Exercise 2: Implementing Dependency Injection

Objective

Implement Dependency Injection (DI) using Spring IoC to wire BookService and BookRepository.

|  |
| --- |
| Technology Used |

|  |  |  |
| --- | --- | --- |
| Spring Core, Java, XML Configuration   |  | | --- | | Dependency Injection Type |  |  | | --- | | Setter Injection | |
| |  | | --- | | Steps Followed |  |  |  |  | | --- | --- | --- | | 1. Modified XML config to inject BookRepository into BookService  2. Verified BookService has proper setter  3. Ran main class to test DI   |  | | --- | | applicationContext.xml Changes |  |  | | --- | | Added <property name="bookRepository" ref="bookRepository" /> inside BookService bean | | |

Files Created

applicationContext.xml

com.library.repository.BookRepository

com.library.service.BookService

Main Class Code

package com.library.main;

import com.library.service.BookService;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class LibraryApp {

public static void main(String[] args) {

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

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

bookService.addBook("Clean Code");

}

}

Output

