**Exercise 1: Configuring a Basic Spring Application &**

**Exercise 2: Implementing Dependency Injection &**

**Exercise 5: Configuring the Spring IoC Container**

**LibraryManagement [Project]**

**Pom.xml**

*<dependency>*

*<groupId>org.springframework</groupId>*

*<artifactId>spring-context</artifactId>*

*<version>5.3.33</version>*

*</dependency>*

**BookRepository.java**

*package com.library.repository;*

*public class BookRepository {*

*public void saveBook() {*

*System.out.println("Book saved in the repo...");*

*}*

*}*

**BookService.java**

*package com.library.service;*

*import com.library.repository.BookRepository;*

*public class BookService extends BookRepository {*

*private BookRepository bookRepository;*

*public void setBookRepository(BookRepository bookRepository) {*

*this.bookRepository = bookRepository;*

*}*

*public void saveBookService() {*

*bookRepository.saveBook();*

*}*

*}*

**applicationContext.xml**

*<?****xml*** *version="1.0" encoding="UTF-8"?>*

*<****beans*** *xmlns="http://www.springframework.org/schema/beans"*

*xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"*

*xsi:schemaLocation="*

*http://www.springframework.org/schema/beans*

*https://www.springframework.org/schema/beans/spring-beans.xsd">*

*<****bean*** *id="bookRepository" class="com.library.repository.BookRepository" />*

*<****bean*** *id="bookService" class="com.library.service.BookService">*

*<****property*** *name="bookRepository" ref="bookRepository" />*

*</****bean****>*

*</****beans****>*

**App.java**

*package com.library;*

*import org.springframework.context.ApplicationContext;*

*import org.springframework.context.support.ClassPathXmlApplicationContext;*

*import com.library.service.BookService;*

*/\*\**

*\* Hello world!*

*\**

*\*/*

*public class App*

*{*

*public static void main( String[] args )*

*{*

*ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");*

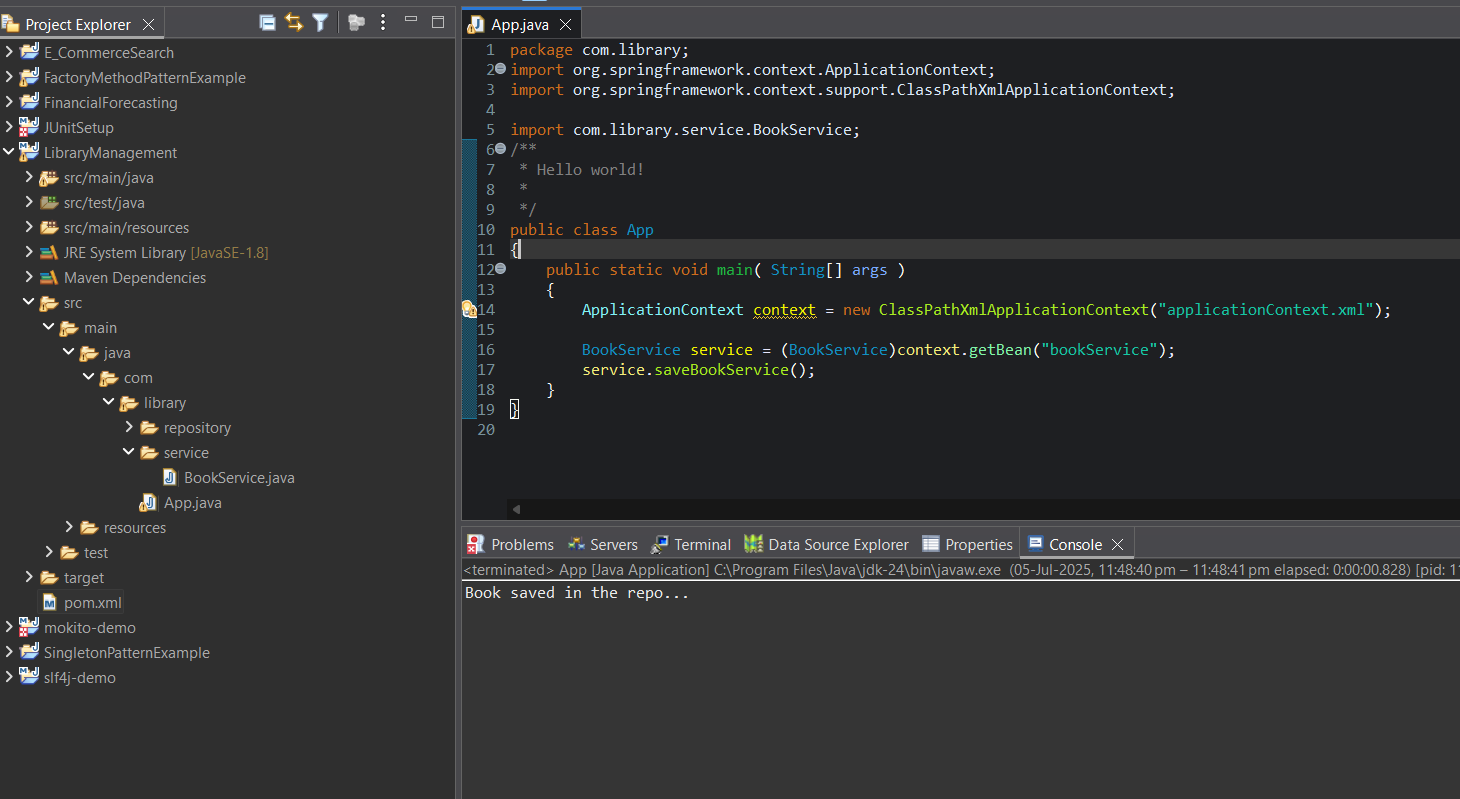
*BookService service = (BookService)context.getBean("bookService");*

