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

**Scenario:**

The library management application requires both constructor and setter injection for better control over bean initialization.

**Steps:**

1. **Configure Constructor Injection:**
   * Update applicationContext.**xml** to configure constructor injection for **BookService**.
2. **Configure Setter Injection:**
   * Ensure that the **BookService** class has a setter method for **BookRepository** and configure it in **applicationContext.xml**.
3. **Test the Injection:**
   * Run the **LibraryManagementApplication** main class to verify both constructor and setter injection.

**Step 1 :**

**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"

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

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

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

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

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

<context:component-scan base-package="com.library" />

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

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

<constructor-arg ref="bookRepository" />

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

</bean>

</beans>

**Step 2 :**

**BookService.java**

package com.library.service;

public interface BookService {

void performService();

}

**BookServiceImpl.java**

package com.library.service;

import com.library.repository.BookRepository;

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

import org.springframework.stereotype.Service;

@Service

public class BookServiceImpl implements BookService {

private BookRepository bookRepository;

@Autowired

public BookServiceImpl(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

// Setter method for BookRepository

public void setBookRepository(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

@Override

public void performService() {

System.out.println("Performing service in BookService.");

bookRepository.performRepositoryTask();

}

}

**Step 3 :**

**Main.java**

package com.library.service;

import com.library.repository.BookRepository;

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

import org.springframework.stereotype.Service;

@Service

public class BookServiceImpl implements BookService {

private BookRepository bookRepository;

@Autowired

public BookServiceImpl(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

// Setter method for BookRepository

public void setBookRepository(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

@Override

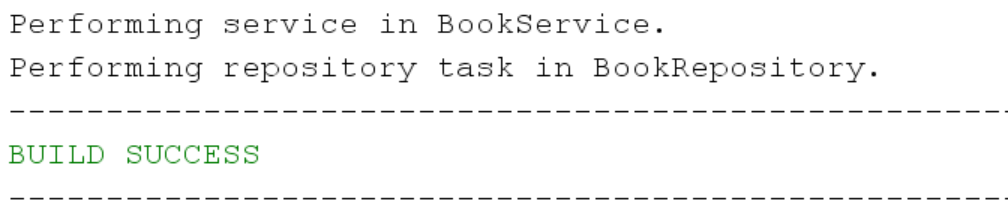
public void performService() {

System.out.println("Performing service in BookService.");

bookRepository.performRepositoryTask();

}

}

****