**Exercise 5: Configuring the Spring IoC Container**

Scenario:

The library management application requires a central configuration for beans and dependencies.

**Introduction**

This document describes the steps taken to configure the Spring IoC (Inversion of Control) container for the Library Management Application. The configuration is centered around the creation of beans and their dependencies using the applicationContext.xml file.

**Steps and Updates**

**1. Creating the Spring Configuration File**

**File Created:**

* applicationContext.xml in src/main/resources

**Beans Defined:**

* BookService
* BookRepository

**2. Updating the BookService Class**

**Class Updated:**

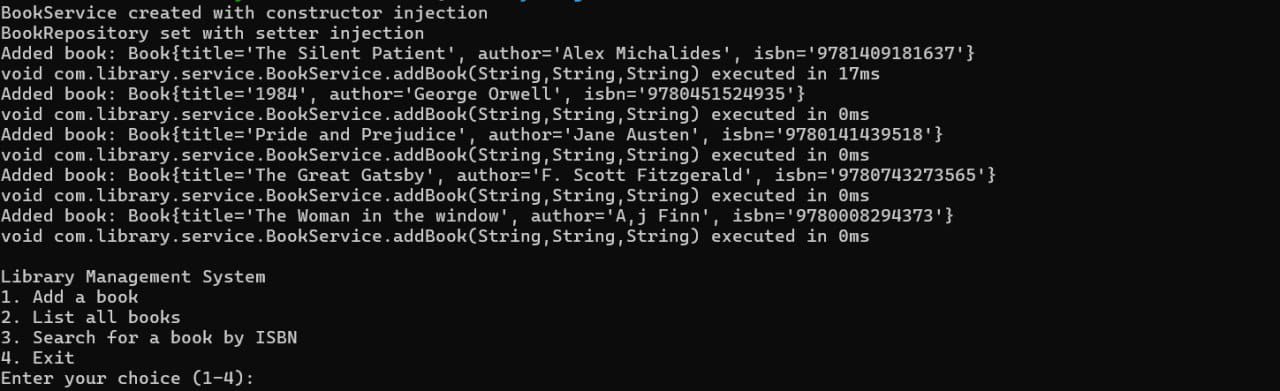
* BookService

**Change Description:**  
The BookService class was updated to include a setter method for injecting the BookRepository dependency.

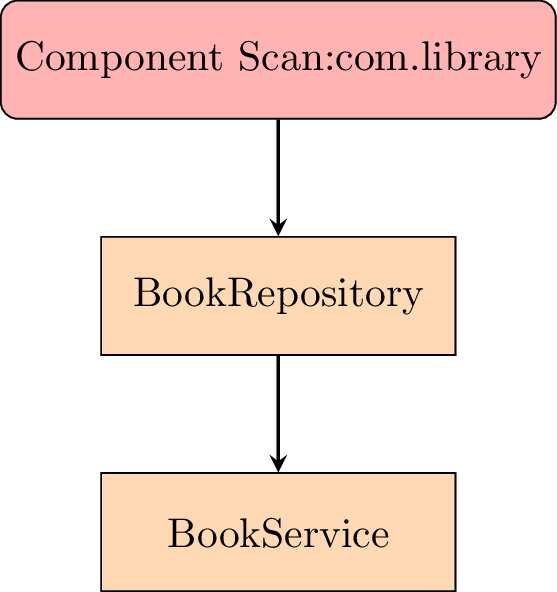
**3. Running the Application**

**Main Class Created:**  
A main class was created to load the Spring context and test the configuration.

**OUTPUT :**



**FLOWCHART of the program :**

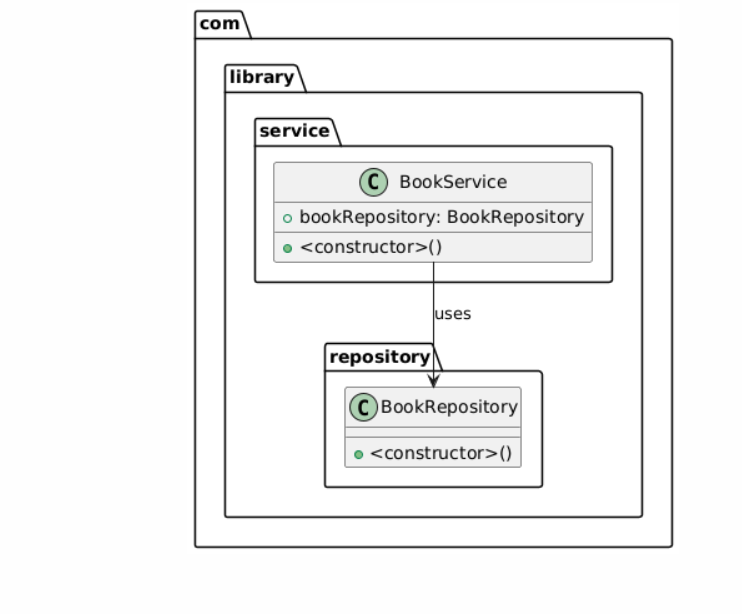


The flowchart visually represents the flow of the Spring IoC container as it initializes and manages the beans within the library management application.

* **Component Scan:** The Spring IoC container scans the package com.library to find and register beans.
* **BookRepository Bean:** The container instantiates the BookRepository bean.
* **BookService Bean:** The container then creates the BookService bean, injecting the BookRepository bean into it.

This flow ensures that the BookService bean has access to all required dependencies (BookRepository) as defined in the configuration.

**CLASS DIAGRAM :**



The class diagram depicts the relationships between the classes involved in the library management application.

* **BookRepository:** Represents the repository layer where book data is managed.
* **BookService:** Represents the service layer that uses BookRepository to perform business operations.
* **Dependency Injection:** The arrow from BookService to BookRepository indicates that BookService depends on BookRepository, which is injected via a setter method.

This diagram helps in understanding the structural relationship between the service and repository layers of the application.