*service.saveBookService();*

*}*

*}*

***Output:***

**

**Exercise 4: Creating and Configuring a Maven Project**

**Pom.xml**

*<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"*

*xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">*

*<modelVersion>4.0.0</modelVersion>*

*<groupId>com.example</groupId>*

*<artifactId>LibraryManagement</artifactId>*

*<version>0.0.1-SNAPSHOT</version>*

*<packaging>jar</packaging>*

*<name>LibraryManagement</name>*

*<url>http://maven.apache.org</url>*

*<properties>*

*<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>*

*</properties>*

*<dependencies>*

*<dependency>*

*<groupId>junit</groupId>*

*<artifactId>junit</artifactId>*

*<version>3.8.1</version>*

*<scope>test</scope>*

*</dependency>*

*<!-- Spring Context -->*

*<dependency>*

*<groupId>org.springframework</groupId>*

*<artifactId>spring-context</artifactId>*

*<version>5.3.33</version>*

*</dependency>*

*<!-- Spring AOP -->*

*<dependency>*

*<groupId>org.springframework</groupId>*

*<artifactId>spring-aop</artifactId>*

*<version>5.3.33</version>*

*</dependency>*

*<!-- Spring WebMVC -->*

*<dependency>*

*<groupId>org.springframework</groupId>*

*<artifactId>spring-webmvc</artifactId>*

*<version>5.3.33</version>*

*</dependency>*

*</dependencies>*

*<build>*

*<plugins>*

*<plugin>*

*<artifactId>maven-compiler-plugin</artifactId>*

*<version>3.8.1</version>*

*<configuration>*

*<source>1.8</source>*

*<target>1.8</target>*

*</configuration>*

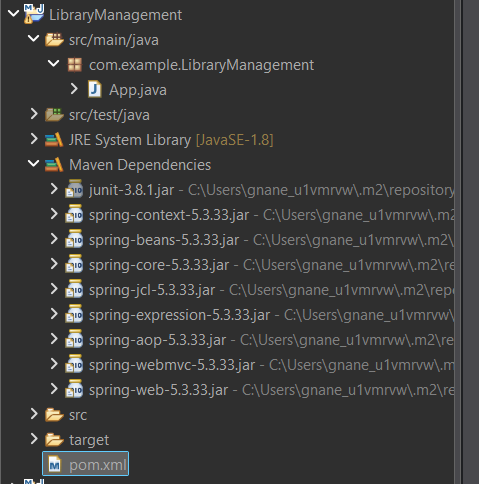
*</plugin>*

*</plugins>*

*</build>*

*</project>*

**Output:**

****

**Exercise 7: Implementing Constructor and Setter Injection**

**BookService.java**

*package com.library.service;*

*import com.library.repository.BookRepository;*

*public class BookService extends BookRepository {*

*private BookRepository bookRepository;*

*//added to the exisiting code*

*public BookService(BookRepository bookRepository) {*

*this.bookRepository = bookRepository;*

*System.****out****.println("Constructor Injection is being Used");*

*}*

*public void setBookRepository(BookRepository bookRepository) {*

*this.bookRepository = bookRepository;*

*}*

*public void saveBookService() {*

*bookRepository.saveBook();*

*}*

*}*

**applicationContext.xml**

*<?****xml*** *version="1.0" encoding="UTF-8"?>*

*<****beans*** *xmlns="http://www.springframework.org/schema/beans"*

*xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"*

*xsi:schemaLocation="*

*http://www.springframework.org/schema/beans*

*https://www.springframework.org/schema/beans/spring-beans.xsd">*

*<****bean*** *id="bookRepository" class="com.library.repository.BookRepository" />*

