

# Library Management System

## API

Gradle version is 8.5

JAVA version 17.0.6

## Integrate API

Git clone [https://github.com/zoalfikar/library\\_managment\\_system.git](https://github.com/zoalfikar/library_managment_system.git)

Cd library\_managment\_system

Gradle build OR .\gradlew build

## Setup

Connecting with mysql

Run mysql server **net start mysql** or **net start mysql80** according to mysql version .

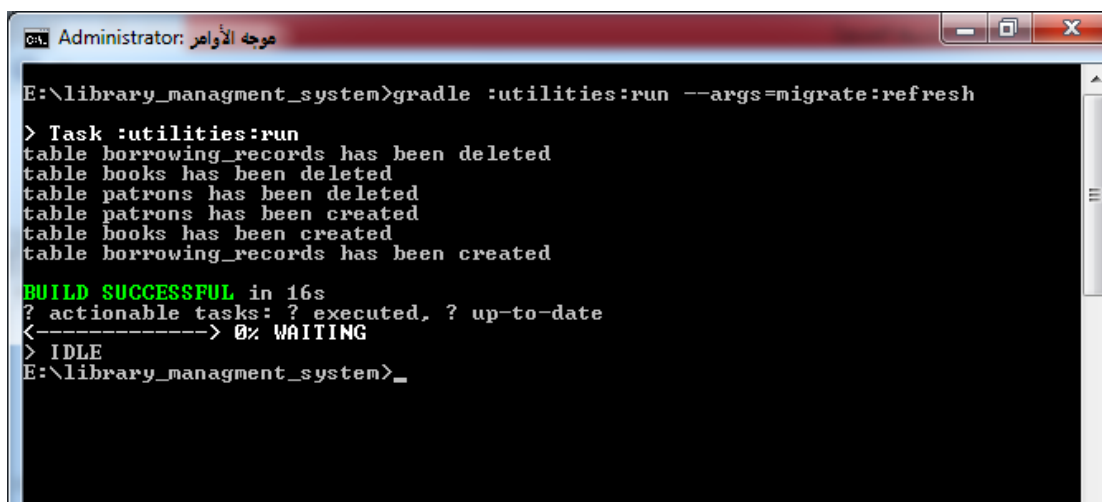
Create new database .

Set up the **application.properties** add [ *datasource.username* and *datasource.password* and *database.name*].

**gradle :utilities:run --args=migrate**

or

**gradle :utilities:run --args=migrate:refresh**



```
C:\A. Administrator: موجه الأوامر

E:\library_managment_system>gradle :utilities:run --args=migrate:refresh

> Task :utilities:run
table borrowing_records has been deleted
table books has been deleted
table patrons has been deleted
table patrons has been created
table books has been created
table borrowing_records has been created

BUILD SUCCESSFUL in 16s
? actionable tasks: ? executed, ? up-to-date
<-----> 0% WAITING
> IDLE
E:\library_managment_system>_
```

to Create a migration Go to utilities\src\main\java\migrations\tables add file {migration\_name}.migration .

gradle :utilities:run --args=migrate:triggers

```
E:\library_managment_system>gradle :utilities:run --args=migrate:triggers
> Task :utilities:run
Trigger book_availiable has been deleted
Trigger book_availiable has been created
BUILD SUCCESSFUL in 6s
? actionable tasks: ? executed, ? up-to-date
E:\library_managment_system>
```

gradle :utilities:run --args=seed:library

```
> Task :utilities:run
insert publication_year = 1926 into books
insert author = Agatha Christie into books
insert isbn = 1111 into books
insert title = The Murder of Roger Ackroyd into books
data inserted
insert publication_year = 1932 into books
insert author = Agatha Christie into books
insert isbn = 2222 into books
insert title = Peril at End House into books
data inserted
insert publication_year = 3333 into books
insert author = Agatha Christie into books
insert isbn = 3333 into books
insert title = Murder on the Orient Express into books
data inserted
insert publication_year = 1935 into books
insert author = Agatha Christie into books
insert isbn = 4444 into books
insert title = The ABC Murders into books
data inserted
insert publication_year = 1393 into books
insert author = Agatha Christie into books
insert isbn = 5555 into books
insert title = And Then There Were None into books
data inserted
insert publication_year = 1943 into books
insert author = Agatha Christie into books
insert isbn = 6666 into books
insert title = Five Little Pigs into books
data inserted
insert publication_year = 1949 into books
insert author = Agatha Christie into books
insert isbn = 7777 into books
insert title = Crooked House into books
data inserted
insert publication_year = 1862 into books
insert author = Victor Hugo into books
insert isbn = 8888 into books
insert title = Les Misérables into books
data inserted
insert publication_year = 1925 into books
insert author = F. Scott Fitzgerald into books
```

To Create a seeder go to utilities\src\main\java\migrations\seeds {seed\_name}.json .

