Projet Bibliothèque Hybride – Spring Boot + REST + SOAP

Membres :

Amadou Tidiane DIOP ( N01220420191 ) et Thierno Sadou BAH ( N04347420191 )

Ce projet est une application de gestion de bibliothèque hybride développée avec Spring Boot, combinant :  
- Une API RESTful pour la gestion des réservations  
- Un service SOAP simulant l'action du bibliothécaire (preterLivre)  
- Une base de données en mémoire H2

# Fonctionnalités principales

- Lister tous les livres   
- Lister les livres disponibles  
- Réserver un livre  
- Lors d’une réservation, appel automatique du service SOAP  
- Stockage en mémoire (H2)



1. On le place dans le workspace eclipse



1. On le place dans document car son chemin d’acces servira plutard pour l’execution dans l’invite de commande

# Configuration

- Port HTTP : 8181  
- Base H2 : en mémoire   
- Langage : Java 17+  
- Framework : Spring Boot 3.x  
- Dépendances clés : Spring Web, Spring Data JPA, Spring WS, H2 Database

# Lancer le projet

1. Ouvrir dans Eclipse IDE for Enterprise Java and Web Developers  
2. Clic droit sur BibliothequeHybrideApplication.java → Run As > Java Application  
3. Accéder à l’API via : http://localhost:8181

# Tester l’API

Console H2 :

- URL : http://localhost:8181/h2-console  
- JDBC URL : jdbc:h2:mem:biblio  
- Username : sa  
- Password : (laisser vide)

Ajouter des livres :

INSERT INTO livre (titre, auteur, disponible) VALUES

('Spring Boot en pratique', 'Khadidia Sy', true),

('Java Moderne', 'Amadou Tidiane', true),

('Clean Code', 'Robert C. Martin', true),

('Hibernate avancé', 'Gavin King', true),

('API REST simplifiée', 'Ada Lovelace', true),

('React pour les Nuls', 'Dan Abramov', true),

('Ux design', 'Thierno Sadou BAH', true);

Supprimer un livre :

DELETE FROM livre WHERE id = 1; l’id dependra du livre a supprime

Mettre un livre en indisponible pour simuler un prêt :

UPDATE livre SET disponible = false WHERE id = 1; l’id dependra du livre a rendre indisponible

Annuler une reservation ou rendre disponible :

UPDATE livre SET disponible = true WHERE id = 1; l’id dependra du livre

Pour l’interface Graphique :

**On lance le CMD ensuite on rentre le chemin d’acces du dossier** **frontend\_bibliotheque ( pour ma part le chemin d’acces est celui-ci dessous )**

cd C:\Users\diop1\OneDrive\Documents\frontend\_bibliotheque\frontend

npm run dev

**Sur le navigatuer pour l’interface graphique on tape ;**

localhost:3000

POSTMAN :

1. Récupérer la liste de tous les livres

GET <http://localhost:8181/livres>

1. Afficher les informations d’un livre (par ID)

GEThttp://localhost:8181/livres/4

1. Retourner les livres disponibles (non prêtés et non réservés)

GET <http://localhost:8181/livres/disponibles>

1. Suivi d’une réservation (par ID)

GET http://localhost:8181/reservations/3

1. Réserver un livre (appel REST → SOAP)

POST <http://localhost:8181/reservations>

Body (exemple pour livre ID 4)

{

"userId": "Tidiane",

"livreId": 4,

"dateDebut": "2025-06-28",

"dateFin": "2025-07-07"

}

* Pour modifier un livre

PUT <http://localhost:8181/livres/4>

Body (exemple pour livre ID 4)

{

"id": 4,

"titre": "Hibernate avancé – la modification",

"auteur": "Gavin King",

"disponible": false

}

* Pour supprimer un livre

DELETE <http://localhost:8181/livres/4> id au choix

# Remarques

- Les livres déjà réservés (non disponibles) ne peuvent plus être réservés.