*<****bean*** *id="bookService" class="com.library.service.BookService">*

*<****property*** *name="bookRepository" ref="bookRepository" />*

*<****constructor-arg*** *ref="bookRepository"></****constructor-arg****>*

*</****bean****>*

*</****beans****>*

**App.java**

*package com.library;*

*import org.springframework.context.ApplicationContext;*

*import org.springframework.context.support.ClassPathXmlApplicationContext;*

*import com.library.service.BookService;*

*public class App*

*{*

*public static void main( String[] args )*

*{*

*ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");*

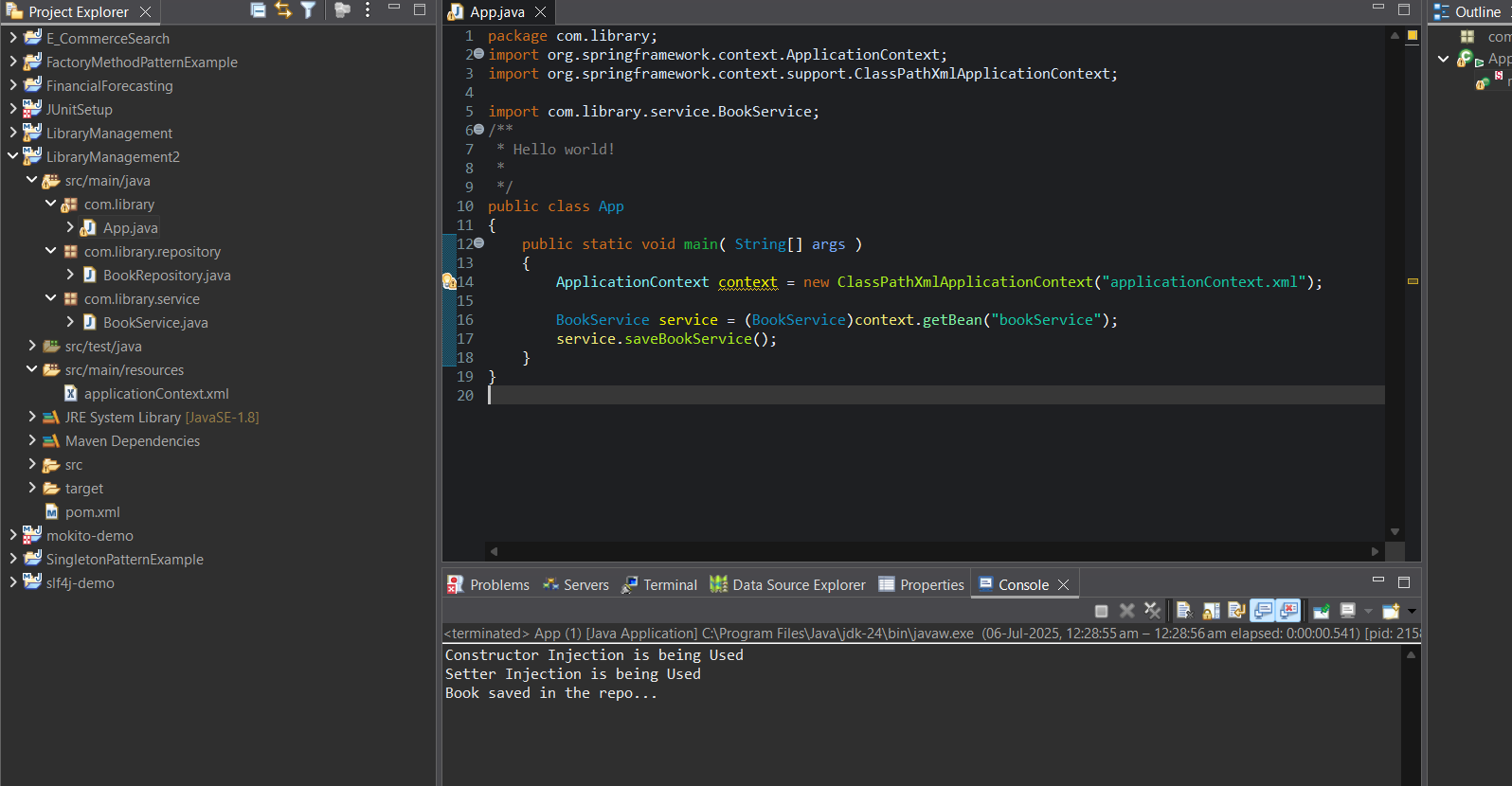
*BookService service = (BookService)context.getBean("bookService");*

