## Exercise 2: Implementing a Dependency Injection

## Adding dependencies in 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 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.library1</groupId>

<artifactId>LibraryManagement1</artifactId>

<version>1.0</version>

<dependencies>

<!-- Spring Core & Context -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.34</version>

</dependency>

</dependencies>

</project>

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 the BookRepository bean -->

<bean id="bookRepository" class="com.library1.repository.BookRepository" />

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

<bean id="bookService" class="com.library1.service.BookService">

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

</bean>

</beans>

**BookRepository.java:**

**package com.library1.repository;**

**public class BookRepository {**

**public void save(String title) {**

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

**}**

**}**

**BookService.java:**

package com.library1.service;

import com.library1.repository.BookRepository;

public class BookService {

private BookRepository bookRepository;

// Setter for Spring DI

public void setBookRepository(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

public void addBook(String title) {

System.out.println("BookService: Adding book - " + title);

bookRepository.save(title);

}

}

**MainApp.java:**

package com.library1;

import com.library1.service.BookService;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class MainApp {

public static void main(String[] args) {

// Load Spring context from XML

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

// Get BookService bean

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

// Call method to test injection

bookService.addBook("The Great Gatsby");

}

}

**OUTPUT:**

