Introduction aux bases de données

***Répondre aux questions suivantes***

Qu’est-ce qu’une base de données ?

Une base de données (BD) regroupe un ensemble de données structurées et organisées.

Le but c’est de données un accès simplifié aux utilisateurs ou aux autres programmes.

Quelles sont ses avantages ?

Les BD ont l’avantage de permettre de combiner, relier facilement les données sans connaître la structure exacte de l’enregistrement des données

**Fichier élèves :**

Nom1 prenom1 Classe1 Email1

Nom2 prenom2 Classe2 Email2

….

NomN prenomN ClasseN EmailN

**Programme de lecture :**

Pour lire ce fichier il faut connaître parfaitement sa structure.

Lorsqu’on modifie le fichier il faut obligatoirement modifier les programmes.

Nous sommes pas sûre d’avoir une

Seule version de fichiers.

**BD élèves :**

**Nom** **Prénom** **Classe**

Nom1Prenom1 classe1

Nom2Prenom2 classe2

… … …

NomNPrenomN classeN

**Programme de lecture BD :**

Pour lire une information pas besoin

De connaître la structure de la BD.

Si on ajoute un champ les programmes restent valident

Les BD sont multi-utilisateur tout le monde travail sur les mêmes données

Qu’est-ce qu’un SGBD ?

C’est un Système de Gestion de Base de Données, c’est un ensemble de logiciels

qui gère une base de données, il permet :

- Créer la base de données

- modifier la base de données (insérer, modifier, supprimer des données)

- rechercher les données

- partager les données et spécifier des droits d’accès

- etc

Exemple de SGBD : **MySQL, MariaDB,** Oracle, SQL Server, Access, etc

Qu’est-ce qu’une relation ?

Dans une BD les informations sont regroupées dans des relations,

exemple de la relation Etudiant.

**Etudiant**(Nom, Prénom, Age, Classe)

**Cours**(Nomcours, durée,date)

Qu’est-ce qu’une table ?

C’est la représentation graphique d’une relation.

TABLE Etudiant

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Prénom | **Age** | **Classe** |
| Nom1 | Prenom1 | Age1 | classe1 |
| Nom2 | Prenom2 | Age2 | classe2 |

Qu’est-ce qu’une clé **primaire**?

C’est un champ ou un ensemble de champ qui permet d’identifier de manière unique un enregistrement (ligne) de la table.

Par exemple pour la table « Etudiant » on pourrait choisir comme clé primaire «l’association « nom » + « prenom », mais ce choix n’est pas judicieux car on peut avoir des homonymes (meme nom et prénom).

Le choix le plus judicieux est d’ajouter à la relation « Etudiant » un champ unique pour faire office de clé primaire par exemple :

- idEtudiant

- Email

TABLE Etudiant

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IdEtudiant | Nom | Prénom | **Age** | **Classe** |
| 1 | Nom1 | Prenom1 | Age1 | classe1 |
| 2 | Nom2 | Prenom2 | Age2 | classe2 |

Qu’est-ce qu’une clé **étrangère**?

C’est un champ d’une relation qui est clé primaire dans une autre table.

Elles permet de lier virtuellement les deux tables pour par exemple combiner les informations

TABLE Etudiant

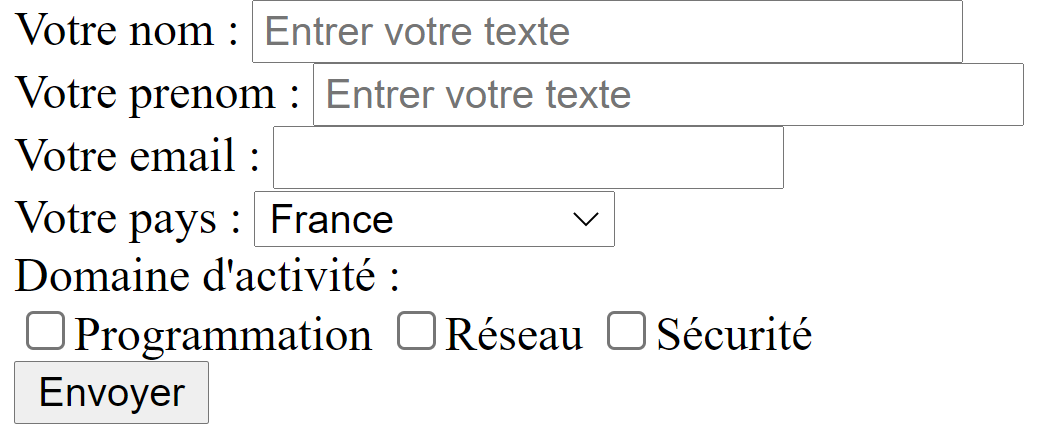
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IdEtudiant | Nom | Prénom | **Age** | **#idClasse** |
| 1 | Nom1 | Prenom1 | Age1 | 1 |
| 2 | Nom2 | Prenom2 | Age2 | 2 |
| 3 | Nom3 | Prenom3 | Age3 | 1 |

