

RAPPORT DE PROJET

Mise sur pied d'une bibliothèque Universitaire



INTRODUCTION

a. Contexte

La bibliothèque par essence, est le centre de concentration du savoir, elle conserve une trace écrite des pensées et des prouesses humaines. Dans notre université, nous avons un système de gestion de bibliothèque encore archaïque qui, plutôt que de conserver le savoir, le pousse bien au contraire à se perdre. Nous constatons des lourdeurs dans les processus de traitement et de gestion de l'ensemble des opérations au sein de la bibliothèque. Le service est très lent, la gestion des stocks se fait encore au décompte et il est difficile de suivre l'historique des emprunts et des réservations pouvant entraîner des pertes de livres.

Tout ceci rend le processus de travail archaïque et inefficace face à la demande sans cesse croissante au sein de la bibliothèque due aux grands effectifs de l'université et qui sans doute augmenteront les rentrées prochaines considérablement.

b. Objectifs

Notre initiative à travers ce projet que nous lançons vise la modernisation de notre bibliothèque ceci à travers un **“SYSTÈME DE GESTION EFFICACE ET DURABLE DE LA BIBLIOTHÈQUE”** avec une base de données que nous alimenterons au fil du temps en fonction du nombre d'abonnés qui viendront s'enrôler en salle de lecture.

MÉTHODOLOGIE

Pour parvenir à un système qui répond pleinement aux exigences susmentionnées, nous avons choisi de procéder méthodiquement.

1. Choix des entités et définition des associations

Suivant les exigences de notre système qui sont entre autres :

a. *La gestion des livres :*

- Ajout, modification et suppression de livres ;
- Recherche de livres par titre, auteur, genre, catégorie ;
- Gestion des stocks et des exemplaires disponibles ;
- Notation des livres.

b. *La gestion des Utilisateurs :*

- Inscription des étudiants et des membres du personnel ;
- Authentification et gestion du compte ;
- Gestion des droits d'accès ;
- Gestion des informations des utilisateurs (nom, prénom, numéro d'étudiant, contacts...) ;
- Gestion des emprunts et des réservations par utilisateur.

c. *La gestion des prêts et retours :*

- Enregistrement des prêts de livres ;
- Suivi des dates de retour prévues et des retards ;
- Enregistrement des retours et mise à jour des stocks.

Nous avons répertorié 4 entités et 2 associations

d. Entités

- ❖ **Livres** : id_livre, titre, auteur, annee_publication, genre, nombre_exemplaires
- ❖ **Étudiants** : id_etudiant, nom, prenom, date_naissance, email, telephone, date_inscription
- ❖ **Emprunts** : id_emprunt, id_etudiant, id_livre, date_emprunt, date_retour
- ❖ **Genres** : id_genre, nom_genre

Associations

Association	Cardinalité	Explication
emprunter	(0.N : 0.N)	1 étudiant peut emprunter 0 ou plusieurs livres de même qu'un livre peut être emprunté par 0 ou plusieurs plusieurs étudiants
Appartenir	(0.N : 1.1)	1 livre ne peut appartenir qu'à un et un seul genre par contre un genre peut contenir 0 ou plusieurs livres

2. Création de la Base de données et des tables

Suivant le rendu de notre MCD présenté dans la documentation technique, nous avons créé une base de données “**Bibliothèque**” ; des tables à base du langage SQL et nous leur avons attribué les caractéristiques suivantes.

a. Table Livres

Caractéristiques	Nom du champ	Type	NULL?
Identifiant du livre	id_livre	AUTO_INCREMENT	Non
titre du livre	titre	VARCHAR (50)	Non
auteur du livre	auteur	VARCHAR (50)	Non
année de publication du livre	année_public	INT	Non
nombre d'exemplaire du livre	nbr_exemplaire	INT (25)	Non
identifiant du genre du livre	id_genre_genres	INT (25)	Non

b. Table genre

Caractéristiques	Nom du champ	Type	NULL?
Identifiant du genre	id_genre	AUTO_INCREMENT	Non
Nom du genre	nom_genre	VARCHAR (50)	Non

c. Table étudiants

Caractéristiques	Nom du champ	Type	NULL?
Identifiant de l'étudiant	id_etudiant	AUTO_INCREMENT	Non
nom de l'étudiant	nom	VARCHAR (25)	Non
prénom de l'étudiant	prénom	VARCHAR (25)	Non
date de naissance de l'étudiant	date_naiss	DATE	Non
adresse email de l'étudiant	email	VARCHAR (50)	Non
téléphone de l'étudiant	tel	VARCHAR (50)	Non
date d'inscription de l'étudiant	date_inscription	DATE	Non

d. Table emprunts

Caractéristiques	Nom du champ	Type	NULL?
Date de l'emprunt	date_empt	DATE	Non
Date de retour de l'emprunt	date_retour	DATE	Non
état du retour du livre	retour	BOLL	Non
identifiant du livre	id_livre_livres	AUTO_INCREMENT	Non
identifiant de l'emprunteur	id_etudiant_étudiants	AUTO_INCREMENT	Non

