**Creating a Spring Boot Application**

**Goal:**

This exercise guides through creating a Spring Boot application for the library management system. Spring Boot simplifies configuration and deployment compared to traditional Spring applications.

**Steps:**

1. **Create a Spring Boot Project:**

Went to [Spring Initializr](https://start.spring.io/).

project details:

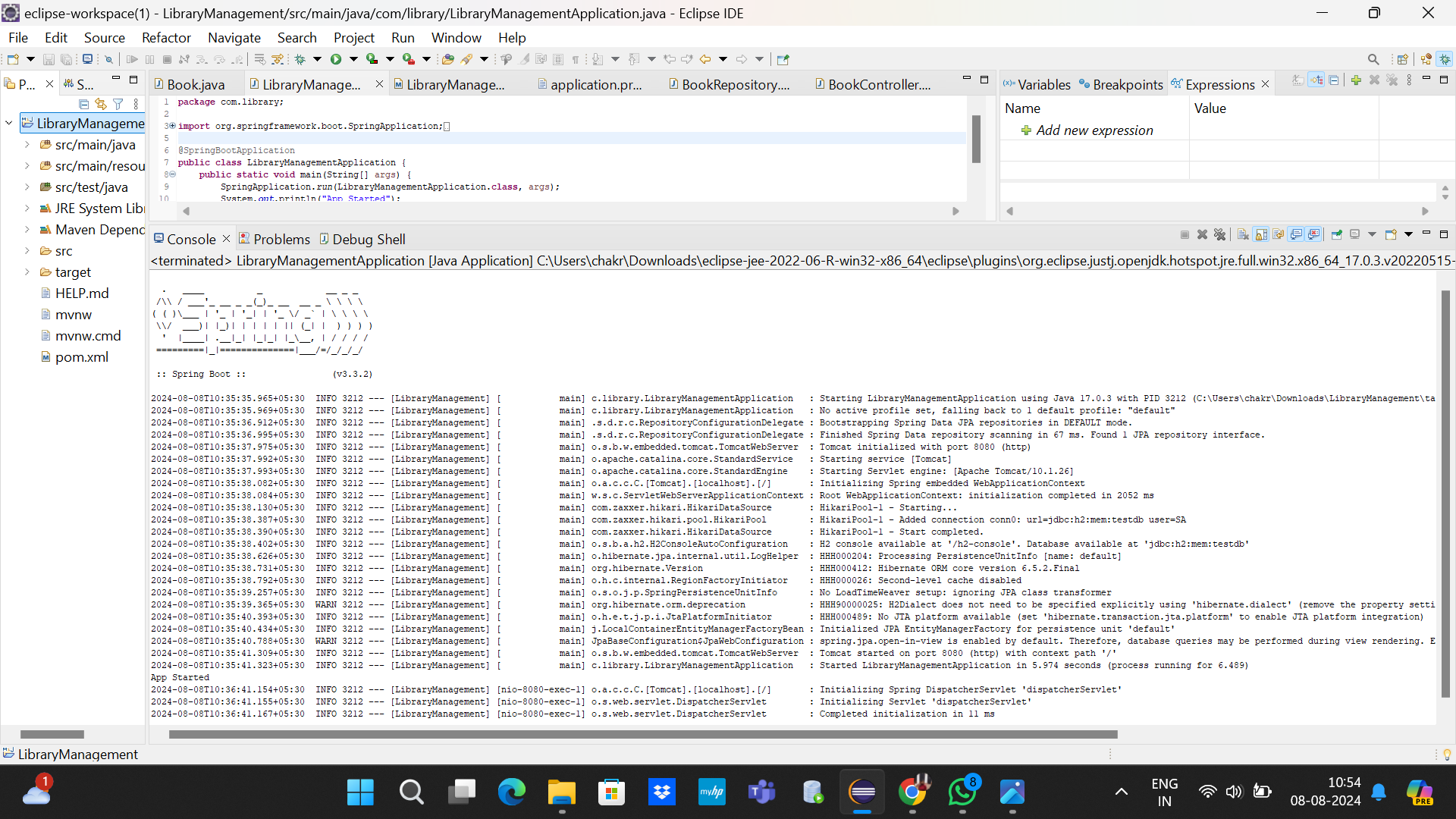
* + Project: Maven Project
  + Language: Java
  + Spring Boot: Latest version
  + Group: com.library
  + Artifact: LibraryManagement
  + Name: LibraryManagement
  + Package name: com.library
  + Packaging: Jar
  + Java: 22

