**Exercise 2: Implementing Dependency Injection**

**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>1.0.0</version>

<!-- compile with java 17 -->

<properties>

<maven.compiler.source>17</maven.compiler.source>

<maven.compiler.target>17</maven.compiler.target>

</properties>

<dependencies>

<!-- spring core / context -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.30</version>

</dependency>

</dependencies>

<build>

<plugins>

<!-- runs the main class without manual classpath -->

<plugin>

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

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

<version>3.1.0</version>

<configuration>

<mainClass>com.library.LibraryApp</mainClass>

</configuration>

</plugin>

</plugins>

</build>

</project>

**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

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

<!-- repository bean -->

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

<!-- service bean with setter‑based di -->

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

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

</bean>

</beans>

**src/main/java/com/library/model/Book.java**

java

CopyEdit

package com.library.model;

public class Book {

private String isbn;

private String title;

public Book(String isbn, String title) {

this.isbn = isbn;

this.title = title;

}

@Override

public String toString() {

return "[" + isbn + "] " + title;

}

}

**src/main/java/com/library/repository/BookRepository.java**

java

CopyEdit

package com.library.repository;

import com.library.model.Book;

public class BookRepository {

public void save(Book book) {

System.out.println("saved: " + book);

}

}

**src/main/java/com/library/service/BookService.java**

java

CopyEdit

package com.library.service;

import com.library.model.Book;

import com.library.repository.BookRepository;

public class BookService {

private BookRepository bookRepository; // injected by spring

// setter used for dependency injection

public void setBookRepository(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

public void addBook(Book book) {

bookRepository.save(book);

}

}

**src/main/java/com/library/LibraryApp.java**

java

CopyEdit

package com.library;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.library.model.Book;

import com.library.service.BookService;

public class LibraryApp {

public static void main(String[] args) {

// load spring context

ClassPathXmlApplicationContext ctx =

new ClassPathXmlApplicationContext("applicationContext.xml");

// get injected service bean

BookService service = ctx.getBean(BookService.class);

// use it

service.addBook(new Book("978-0132350884", "clean code"));

ctx.close();

}

}

**OUTPUT:**