TABLE Classe

|  |  |  |
| --- | --- | --- |
| IdClasse | NomClasse | **Annee\_Scolaire** |
| 1 | 1TSSNIR | 2021-2022 |
| 2 | 2TSSNIR | 2021-2022 |
| 3 | 1CPI | 2021-2022 |

Qu’est-ce qu’un formulaire ?

C’est une interface graphique qui permet à l’utilisateur de saisir des informations.



Qu’est-ce qu’une requête ?

C’est une demande envoyer au Système de Gestion de Base de Données en utilisant un langage précis.

Qu’est-ce que le langage SQL ?

C’est un langage normalisé (langage commun à tous), haut niveau(langage proche du langage humain) qui permet de dialoguer avec le serveur de base de données.

Il y a des requêtes SQL pour :

- **Rechercher** des données (SELECT…)

- **Modifier** des données (UPDATE…)

- **Supprimer** des données (DELETE…)

- **Créer** des tables (CREATE TABLE…)

- **Insérer** des données (INSERT INTO…)

- **Donner** des **droits** (GRANT…)

- **Retirer** des **droits** (REVOKE…)

- etc.

***Soit la Base de Données relationnelle simplifiée d’une bibliothèque :***

**Schéma de la base de données**

**Livre**

NumLivre

Titre

Auteur

Genre

(Prix)

Personne

NumPersonne

Nom

Prénom

Ville

**Emprunt**

#NumLivre

sortie

(retour)

#NumPersonne

- Les clés sont soulignées

* Une contrainte référentielle est représentée par une flèche qui part de clé étrangère et pointe sur le cartouche de l’entité référence.
* Les attributs facultatifs sont placés entre parenthèses.

**Extension ou : Instance de la B.D**

La Table **livre**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NumLivre** | **titre** | **auteur** | **genre** | **prix** |
| 1 | Les chouans | Balzac | roman | 80 |
| 2 | Germinal | Zola | roman | 75 |
| 3 | L’assommoir | Zola | roman | 95 |
| 4 | La bête humaine | Zola | roman | 70 |
| 5 | Les misérables | Hugo | roman | 105 |
| 6 | La peste | Camus | roman | 112 |
| 7 | Les lettres persanes | Maupassant | roman | 140 |
| 8 | Bel ami | Maupassant | roman | 76 |
| 9 | Les lettres de mon moulin | Daudet | roman | 100 |
| 10 | César | Pagnol | roman | 100 |
| 11 | Marius | Pagnol | roman | 65 |
| 12 | Fanny | Pagnol | roman | 72 |
| 13 | Les fleurs du mal | Baudelaire | poésie | 130 |
| 14 | Paroles | Prévert | poésie | 120 |
| 15 | Les raisins de la colère | Steinbeck | roman | *135* |

La table **personne**:

|  |  |  |  |
| --- | --- | --- | --- |
| **NumPersonne** | **nom** | **prenom** | **ville** |
| 1 | Durand | Jean-Pierre | Toulouse |
| 2 | Brieusel | Chantai | Colomiers |
| 3 | Riols | Jacques | Toulouse |
| 4 | Denayville | Hélêne | Toulouse |
| *5* | Planchon | André | Muret |
| 6 | Pêne | Gérôme | Albi |
| 7 | Bert | Jean-Pierre | StOrens |
| 8 | Gonzales | Alain | Toulouse |
| *9* | Martin | François | Balma |
| 10 | Jourda | Véroni ue | Colomiers |