3. Ajout des données

La création de nos différentes tables étant faite, nous avons pour une opérationnalisation de notre base de données inséré une série de données pour tester son bon fonctionnement.

Chacune de ces données remplit pleinement les caractéristiques que nous avons définies dans les tables ci-dessus. Nous avons respectivement dans notre bibliothèque à l'heure actuelle :

- 06 Genres
- 10 étudiants inscrits
- 19 livres
- 15 emprunts déjà enregistrés dont 7 retours de livres effectifs et 8 en cours.

4. Création des requêtes

Notre système en plus de gérer les emprunts aide le bibliothécaire à émettre des rapports et à faire un bilan de gestion ceci à travers des requêtes qu'il envoie au système.

Nous avons prévu des requêtes pour :

- ★ Trouver les étudiants ayant emprunté un certain livre
- ★ Compter le nombre total d'emprunts par livre
- ★ Lister les livres d'un certain genre

5. Définition des fonctions et procédures de notre système

Une fonction est un code qui effectue une série d'instructions bien précises, et renvoie le résultat de ces instructions.

En matière de fonctions et de procédures, notre système prend en compte :

- Une procédure d'ajout d'un emprunt
- Une fonction de vérification de la disponibilité d'un livre

6. Mise en place des vues et triggers

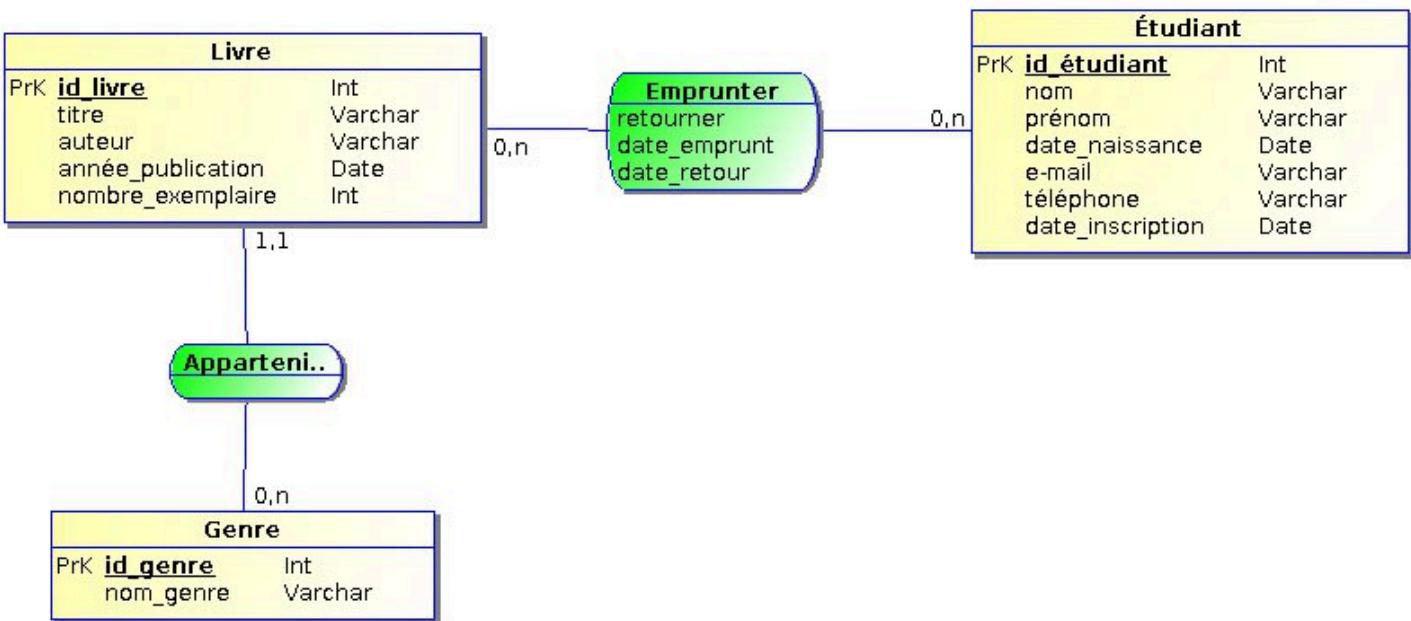
Les vues sont des objets de la base de données, constitués d'un nom, et d'une requête de sélection. Une fois qu'une vue est définie, on peut l'utiliser comme on le ferait avec une table ; table qui serait constituée des données sélectionnées par la requête définissant la vue. Pour notre projet nous avons défini une vue pour **afficher les détails d'un emprunt**.

