**SPRING CORE AND MAVEN**

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

LibraryManagement\src\main\java\com\library\**MainApp.java**

package com.library;

import com.library.service.BookService;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class MainApp {

    public static void main(String[] args) {

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

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

        bookService.addBook("Spring in Action");

    }

}

LibraryManagement\src\main\java\com\library\repository\**BookRepository.java**

package com.library.repository;

public class BookRepository {

    public void saveBook(String bookName) {

        System.out.println("Saving book: " + bookName);

    } }

LibraryManagement\src\main\resources\**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

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

    <!-- Define Repository Bean -->

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

    <!-- Define Service Bean and inject Repository -->

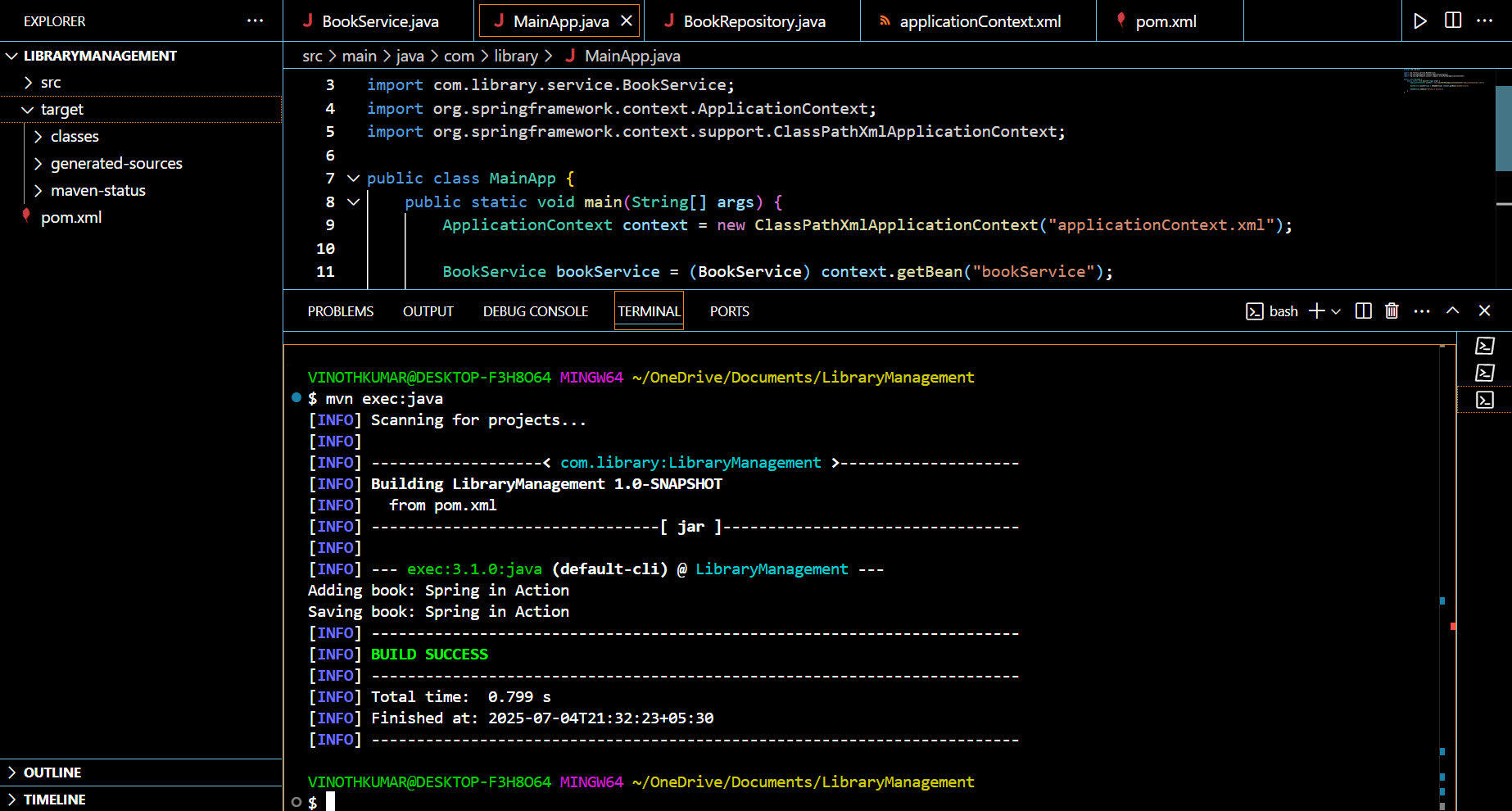
    <bean id="bookService" class="com.library.service.BookService">