Added dependencies:

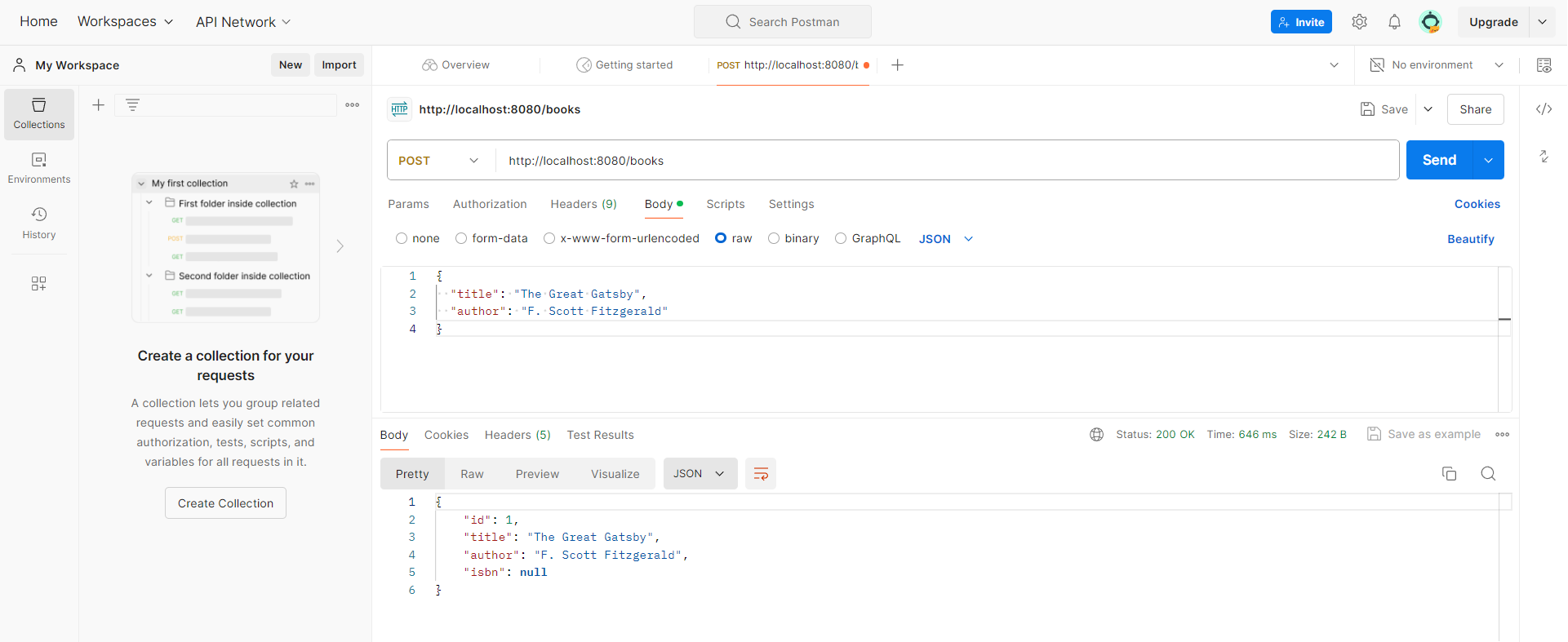
* + Spring Web (REST API development)
  + Spring Data JPA (database access)
  + H2 Database (in-memory database for development)

Clicked on Generate to download the project.

