**Exercise 2: Implementing Dependency Injection**

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.saveBook(bookName);

}

}

applicationContext.xml:

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

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

xmlns:aop="http://www.springframework.org/schema/aop"

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

http://www.springframework.org/schema/aop

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

<!-- Enable AspectJ auto-proxying -->

<aop:aspectj-autoproxy />

<!-- Repository bean -->

<bean id="bookRepository"

class="com.library.repository.BookRepository" />

<!-- Service bean with setter injection -->

<bean id="bookService"

class="com.library.service.BookService">

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

</bean>

<!-- Logging Aspect -->

<bean id="loggingAspect"

class="com.library.aspect.LoggingAspect" />

</beans>

App.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("1984");

}

}

Output:

Adding book: Suppandi

Book 'Suppandi' saved to the database.