Les triggers (ou déclencheurs) sont des objets de la base de données. Attachés à une table, ils vont déclencher l'exécution d'une instruction, ou d'un bloc d'instructions, lorsqu'une, ou plusieurs lignes sont insérées, supprimées ou modifiées dans la table à laquelle ils sont attachés. Pour notre projet nous avons défini un trigger pour **actualiser l'état du stock des livres quand un emprunt est achevé (pris et rendu)**

Résultats

Dans cette section vous trouverez pour chaque étape de notre projet des captures d'écran montrant l'ensemble de nos actions sous forme de scripts SQL et le rendu lors de l'implémentation dans MySQL

1. Choix des entités et définition des associations



2. Création de la base de données et des tables

```
#Ma base de données :
```

```
#-----  
# 1. database creation  
#-----
```

```
drop database if exists bibliotheque;  
CREATE DATABASE bibliotheque CHARACTER SET 'utf8';  
USE bibliotheque;
```

```
#-----  
# 2. tables creation  
#-----
```

```
DROP TABLE IF EXISTS livres;  
CREATE TABLE livres(  
    id_livre      int (50) not null,  
    titre        Varchar (50) not null,  
    auteur       Varchar (50) not null,  
    année_public   int not null,  
    nbr_exemplaire  Int (25) not null,  
    id_genre_genres  Int (25) not null,  
    PRIMARY KEY (id_livre)  
)ENGINE=InnoDB;
```

```
DROP TABLE IF EXISTS genres;  
CREATE TABLE genres(  
    id_genre      int (50) not null,  
    nom_genre     Varchar (50) not null,  
    PRIMARY KEY (id_genre)  
)ENGINE=InnoDB;
```

```
DROP TABLE IF EXISTS étudiants;  
CREATE TABLE étudiants(  
    id_etudiant    int (50) not null,  
    nom          Varchar (25) not null,  
    prénom        Varchar (25) not null,  
    date_naiss    Date not null,  
    email         Varchar (50) not null,  
    tel           Varchar (50) not null,  
    date_inscription Date not null,  
    PRIMARY KEY (id_etudiant)  
)ENGINE=InnoDB;
```

```
DROP TABLE IF EXISTS emprunts;  
CREATE TABLE emprunts(  
    date_empt     Date not null,  
    date_retour   Date not null,  
    retour        Bool not null,  
    id_livre_livres  int (50) not null,  
    id_etudiant_étudiants  int (50) not null,  
    PRIMARY KEY (id_livre_livres,id_etudiant_étudiants)  
)ENGINE=InnoDB;
```

```
+-----+  
| *****CREATION DE LA BASE DE DONNEES***** |  
+-----+  
| *****CREATION DE LA BASE DE DONNEES***** |  
+-----+  
1 row in set (0,00 sec)  
  
Query OK, 5 rows affected (1,66 sec)  
  
Query OK, 1 row affected, 1 warning (0,14 sec)  
  
Database changed  
  
Database changed  
+-----+  
| *****CREATION DES TABLES***** |  
|  
+-----+  
| *****CREATION DES TABLES***** |  
|  
+-----+  
1 row in set (0,00 sec)  
  
Query OK, 0 rows affected, 1 warning (0,09 sec)  
  
Query OK, 0 rows affected, 3 warnings (1,48 sec)  
  
Query OK, 0 rows affected, 1 warning (0,06 sec)  
  
Query OK, 0 rows affected, 1 warning (0,67 sec)  
  
Query OK, 0 rows affected, 1 warning (0,50 sec)  
  
Query OK, 0 rows affected, 1 warning (1,17 sec)  
  
Query OK, 0 rows affected, 1 warning (0,10 sec)  
  
Query OK, 0 rows affected, 2 warnings (1,81 sec)  
  
Query OK, 0 rows affected, 1 warning (3,50 sec)  
Records: 0 Duplicates: 0 Warnings: 1  
  
Query OK, 0 rows affected, 1 warning (3,72 sec)  
Records: 0 Duplicates: 0 Warnings: 1  
  
Query OK, 0 rows affected, 1 warning (5,13 sec)  
Records: 0 Duplicates: 0 Warnings: 1  
  
Query OK, 0 rows affected, 1 warning (4,69 sec)  
Records: 0 Duplicates: 0 Warnings: 1
```