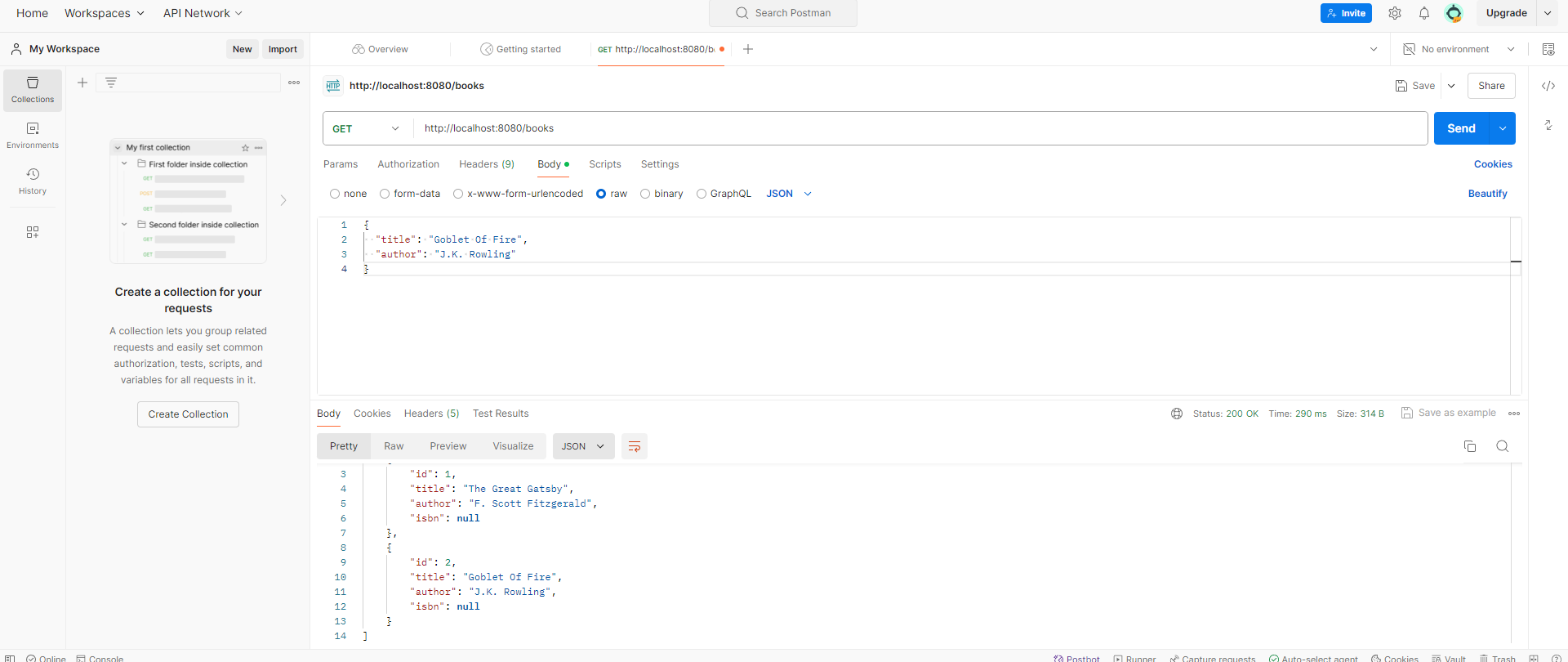
Unzipped the downloaded project and opened it in Eclipse.