La table **emprunt**

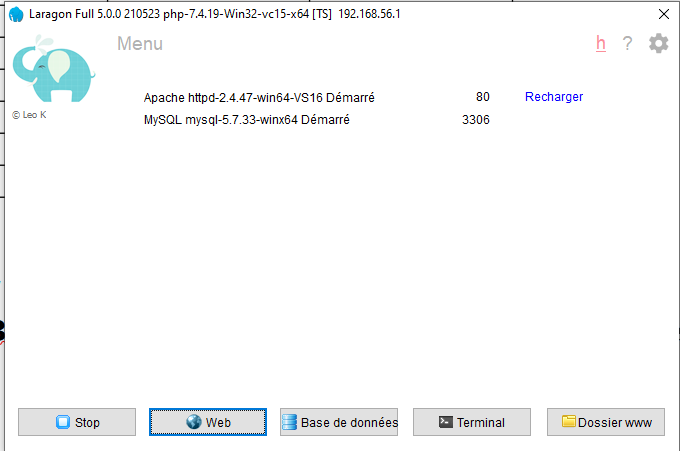
|  |  |  |  |
| --- | --- | --- | --- |
| **#NumPersonne** | **#Numlivre** | **sortie** | **(retour)** |
| 4 | 14 | 01/01/99 |  |
| 1 | 3 | 03/03/99 | 30/03/99 |
| 7 | 9 | 05/03/99 | 21/03/99 |
| 2 | 11 | 18/03/99 |  |
| 3 | 3 | 30/03/99 | 15/04/99 |
| 3 | 4 | 30/03/99 |  |
| 8 | 7 | 31/03/99 | 18/04/99 |
| 8 | 1 | 02/04/99 |  |
|  |  |  |  |

***TRAVAIL A FAIRE :***

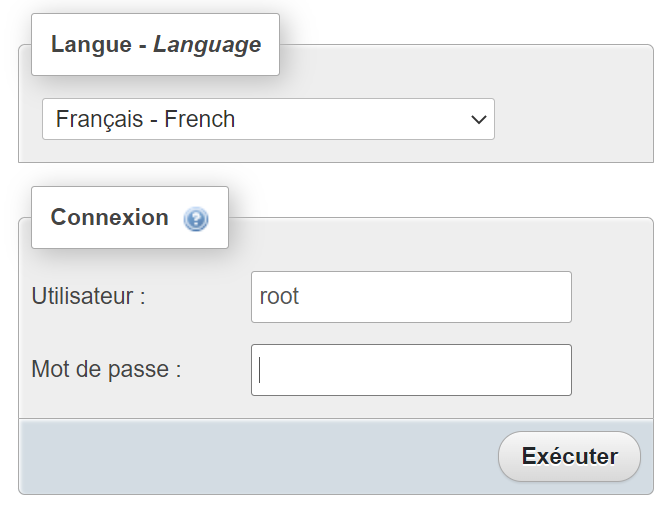
1. **Utiliser le SGBD MySQL pour créer une application de gestion de prêt de cette bibliothèque.**

**Étapes :**

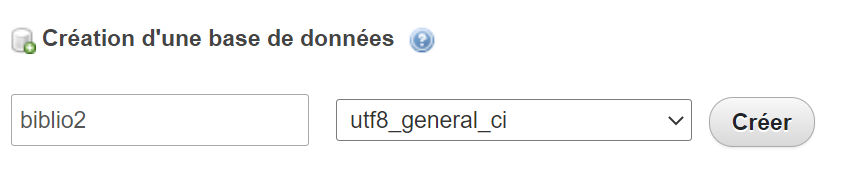
1. Lancer le logiciel « Laragon »
2. cliquer sur le bouton « Base de données »



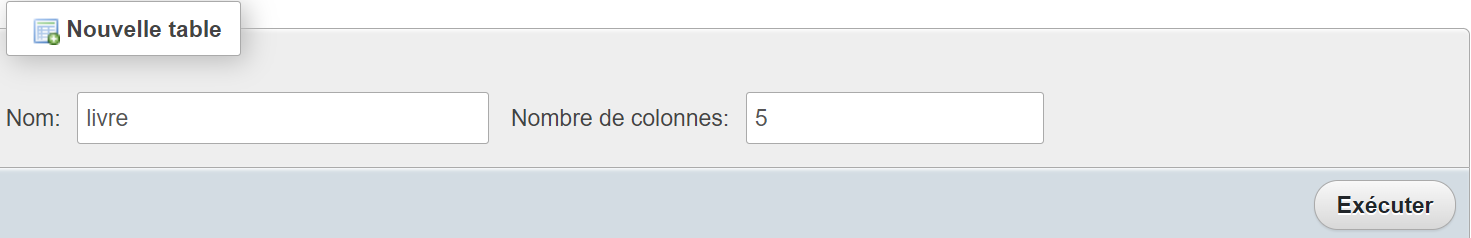
C) S’authentifier en tant qu’utilisateur « root » (super utilisateur) sans mot de passe.