3. Ajout des données

```
#-----#
# 3. inserting data in tables
#-----#  
  
INSERT INTO genres(nom_genre)
VALUES("Fiction"),
      ("Christian"),
      ("Adventure"),
      ("romances"),
      ("mystery"),
      ("Whitchcraft");  
  
INSERT INTO étudiants(nom, prénom, date_naiss, email,tel, date_inscription )
VALUES("Kamga","kotopi","1997-10-23", "kotopi@gmail.com",
"698741234", "2022-03-05"),
      ("Siewe","Mignon", "1999-05-23", "Mignon@gmail.com",
"691547834", "2022-10-15"),
      ("AbesooLo","Hughes", "1993-10-03", "Hughes@gmail.com",
"698851234", "2022-03-13"),
      ("Tema","Landry", "1992-11-23", "Tema@gmail.com", "69941234", "2022-02-01"),
      ("Atagoun","Eric", "1999-02-02", "eric@gmail.com",
"694722134", "2022-02-21"),
      ("Alioum","Alfred", "1997-02-03", "alfred@gmail.com",
"692747834", "2022-07-10"),
      ("Satching","le", "1997-03-04", "sule@gmail.com",
"693794134", "2022-04-04"),
      ("Tenor","Ellie", "1997-07-14", "ellie@gmail.com",
"691789634", "2022-01-21"),
      ("Kella","Joyce", "1997-08-07", "kella@gmail.com",
"694754534", "2022-05-08"),
      ("Tegang","karlyne", "1997-10-12", "karlyne@gmail.com",
"693713134", "2022-07-27");  
  
INSERT INTO livres(uteur, titre, année_public, nbr_exemplaire,id_genre_genres)
VALUES("Jean-Jacques Rousseau", "The Confessions", 1981, 60, 2),
      ("Erich Fromm", "The Art of Loving", 1956, 14, 3),
      ("Mark Twain", "The Adventures of Huckleberry Finn", 1884, 28, 4),
      ("James Thurber", "My Life and Hard Times", 1933, 11, 5),
      ("William Shakespeare", "Romeo and Juliet", 1996, 14, 1),
      ("John Barth", "The End of the Road", 1958, 20, 1),
      ("Aldous Huxley", "Brave New World", 1932, 19, 2),
      ("Fritz Peters", "Boyhood with Gurdjieff", 1964, 17, 2),
      ("P.D. Ouspensky", "In Search of the Miraculous", 1949, 38, 1),
      ("Johann Sebastian Bach", "an Introduction to his Life and Works", 1962, 16,
1),
      ("Hunter Davies", "The Beatles", 1968, 34, 2),
      ("P.D. Ouspensky", "The Strange Life of Ivan Osokin", 1955, 16, 1),
      ("René Daumal", "Mount Analogue", 1952, 10, 2),
      ("Bernard Malamud", "The Assistant", 1957, 19, 2),
      ("C.G. Jung", "Memories Dreams Reflections", 1961, 35, 1),
      ("Philip Roth", "Goodbye, Columbus", 1959, 9, 3),
      ("Malcolm X and Alex Haley", "Autobiography", 1964, 4, 3),
      ("Jule Eisenbud, M.D.", "The World of Ted Serios", 1968, 3, 1),
      ("Penelope Gilliatt", "A State of Change", 1967, 22, 3);
```

```

INSERT INTO emprunts(date_empt, date_retour, retour, id_livre_livres,
id_etudiant étudiants)
VALUES("2024-06-24", "2024-07-28", false, 2 ,3),
      ("2024-06-21", "2024-06-23", TRUE, 4 ,8),
      ("2024-03-25", "2024-03-28", TRUE, 5 ,6),
      ("2024-06-23", "2024-06-24", TRUE, 3 ,7),
      ("2024-06-20", "2024-06-25", TRUE, 9 ,6),
      ("2024-06-12", "2024-08-14", false, 1 ,7),
      ("2024-06-04", "2024-010-06",false, 2 ,9),
      ("2024-06-01", "2024-11-12", false, 1 ,3),
      ("2024-06-02", "2024-8-10", false, 2 ,1),
      ("2024-06-05", "2024-09-10", false, 2 ,2),
      ("2024-06-12", "2024-06-15", TRUE, 3 ,5),
      ("2024-06-04", "2024-06-05", TRUE, 7 ,7),
      ("2024-06-11", "2024-08-18", false, 3 ,9),
      ("2024-06-22", "2024-06-28", TRUE, 2 ,10),
      ("2024-06-21", "2024-07-28", false, 5 ,9);