*service.saveBookService();*

*}*

*}*

***Output:***



**Exercise 9: Creating a Spring Boot Application**

**Book.java**

*package com.library.LibraryManagement.Books;*

*import jakarta.persistence.Entity;*

*import jakarta.persistence.GeneratedValue;*

*import jakarta.persistence.GenerationType;*

*import jakarta.persistence.Id;*

*@Entity*

*public class Book {*

*@Id*

*@GeneratedValue(strategy = GenerationType.****IDENTITY****)*

*private Long id;*

*private String title;*

*private String author;*

*public Long getId() {*

*return id;*

*}*

*public void setId(Long id) {*

*this.id = id;*

*}*

*public String getTitle() {*

*return title;*

*}*

*public void setTitle(String title) {*

*this.title = title;*

*}*

*public String getAuthor() {*

*return author;*

*}*

*public void setAuthor(String author) {*

*this.author = author;*

*}*

*}*

**Repository.java**

*package com.library.LibraryManagement;*

*import org.springframework.data.jpa.repository.JpaRepository;*

*import com.library.LibraryManagement.Books.Book;*

*public interface Repository extends JpaRepository<Book, Long> {*

*}*

**BookController.java**

*package com.library.LibraryManagement.controller;*

*import java.util.List;*

*import org.springframework.beans.factory.annotation.Autowired;*

*import org.springframework.web.bind.annotation.DeleteMapping;*

*import org.springframework.web.bind.annotation.GetMapping;*