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

    </bean>

</beans>

**OUTPUT :**

****

**Exercise 2: Implementing Dependency Injection**

LibraryManagement\src\main\java\com\library\repository\**BookRepository.java**

package com.library.repository;

public class BookRepository {

    public void save(String bookName) {

        System.out.println("Saving book: " + bookName);

    }

}

LibraryManagement\src\main\java\com\library\service\**BookService.java**

package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;

    // Setter for dependency injection

    public void setBookRepository(BookRepository bookRepository) {

        this.bookRepository = bookRepository;

    }

    public void addBook(String bookName) {

        System.out.println("Adding book: " + bookName);

        bookRepository.save(bookName);

    }

}

LibraryManagement\src\main\java\com\library\**MainApp.java**

package com.library;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.library.service.BookService;

public class MainApp {

    public static void main(String[] args) {

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

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

        bookService.addBook("Spring in Action");

    } }

LibraryManagement\src\main\resources\**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

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

    <!-- Define BookRepository bean -->

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

    <!-- Define BookService bean and inject BookRepository -->

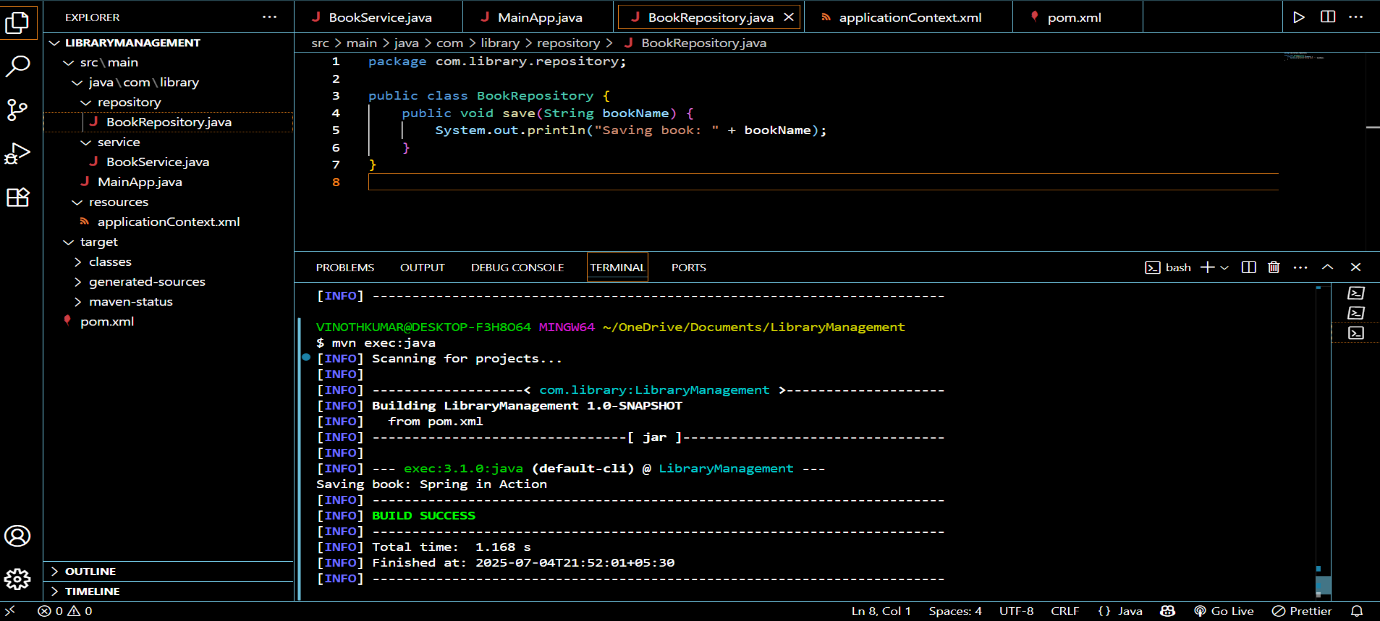
<bean id="bookService" class="com.library.service.BookService">

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

</bean>

</beans>

**OUTPUT :**

****

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

LibraryManagement\**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.library</groupId>

<artifactId>LibraryManagement</artifactId>

<version>1.0-SNAPSHOT</version>

<dependencies>

<!-- Spring Core and 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 Web MVC -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.3.33</version>