```

```

mysql> SELECT*FROM livres;
+-----+-----+-----+-----+-----+-----+
| id_livre | titre | auteur | année_public | nbr_exemplaire | id_genre_genres |
+-----+-----+-----+-----+-----+-----+
| 1 | The Confessions | Jean-Jacques Rousseau | 1981 | 60 | 2 |
| 2 | The Art of Loving | Erich Fromm | 1956 | 14 | 3 |
| 3 | The Adventures of Huckleberry Finn | Mark Twain | 1884 | 28 | 4 |
| 4 | My Life and Hard Times | James Thurber | 1933 | 11 | 5 |
| 5 | Romeo and Juliet | William Shakespeare | 1996 | 14 | 1 |
| 6 | The End of the Road | John Barth | 1958 | 20 | 1 |
| 7 | Brave New World | Aldous Huxley | 1932 | 19 | 2 |
| 8 | Boyhood with Gurdjieff | Fritz Peters | 1964 | 17 | 2 |
| 9 | In Search of the Miraculous | P.D. Ouspensky | 1949 | 38 | 1 |
| 10 | an Introduction to his Life and Works | Johann Sebastian Bach | 1962 | 16 | 1 |
| 11 | The Beatles | Hunter Davies | 1968 | 34 | 2 |
| 12 | The Strange Life of Ivan Osokin | P.D. Ouspensky | 1955 | 16 | 1 |
| 13 | Mount Analogue | René Daumal | 1952 | 10 | 2 |
| 14 | The Assistant | Bernard Malamud | 1957 | 19 | 2 |
| 15 | Memories Dreams Reflections | C.G. Jung | 1961 | 35 | 1 |
| 16 | Goodbye, Columbus | Philip Roth | 1959 | 9 | 3 |
| 17 | Autobiography | Malcolm X and Alex Haley | 1964 | 4 | 3 |
| 18 | The World of Ted Serios | Jule Eisenbud, M.D. | 1968 | 3 | 1 |
| 19 | A State of Change | Penelope Gilliatt | 1967 | 22 | 3 |
+-----+-----+-----+-----+-----+-----+
19 rows in set (0,00 sec)

```

```

mysql> SELECT*FROM étudiants;
+-----+-----+-----+-----+-----+-----+-----+
| id_etudiant | nom | prénom | date_naiss | email | tel | date_inscription |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | Kamga | kotopi | 1997-10-23 | kotopi@gmail.com | 698741234 | 2022-03-05 |
| 2 | Siewe | Mignon | 1999-05-23 | Mignon@gmail.com | 691547834 | 2022-10-15 |
| 3 | Abesoozo | Hughes | 1993-10-03 | Hughes@gmail.com | 698851234 | 2022-03-13 |
| 4 | Tema | Landry | 1992-11-23 | Tema@gmail.com | 69941234 | 2022-02-01 |
| 5 | Atagoun | Eric | 1999-02-02 | eric@gmail.com | 694722134 | 2022-02-21 |
| 6 | Alioum | Alfred | 1997-02-03 | alfred@gmail.com | 692747834 | 2022-07-10 |
| 7 | Satching | le | 1997-03-04 | sule@gmail.com | 693794134 | 2022-04-04 |
| 8 | Tenor | Ellie | 1997-07-14 | ellie@gmail.com | 691789634 | 2022-01-21 |
| 9 | Kella | Joyce | 1997-08-07 | kella@gmail.com | 694754534 | 2022-05-08 |
| 10 | Tegang | karlyne | 1997-10-12 | karlyne@gmail.com | 693713134 | 2022-07-27 |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0,00 sec)