*import org.springframework.web.bind.annotation.PathVariable;*

*import org.springframework.web.bind.annotation.PostMapping;*

*import org.springframework.web.bind.annotation.RequestBody;*

*import org.springframework.web.bind.annotation.RequestMapping;*

*import org.springframework.web.bind.annotation.RestController;*

*import com.library.LibraryManagement.Repository;*

*import com.library.LibraryManagement.Books.Book;*

*@RestController*

*@RequestMapping("/books")*

*public class BookController {*

*@Autowired*

*private Repository bookRepository;*

*@GetMapping*

*public List<Book> getAllBooks() {*

*return bookRepository.findAll();*

*}*

*@PostMapping*

*public Book addBook(@RequestBody Book book) {*

*return bookRepository.save(book);*

*}*

*@GetMapping("/{id}")*

*public Book getBookById(@PathVariable Long id) {*

*return bookRepository.findById(id).orElse(null);*

*}*

*@DeleteMapping("/{id}")*

*public void deleteBook(@PathVariable Long id) {*

*bookRepository.deleteById(id);*

*}*

*}*

***App.java***

*package com.library.LibraryManagement;*

*import org.springframework.boot.SpringApplication;*

*import org.springframework.boot.autoconfigure.SpringBootApplication;*

*@SpringBootApplication*

*public class LibraryManagementApplication {*

*public static void main(String[] args) {*

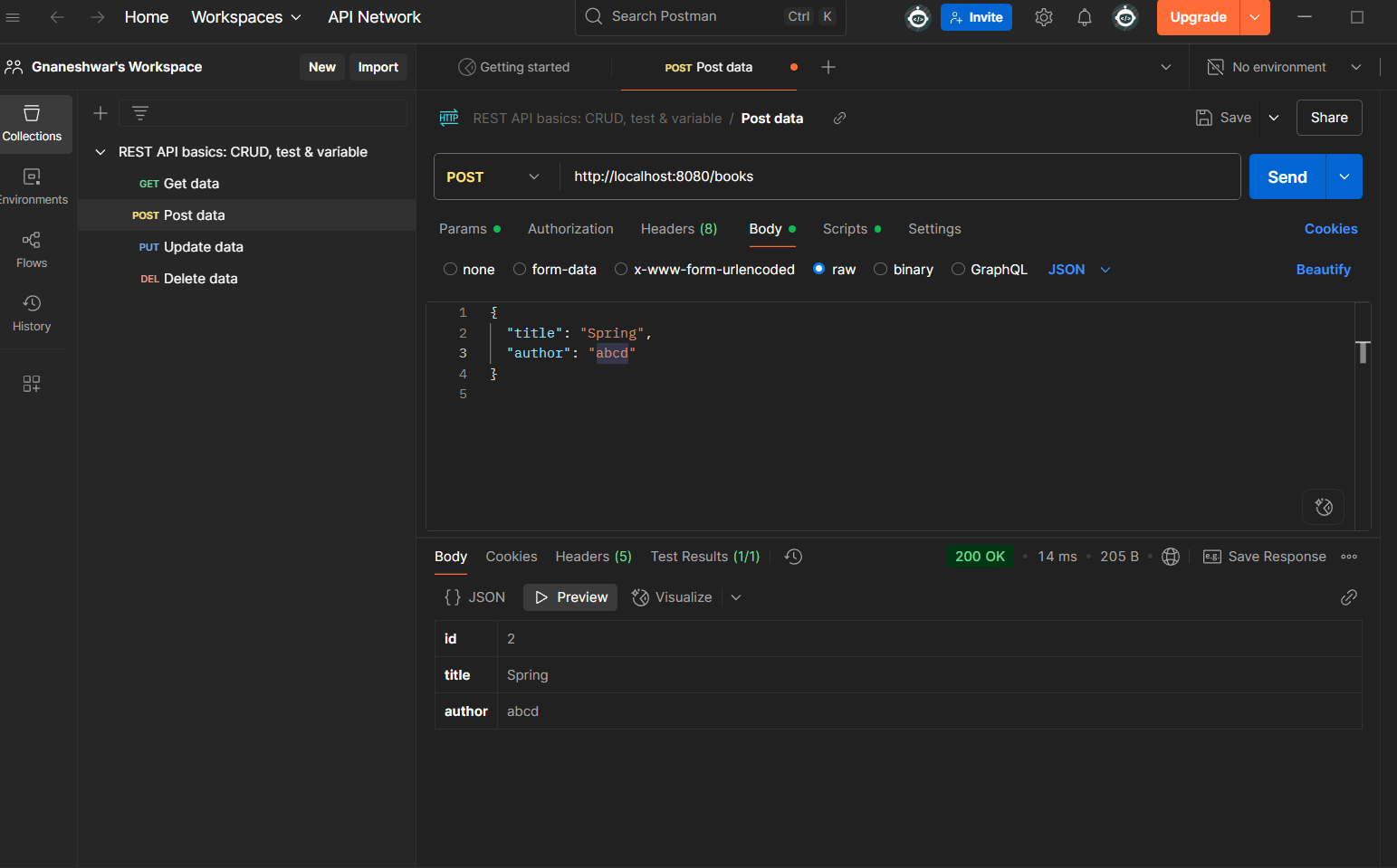
*SpringApplication.run(LibraryManagementApplication.class, args);*