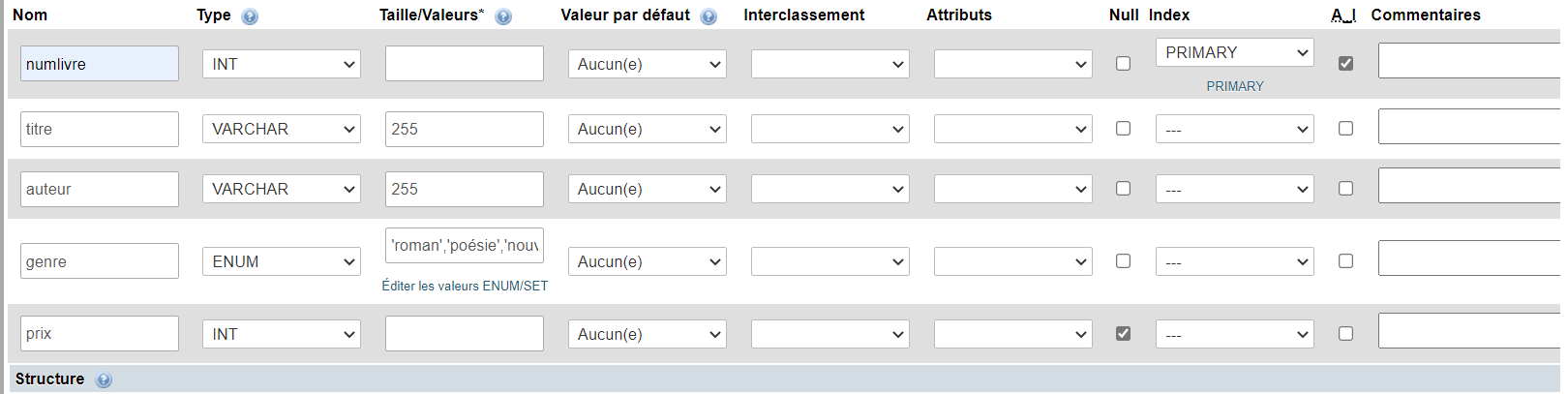
D)Cliquer sur « Base de données » et créer la BD « biblio2 » avec le codage de caractères « utf8\_general\_ci »



E) Créer la table « livre » en spécifiant son nombre de colonnes

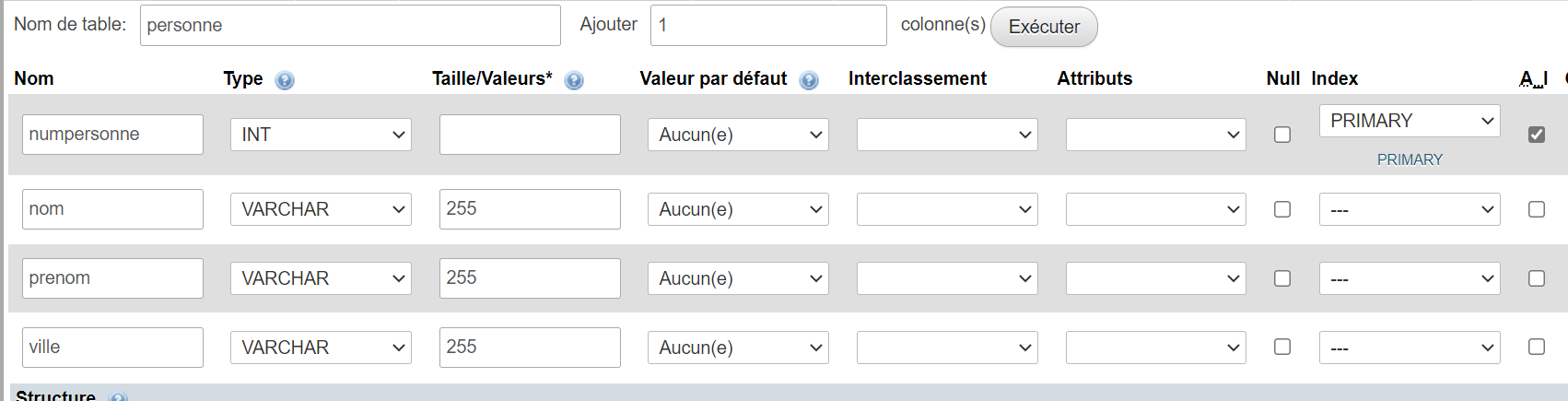


E) Définir le **nom**, le **type** et les **contraintes** des champs(colonnes) de la table livre.

