**Exercise 2: Implementing Dependency Injection**

import org.springframework.context.support.GenericApplicationContext;

import org.springframework.beans.factory.xml.XmlBeanDefinitionReader;

import org.springframework.core.io.ByteArrayResource;

public class LibraryManagementApplication {

public static class BookRepository {

public void fetchBooks() {

System.out.println("Fetching books from the BookRepository...");

}

}

public static class BookService {

private BookRepository bookRepository;

public void setBookRepository(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

public void listAllBooks() {

System.out.println("BookService: Calling repository to fetch books...");

bookRepository.fetchBooks();

}

}

public static void main(String[] args) {

String xmlConfig = """

<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">

<!-- BookRepository Bean -->

<bean id="bookRepository" class="LibraryManagementApplication$BookRepository" />

<!-- BookService Bean with DI -->

<bean id="bookService" class="LibraryManagementApplication$BookService">

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

</bean>

</beans>

""";

GenericApplicationContext context = new GenericApplicationContext();

new XmlBeanDefinitionReader(context).loadBeanDefinitions(new ByteArrayResource(xmlConfig.getBytes()));

context.refresh();

BookService service = context.getBean("bookService", BookService.class);

service.listAllBooks();

context.close();

}

}

**Output:**

BookService: Calling repository to fetch books...

Fetching books from the BookRepository...