1. **Add Application Properties:**
   * Configure the application.properties file with database connection details for H2.
2. **Define Entities and Repositories:**
   * Create a Book entity class representing book information.
   * Annotate it with @Entity for JPA persistence.
   * Define properties like title, author, ISBN, etc.
   * Create a BookRepository interface extending JpaRepository<Book, Long>. This provides CRUD operations for Book entities.
3. **Create a REST Controller:**
   * Develop a BookController class annotated with @RestController.
   * Define methods for CRUD operations on books:
     + @GetMapping: Get all books.
     + @PostMapping: Create a new book.
     + @GetMapping("/{id}): Get a book by ID.
     + @PutMapping("/{id}): Update a book by ID.
     + @DeleteMapping("/{id}): Delete a book by ID.
   * Utilize BookRepository methods to interact with the database.
   * Use @Autowired to inject the BookRepository bean.
   * Handle successful and unsuccessful operations with appropriate HTTP status codes.
4. **Run the Application:**
   * Run the LibraryManagementApplication class as a Java application.
   * This starts the embedded Tomcat server and deploys the Spring Boot application.
5. **Testing the Application:**
   * Use Postman tool to test the REST endpoints.

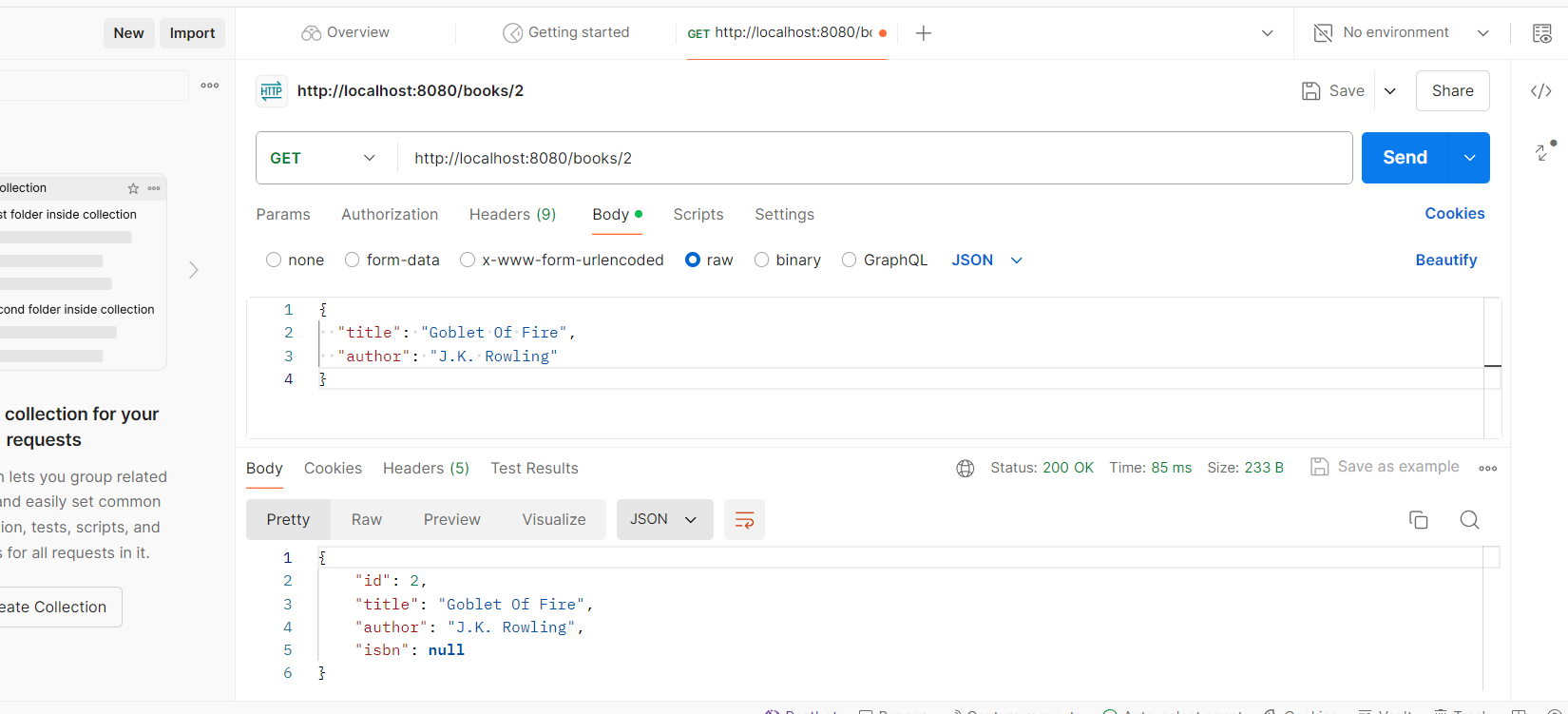
**POST BOOKS (Create Operation)-**



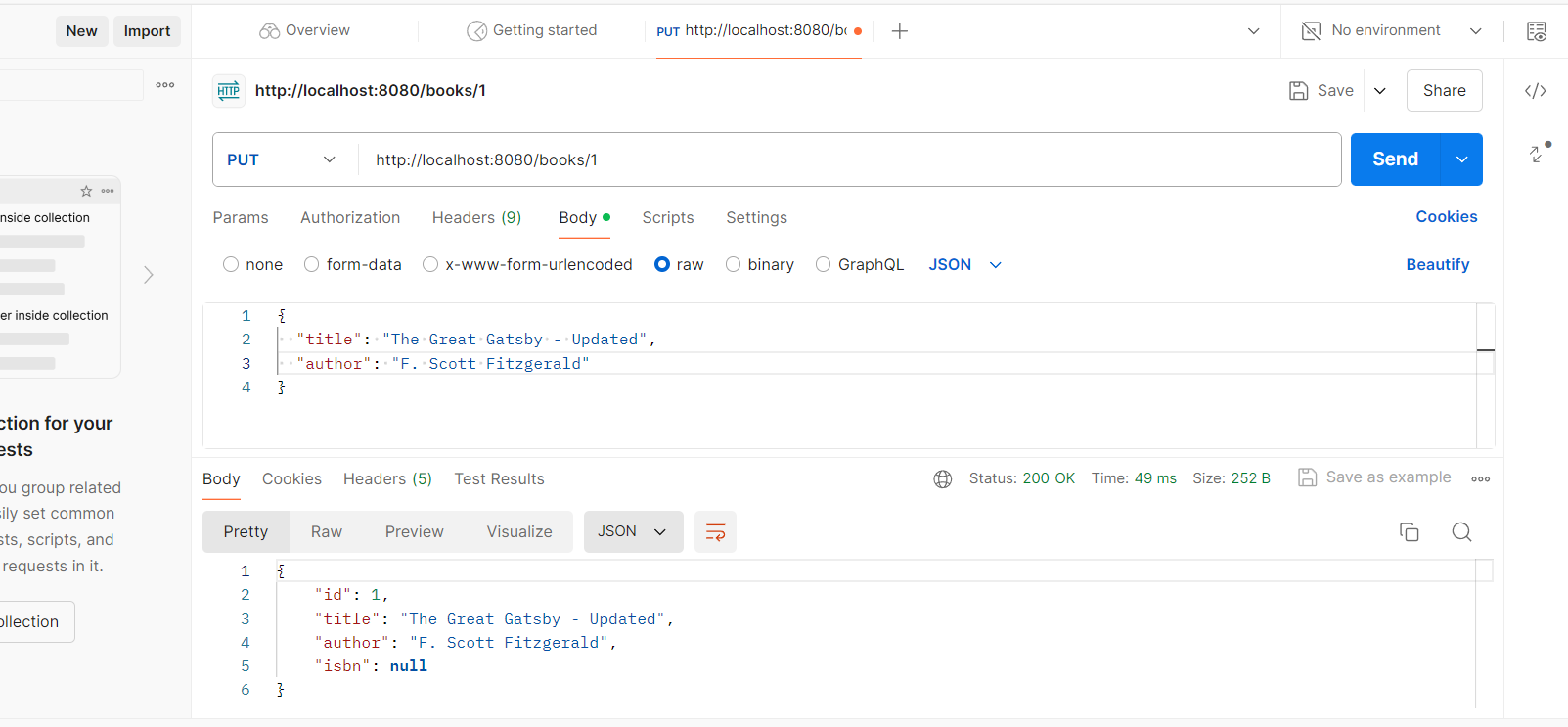
**GET BOOKS (Read Operation)-**



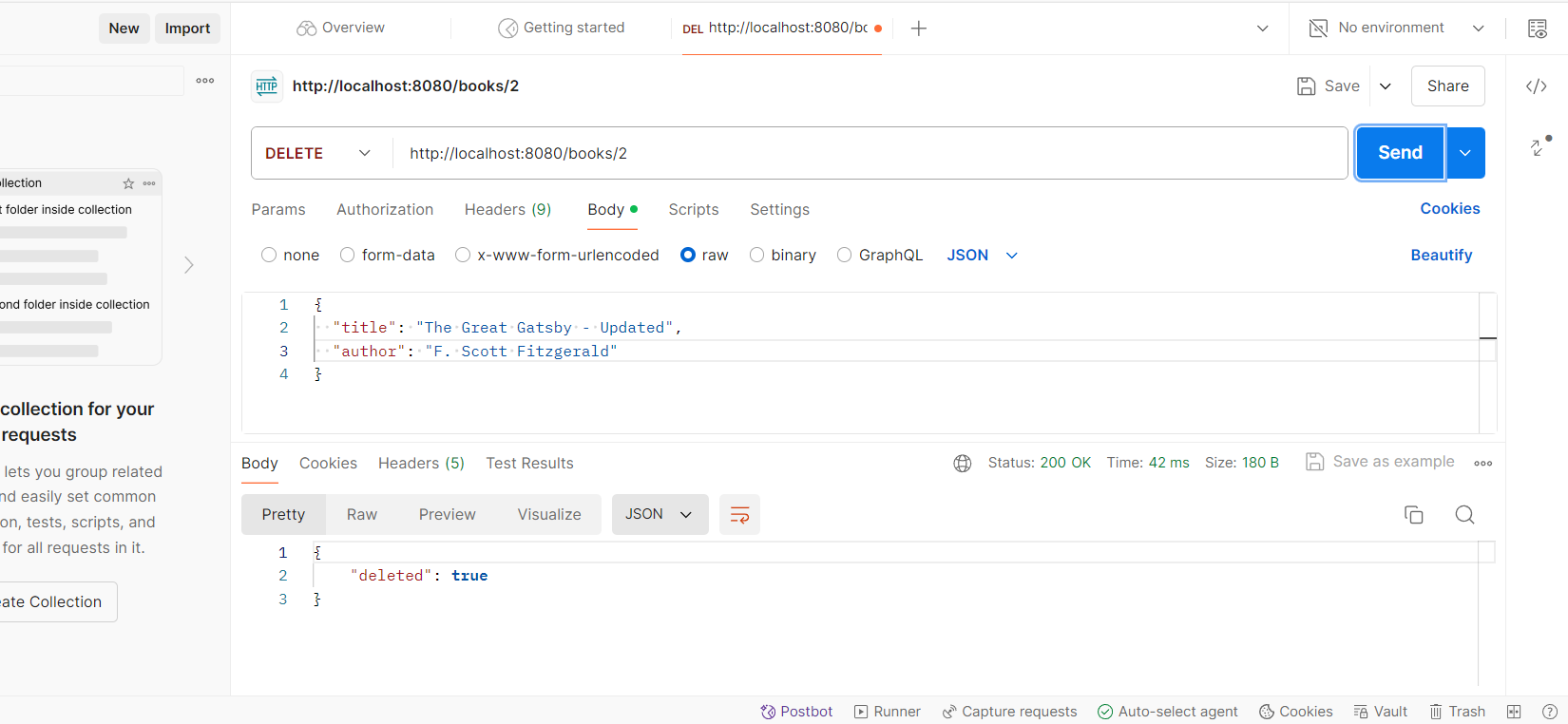
