

## Prerequisites:

- **Java 8+**
- **Maven** for dependency management
- **H2 Database** (embedded or local file-based database)
- **IDE** ( IntelliJ IDEA)

## API Endpoints:

**Token Required for Each Request**

**Authorization: Bearer your-jwt-token-here**

### Book Management:

#### 1. Get All Books

**URL:** /api/books

**Method:** GET

#### 2. Get a Book by ID

**URL:** /api/books/{id}

**Method:** GET

#### 3. Add a New Book

**URL:** /api/books

**Method:** POST

*Request Body:*

```
json
{
  "title": "Spring Boot Project",
  "author": "Lana Baradan",
  "publicationYear": 2025,
  "isbn": "123-456789124"
}
```

#### 4. Update an Existing Book

**URL:** /api/books/{id}

**Method:** PUT

*Request Body:*

```
json
{
  "title": "Spring Boot Project",
  "author": "Lana Baradan",
  "publicationYear": 2025,
  "isbn": "123-456789124"
}
```

## 5. Delete a Book

**URL:** /api/books/{id}

**Method:** DELETE

## Borrowing Record Management:

### 1. Borrow a Book

**URL:** /api/borrow/{bookId}/patron/{patronId}

**Method:** POST

### 2. Return a Borrowed Book

**URL:** /api/return/{bookId}/patron/{patronId}

**Method:** PUT

## Patron Management

### 1. Get All Patrons

**URL:** /api/patrons

**Method:** GET

### 2. Get Patron by ID

**URL:** /api/patrons/{id}

**Method:** GET

### 3. Create a New Patron

**URL:** /api/patrons

**Method:** POST

*Request Body:*

```
json
{
  "name": "John Doe",
  "email": "john@example.com",
  "phoneNumber": 9987654567
}
```

### 4. Update an Existing Patron

**URL:** /api/patrons/{id}

**Method:** PUT

*Request Body:*

```
json
{
  "name": "John Doe",
  "email": "john@example.com",
  "phoneNumber": 9987654567
}
```

### 5. Delete a Patron

**URL:** /api/patrons/{id}

**Method:** DELETE

## Authentication Using JWT

### AuthController :

#### 1. User Login

**URL:** /api/auth/login

**Method:** POST

*Request Body:*

```
json
{
  "username": "baraa baradan",
  "password": "Password@123"
}
```

```
}
```

## 2. User Registration

**URL:** /api/auth/register

**Method:** POST

*Request Body:*

```
json
{
  "username": "baraa baradan",
  "password": "Password@123"
}
```