Start application

Gradle bootrun

## Testing th Api

Gradle test

## Useage

```
curl http://localhost:7788
```

```
{"message":"welcome in library managment system"}
```

## APIs

### CURD Patron

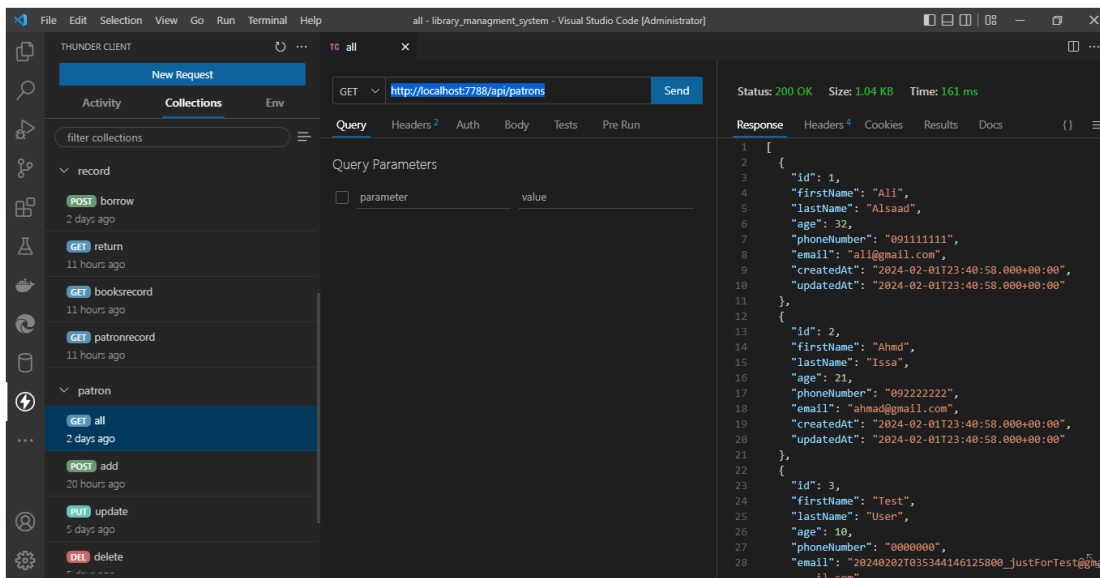
get [api/patrons](#) all patrons

get [api/patrons/{patronId}](#) retrieve patron

post [api/patrons](#) add new patron

put [api/patrons/{patronId}](#) update patron

delete [api/patrons/{patronId}](#) delete patron



## CURD Book

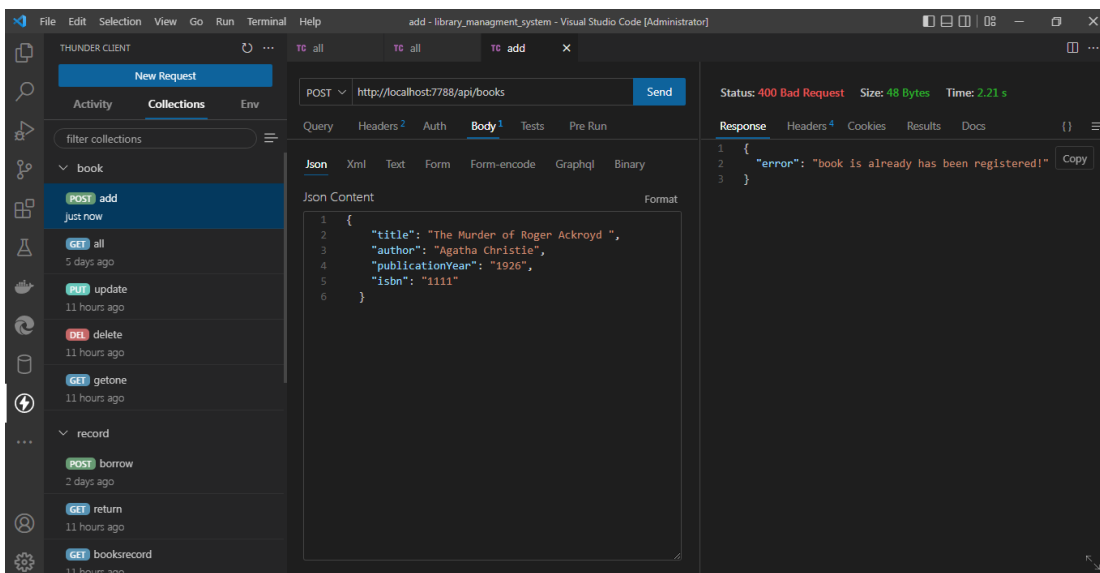
get `api/ books` all books

get `api/books /{bookId}` retrieve book

post `api/books` add new book

put `api/books /{bookId}` update book

delete `api/ books /{bookId}` delete book



## Library Managment

Post `/api/borrow/{bookId}/patron/{patronId}` patron borrow a book

Put `/api/return/{bookId}/patron/{patronId}` patron return a book

Get `/api/borrowing/records/patron/{patronId}` get patron's borrowing records

Get [/api/borrowing/records/book/{bookId}](#) get book's borrowing records