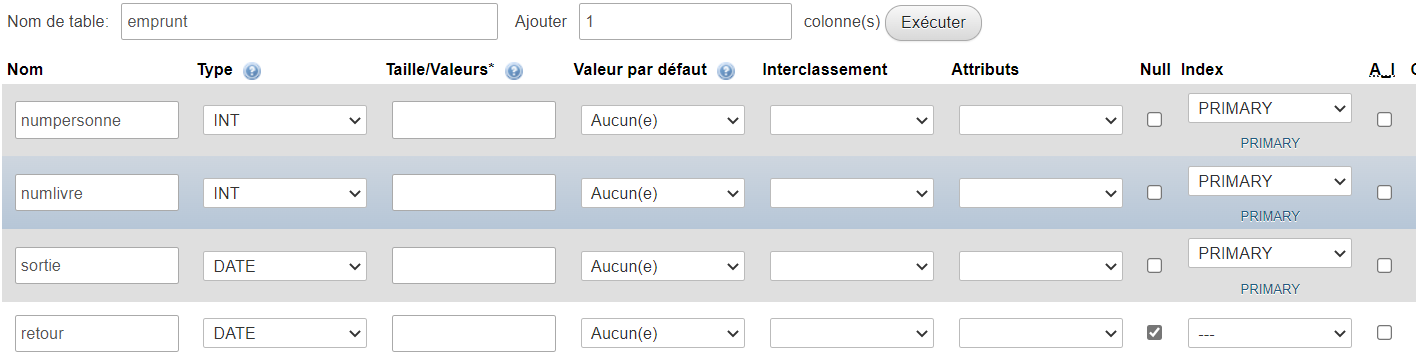


F) Créer la table «personne» en spécifiant son nombre de colonnes

G) Définir le **nom**, le **type** et les **contraintes** des champs(colonnes) de la table personne.

H) Créer la table «emprunt» en spécifiant son nombre de colonnes

I) Définir le **nom**, le **type** et les **contraintes** des champs(colonnes) de la table emprunt.

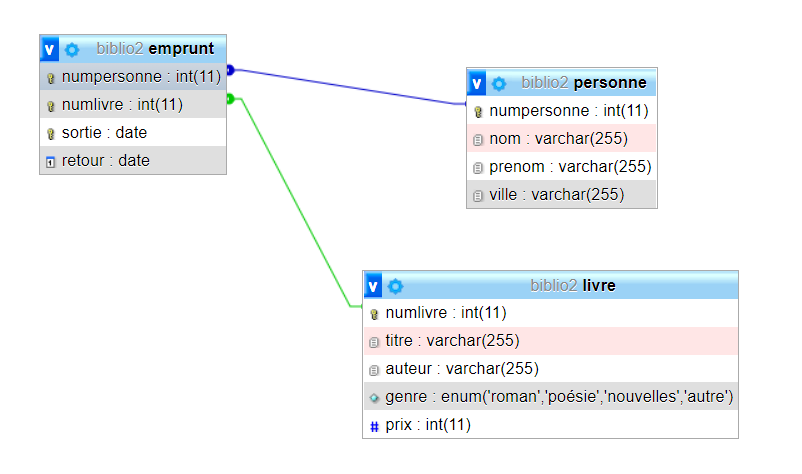


J) Définir les contraintes d’intégrité référentiel des clés étrangères dans la table emprunt.

- Sélectionner sur la gauche la BD « Biblio2 », cliquez sur l’onglet « Concepteur »

- ajouter une relation entre le champ « numlivre » de la table « livre » vers le champ « numlivre » de la table « emprunt »

- ajouter une relation entre le champ « numpersonne » de la table «personne» vers le champ « numpersonne » de la table « emprunt »



1. **Écrire les requêtes SQL qui permettent de répondre aux questions suivantes :**

**A. Requêtes sur une table**

* les livres écrit par Zola et dont le prix est inférieur à 75 francs ?
* les livres écrit ni par Balzac et ni par Zola ?
* les auteurs qui se trouvent après camus dans l’ordre alphabétique ?
* les titres dont le prix est compris entre 80 et 100 et les ordonner par prix croissant ?
* les romans rangés par auteur et par titre ordonnés de façon décroissante ?
* livres en cours d’emprunt ?
* les titres commençant par « les lettres » ?
* les noms de personnes de 6 lettres ?
* les emprunteurs habitant Toulouse, Colomiers ou Muret