```

```
mysql> SELECT*FROM emprunts;
+-----+-----+-----+-----+-----+
| date_empt | date_retour | retour | id_livre_livres | id_etudiant_étudiants |
+-----+-----+-----+-----+
| 2024-06-01 | 2024-11-12 | 0 | 1 | 3 |
| 2024-06-12 | 2024-08-14 | 0 | 1 | 7 |
| 2024-06-02 | 2024-08-10 | 0 | 2 | 1 |
| 2024-06-05 | 2024-09-10 | 0 | 2 | 2 |
| 2024-06-24 | 2024-07-28 | 0 | 2 | 3 |
| 2024-06-04 | 2024-10-06 | 0 | 2 | 9 |
| 2024-06-22 | 2024-06-28 | 1 | 2 | 10 |
| 2024-06-12 | 2024-06-15 | 1 | 3 | 5 |
| 2024-06-23 | 2024-06-24 | 1 | 3 | 7 |
| 2024-06-11 | 2024-08-18 | 0 | 3 | 9 |
| 2024-06-21 | 2024-06-23 | 1 | 4 | 8 |
| 2024-03-25 | 2024-03-28 | 1 | 5 | 6 |
| 2024-06-21 | 2024-07-28 | 0 | 5 | 9 |
| 2024-06-04 | 2024-06-05 | 1 | 7 | 7 |
| 2024-06-20 | 2024-06-25 | 1 | 9 | 6 |
+-----+-----+-----+-----+
```

```
mysql> SELECT*FROM genres;
+-----+-----+
| id_genre | nom_genre |
+-----+-----+
| 1 | Fiction |
| 2 | Christian |
| 3 | Adventure |
| 4 | romances |
| 5 | mystery |
| 6 | Whitchcraft |
+-----+-----+
6 rows in set (0,00 sec)
```

4. Création des requêtes

```
#-----#
# 5. various sql queries
#-----#

# requête pour trouver les étudiants ayant emprunter un certain livre
SELECT e.id_etudiant, e.nom, e.prenom, e.date_naiss, e.email, e.tel, e.date_inscription
FROM étudiants e JOIN emprunts em ON e.id_etudiant = em.id_etudiant_étudiants JOIN livres
l ON em.id_livre_livres = l.id_livre WHERE l.id_livre = 3;
# Requête pour lister les emprunts en cours d'un étudiant
SELECT em.date_empt, em.date_retour, em.retour, l.titre, l.auteur FROM emprunts em JOIN
livres l ON em.id_livre_livres = l.id_livre WHERE em.id_etudiant_étudiants = 3 AND
em.retour = FALSE;

# Requête pour compter le nombre total d_emprunts par livre
SELECT l.id_livre, l.titre, l.auteur, COUNT(em.id_etudiant_étudiants &&
em.id_livre_livres) AS nombre_emprunts FROM livres l LEFT JOIN emprunts em ON l.id_livre =
em.id_livre_livres GROUP BY l.id_livre, l.titre, l.auteur ORDER BY `l`.`id_livre` ASC;

# requête pour lister les livre d'un certain genre
SELECT * FROM livres WHERE id_genre_genres = 2;

ALTER TABLE livres ADD CONSTRAINT FK_livres_id_genre_genres FOREIGN KEY (id_genre_genres)
REFERENCES genres(id_genre);
ALTER TABLE emprunts ADD CONSTRAINT FK_emprunts_id_livre_livres FOREIGN KEY
(id_livre_livres) REFERENCES livres(id_livre);
ALTER TABLE emprunts ADD CONSTRAINT FK_emprunts_id_etudiant_étudiants FOREIGN KEY
(id_etudiant_étudiants) REFERENCES étudiants(id_etudiant);
```

```
+-----+
| *****requête pour trouver les étudiants ayant emprunter un certain livre*****|
+-----+
| *****requête pour trouver les étudiants ayant emprunter un certain livre*****|
+-----+
1 row in set (0,00 sec)

+-----+-----+-----+-----+-----+-----+
| id_etudiant | nom      | prénom   | date_naiss | email        | tel       | date_inscription |
+-----+-----+-----+-----+-----+-----+
|      5 | Atagoun | Eric     | 1999-02-02 | eric@gmail.com | 694722134 | 2022-02-21
|      7 | Satching | le       | 1997-03-04 | sule@gmail.com | 693794134 | 2022-04-04
|      9 | Kella    | Joyce    | 1997-08-07 | kella@gmail.com | 694754534 | 2022-05-08
+-----+-----+-----+-----+-----+-----+
3 rows in set (0,01 sec)
```

```

+-----+
| *****Requête pour lister les emprunts en cours d'un étudiant*****
|
+-----+
| *****Requête pour lister les emprunts en cours d'un étudiant*****
|
+-----+
1 row in set (0,00 sec)

+-----+-----+-----+-----+
| date_empt | date_retour | retour | titre           | auteur
+-----+-----+-----+-----+
| 2024-06-01 | 2024-11-12 |     0 | The Confessions | Jean-Jacques Rousseau
| 2024-06-24 | 2024-07-28 |     0 | The Art of Loving | Erich Fromm
+-----+-----+-----+-----+
2 rows in set (0,00 sec)