</dependency>

</dependencies>

<!-- Compiler Plugin Configuration -->

<build>

<plugins>

<!-- Compiler plugin -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

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

<version>3.11.0</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

<!-- Maven exec plugin to run main class -->

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>exec-maven-plugin</artifactId>

<version>3.1.0</version>

<configuration>

<mainClass>com.library.MainApp</mainClass>

</configuration>

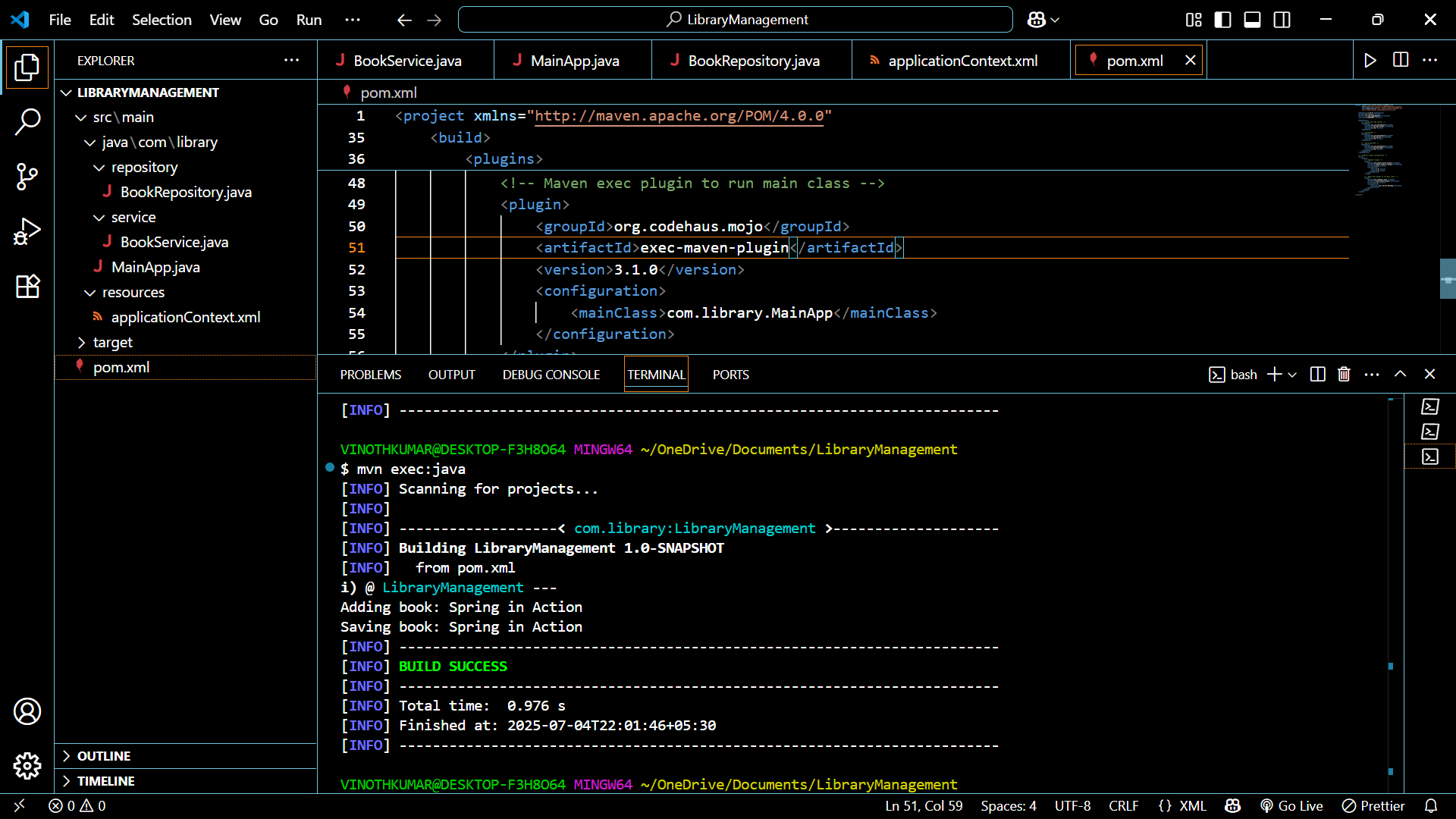
</plugin>

</plugins>

</build>

</project>

**OUTPUT :**

****