**GET BOOK BY ID-**



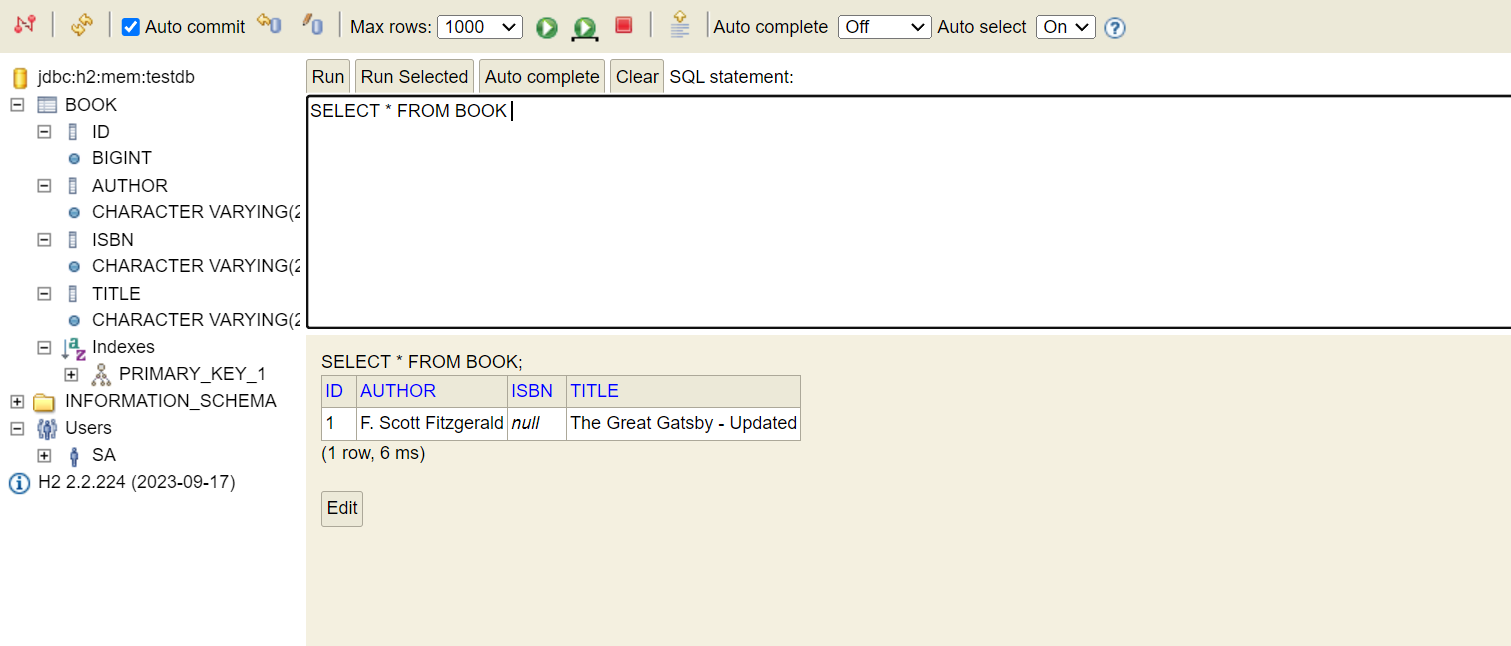
**PUT BOOK-(Update Operation)**



**DELETE BOOK-**



* + Access H2 database console (http://localhost:8080/h2-console) to verify data persistence.



**Benefits of Spring Boot:**

* Simplified configuration with minimal XML configuration.
* Faster development and deployment with auto-configuration.
* Embedded server (Tomcat) for easy local testing.
* Centralized management of dependencies.