```

```

+-----+
| *****Requête pour compter le nombre total d_emprunts par livre*****
|
+-----+
| *****Requête pour compter le nombre total d_emprunts par livre*****
|
+-----+
1 row in set (0,00 sec)

+-----+-----+-----+
| id_livre | titre           | auteur          | nombre_emprunts |
+-----+-----+-----+
| 1 | The Confessions | Jean-Jacques Rousseau | 2
| 2 | The Art of Loving | Erich Fromm      | 5
| 3 | The Adventures of Huckleberry Finn | Mark Twain    | 3
| 4 | My Life and Hard Times | James Thurber | 1
| 5 | Romeo and Juliet | William Shakespeare | 2
| 6 | The End of the Road | John Barth      | 0
| 7 | Brave New World | Aldous Huxley   | 1
| 8 | Boyhood with Gurdjieff | Fritz Peters | 0
| 9 | In Search of the Miraculous | P.D. Ouspensky | 1
| 10 | an Introduction to his Life and Works | Johann Sebastian Bach | 0
| 11 | The Beatles | Hunter Davies | 0
| 12 | The Strange Life of Ivan Osokin | P.D. Ouspensky | 0
| 13 | Mount Analogue | René Daumal | 0
| 14 | The Assistant | Bernard Malamud | 0
| 15 | Memories Dreams Reflections | C.G. Jung | 0
| 16 | Goodbye, Columbus | Philip Roth | 0
| 17 | Autobiography | Malcolm X and Alex Haley | 0
| 18 | The World of Ted Serios | Jule Eisenbud, M.D. | 0
| 19 | A State of Change | Penelope Gilliatt | 0
+-----+-----+-----+
19 rows in set, 1 warning (0,00 sec)

```

```

+-----+
| *****requête pour lister les livre d'un certain genre*****
|
+-----+
| *****requête pour lister les livre d'un certain genre*****
|
+-----+
1 row in set (0,00 sec)

+-----+-----+-----+-----+-----+-----+
| id_livre | titre           | auteur          | année_public | nbr_exemplaire | id_genre_genres |
+-----+-----+-----+-----+-----+-----+
| 1 | The Confessions | Jean-Jacques Rousseau | 1981 | 60 | 2
| 7 | Brave New World | Aldous Huxley   | 1932 | 19 | 2
| 8 | Boyhood with Gurdjieff | Fritz Peters | 1964 | 17 | 2
| 11 | The Beatles | Hunter Davies | 1968 | 34 | 2
| 13 | Mount Analogue | René Daumal | 1952 | 10 | 2
| 14 | The Assistant | Bernard Malamud | 1957 | 19 | 2
+-----+-----+-----+-----+-----+-----+
6 rows in set (0,00 sec)

Query OK, 19 rows affected (4,59 sec)
Records: 19  Duplicates: 0  Warnings: 0

Query OK, 15 rows affected (4,04 sec)
Records: 15  Duplicates: 0  Warnings: 0

Query OK, 15 rows affected (4,44 sec)
Records: 15  Duplicates: 0  Warnings: 0

```

5. Définition des fonctions et procédures de notre système

```
#-----#
# 4. procedure and function
#-----#

# procedure to add a borrow

drop procedure if exists addEmprunt;
delimiter |
CREATE PROCEDURE addEmprunt (in livre_emprunte INT, etudiant_emprunteur INT, in days INT)
BEGIN
    INSERT INTO emprunts (date_empt,
                          date_retour,
                          retour,
                          id_livre_livres,
                          id_etudiant_étudiants)
        VALUES (NOW(),DATE_ADD(NOW(), INTERVAL days day), false, livre_emprunte,
                etudiant_emprunteur);
END |
delimiter ;

# function to verify the availability of a book
drop function if exists isBook;
delimiter |
CREATE function isBook (id_livre_recherche int)
returns boolean
deterministic
BEGIN
    DECLARE response boolean;
    DECLARE result int;

    SELECT nbr_exemplaire INTO result FROM livres WHERE id_livre = id_livre_recherche;
    IF (result > 0) THEN
        set response = true;
    ELSE
        set response = false;
    END IF;
    return response;
END |
delimiter ;
```

```
+-----+
| *****CREATION PROCEDURE to add a borrow*****
|
+-----+
| *****CREATION PROCEDURE to add a borrow*****
|
+-----+
1 row in set (0,00 sec)

Query OK, 0 rows affected, 1 warning (0,12 sec)

Query OK, 0 rows affected (0,29 sec)

Query OK, 0 rows affected, 1 warning (0,12 sec)

Query OK, 0 rows affected (0,32 sec)
```