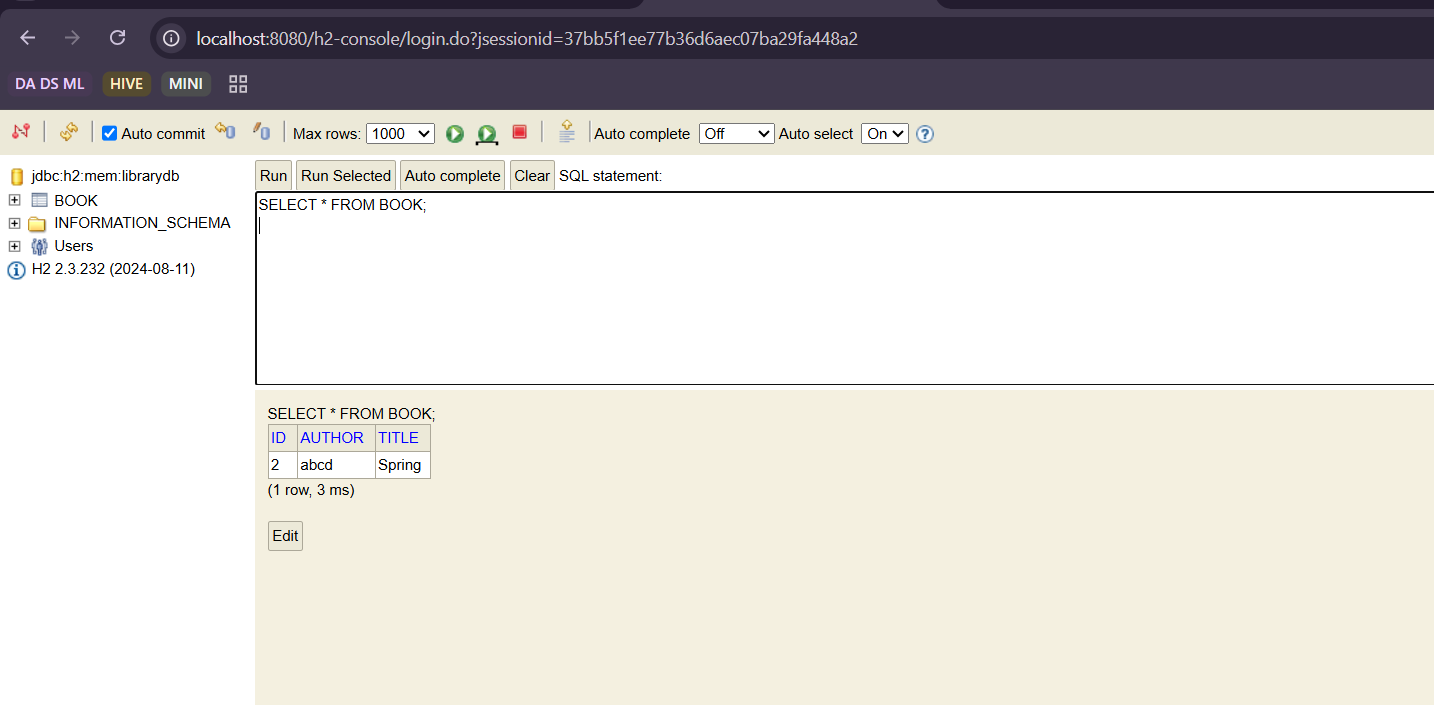
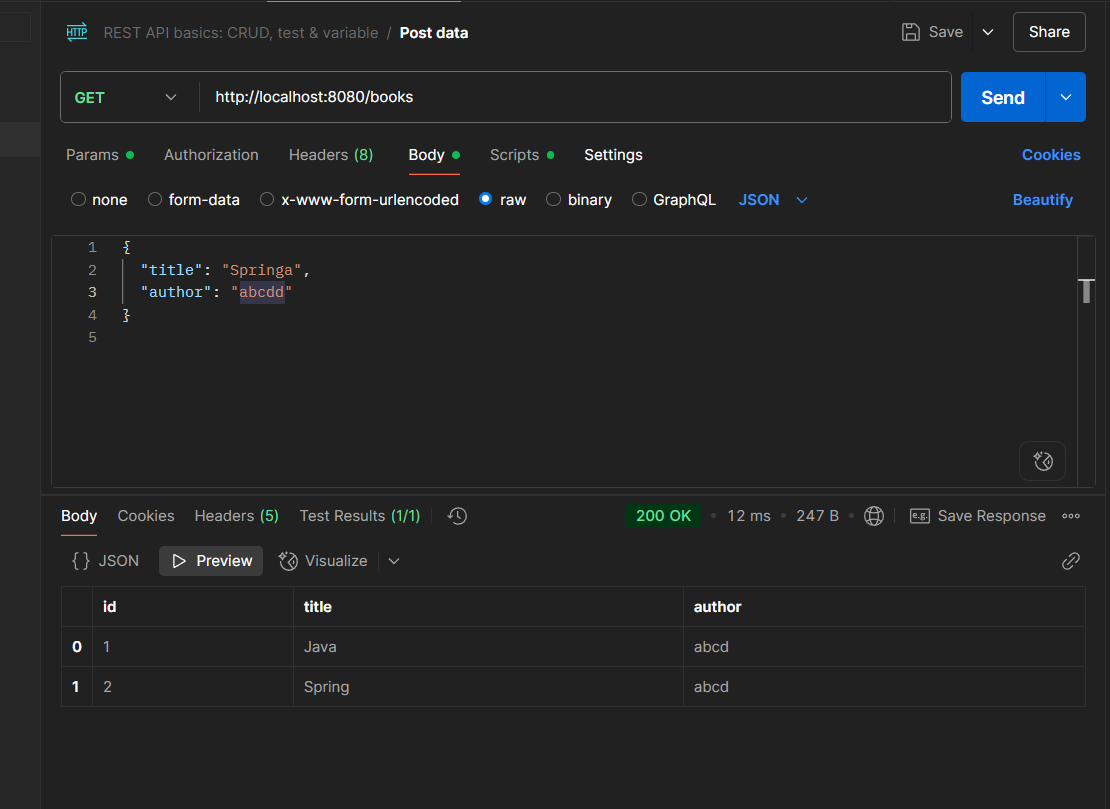
*}*

*}*

**Output:**

Posting using postman



Viewing the data in the databaseto get the data present 

deleting a record/book