;

-- reservation ouvrage
INSERT INTO Reservation(isbn, id_lecteur)
VALUES( 9780201141924 , 3 );

-- emprunt ouvrage
UPDATE Ouvrage
SET date_emprunt = date('now'), id_lecteur = 1
WHERE isbn = 9781558601901;

-- reservation ouvrage
INSERT INTO Reservation(isbn, id_lecteur)
VALUES( 9781558601901 , 2 );
```

Tables

| isbn | titre | date_emprunt | id_lecteur |
|---------------|--|--------------|------------|
| 9782711786459 | Fondements des bases de données | 2020-06-23 | 2 |
| 9781558601901 | Transaction Processing | 2020-06-23 | 1 |
| 9780201099553 | Relational Databases and Knowledge Bases | | |
| 9780201141924 | The Relational Model for Database Management | 2020-06-23 | 2 |

| id_lecteur | nom | adresse |
|------------|-----------------|------------------------------------|
| 1 | Edgar Cobb | Exeter College, Oxford |
| 2 | James Gray | University of California, Berkeley |
| 3 | Serge Abiteboul | University of Southern California |

| isbn | id_lecteur | instant_reservation |
|---------------|------------|---------------------|
| 9782711786459 | 3 | 2020-06-23 12:19:32 |
| 9780201141924 | 3 | 2020-06-23 12:24:13 |
| 9781558601901 | 2 | 2020-06-23 12:25:29 |