Questions supplémentaires :

1. Afficher toutes les données sur les poésies ?
2. Trier la table livre par prix décroissant ?
3. Afficher les livres de Balzac ?
4. Afficher les données sur les personnes de Toulouse ou de Muret ?
5. Afficher pour les livres en cours d’emprunt la date de sortie et de retour ?
6. Afficher tous les livres dont le titre contient « les » ?
7. Afficher toutes les personnes dont le nom contient un « L » ?
8. Afficher les personnes triées par ordre alphabétique décroisant ?
9. Afficher tous les emprunts de janvier 1999 ?
10. Afficher les livres écrits par ‘Balzac’, ‘Camus’ ou ‘Zola’ ?

**B. Requêtes utilisant la jointure**

* Titre et auteur des livres en cours d’emprunt ?
* Nom des personnes qui possède des livres en cours emprunt ?
* Ensemble des livres lus et retournés par Jean-Pierre Durand

1. **Requête utilisant les fonctions de calcul**

**Principales fonctions de calcul du SQL :**

**SUM**(<nom d’un champ>) : retourne la somme

**AVG**(<nom d’un champ>) : retourne la moyenne

**MAX**(<nom d’un champ>) :retourne la valeur maximale

**MIN**(<nom d’un champ>) :retourne la valeur minimal

**COUNT(\*)** ou **COUNT**(<nom d’un champ>) : retourne le nombre de lignes (enregistrement)

**Principales fonctions sur les dates :**

**YEAR (**<nom d’un champ de type date>) : retourne l’année

**MONTH (**<nom d’un champ de type date>) : retourne le mois

**DAY (**<nom d’un champ de type date>) : retourne le jour

**NOW()** : retourne la date actuelle

* + Total des prix des livres ?
  + Livre le plus coûteux et le moins ?
  + Nombre d’auteurs ?
  + La liste des auteurs avec le nombre de livres de chacun ?
  + Le nombre de livres, le prix moyen, l’écart de prix maximum ?

1. **Requête utilisant la clause Having (restreindre après un GROUP BY)**
   * Pour chaque auteur ayant écrit plus de 2 livres, indiquer le prix du livre le moins coûteux et celui du plus coûteux

**E. utilisation des sous requêtes**

**-** titre, auteur, genre et prix du livre le moins coûteux ?

- informations sur les livres dont le prix est supérieur au prix moyen ?

- personne qui habite la même ville que jean-Pierre Durant ?

- informations sur les livres jamais emprunté

1. **Écrire les requêtes SQL qui permettent de modifier la base de données :**

- **Ajouter** le « petit prince » de St Exupéry dans la table livre

- **Ajoutez** vous dans la table personne

- Aujourd’hui vous empruntez « le petit prince », **ajouter** les données dans la table emprunt

- Jean-Pierre Durant déménage à Nice, **modifier** la table personne

- **Augmentez** le prix du « petit prince de 5 €

- **Supprimez** vous de la table personne

1. **Écrire les requêtes SQL qui permettent de créer des vues :**

a) créer une vue qui donnes la liste des livres jamais empruntés

**CREATE VIEW** viewJamaisEmprunté **AS**

SELECT \*

FROM livre

WHERE numlivre NOT IN (SELECT numlivre

FROM emprunt);

b) créer une vue qui liste les emprunts effectués par des toulousains

CREATE VIEW emprunt ParTousain AS

SELECT nom, prenom, ville,sortie,retour

FROM personne JOIN emprunt ON personne.numpersonne =emprunt.numpersonne

WHERE ville = 'Toulouse';

c) créer une vue qui donnes le nombre d’emprunt par ville

CREATE VIEW NombreEmpruntParVille AS

SELECT ville,COUNT(\*) AS nb\_emprunts

FROM personne JOIN emprunt ON personne.numpersonne =emprunt.numpersonne

GROUP BY ville

d) créer une vue qui donnes les informations sur les livres les plus empruntés

CREATE VIEW livreLePlusEmprunte AS

SELECT emprunt.numlivre,titre,auteur, COUNT(\*) nbEmprunt

FROM emprunt JOIN livre ON emprunt.numlivre = livre.numlivre

GROUP BY emprunt.numlivre,titre,auteur

ORDER BY nbEmprunt DESC

LIMIT 1;