6. Mise en place des vues et triggers

```
#-----#
# 5. view and trigger to update livres at borrow and return
#-----#

# 1. view to display emprunts details
#-----#
drop view if exists empruntDetails;
create view empruntDetails as
select titre, nom, date_empt, date_retour from emprunts inner join
livres on id_livre = id_livre_livres inner join
étudiants on id_etudiant = id_etudiant_étudiants;

# calling the view
select * from empruntDetails order by date_empt;
#-----#
# 2. tirgger update book stock when a book is borrow and return
#-----#
# a. when a book is borrowed
drop trigger if exists reduceBook;
delimiter //
create trigger reduceBook
after insert on emprunts
for each row
begin
    update livres set nbr_exemplaire = nbr_exemplaire - 1
    where id_livre = new.id_livre_livres;
end //
delimiter ;

# b. when a book is returned
drop trigger if exists addBook;
delimiter //
create trigger addBook
after update on emprunts
for each row
begin
    update livres set nbr_exemplaire = nbr_exemplaire + 1
    where id_livre = new.id_livre_livres;
end //
delimiter ;
```

```

+-----+
| *****VIEW AND TRIGGER*****+
|
+-----+
| *****VIEW AND TRIGGER*****+
|
+-----+
1 row in set (0,00 sec)

+-----+
| *****View to display emprunts details*****+
|
+-----+
| *****View to display emprunts details*****+
|
+-----+
1 row in set (0,00 sec)

Query OK, 0 rows affected, 1 warning (0,99 sec)

Query OK, 0 rows affected (0,38 sec)

+-----+-----+-----+-----+
| titre | nom | date_empt | date_retour |
+-----+-----+-----+-----+
| Romeo and Juliet | Alioum | 2024-03-25 | 2024-03-28 |
| The Confessions | Abesooolo | 2024-06-01 | 2024-11-12 |
| The Art of Loving | Kamga | 2024-06-02 | 2024-08-10 |
| The Art of Loving | Kella | 2024-06-04 | 2024-10-06 |
| Brave New World | Satching | 2024-06-04 | 2024-06-05 |
| The Art of Loving | Siewe | 2024-06-05 | 2024-09-10 |
| The Adventures of Huckleberry Finn | Kella | 2024-06-11 | 2024-08-18 |
| The Confessions | Satching | 2024-06-12 | 2024-08-14 |
| The Adventures of Huckleberry Finn | Atagoun | 2024-06-12 | 2024-06-15 |
| In Search of the Miraculous | Alioum | 2024-06-20 | 2024-06-25 |
| My Life and Hard Times | Tenor | 2024-06-21 | 2024-06-23 |
| Romeo and Juliet | Kella | 2024-06-21 | 2024-07-28 |
| The Art of Loving | Tegang | 2024-06-22 | 2024-06-28 |
| The Adventures of Huckleberry Finn | Satching | 2024-06-23 | 2024-06-24 |
| The Art of Loving | Abesooolo | 2024-06-24 | 2024-07-28 |
+-----+-----+-----+-----+
15 rows in set (0,00 sec)

```

```

+-----+
| *****Trigger update book stock*****+
|
+-----+
| *****Trigger update book stock*****+
|
+-----+
1 row in set (0,00 sec)

Query OK, 0 rows affected, 1 warning (0,08 sec)

Query OK, 0 rows affected (0,21 sec)

Query OK, 0 rows affected, 1 warning (0,13 sec)

Query OK, 0 rows affected (0,64 sec)

```

CONCLUSION

À travers notre base de données bibliothèque que nous avons créé, nous pouvons dire que la modernisation de notre bibliothèque va bon train. Nous avons pu dématérialiser la grande majorité des procédures au sein de la bibliothèque.

En utilisant notre système, nous pouvons avoir une traçabilité des actions menées au sein de la bibliothèque et nous pouvons gérer plusieurs générations d'apprenants.