Exercise 1: Configuring a Basic Spring Application

Scenario:

Your company is developing a web application for managing a library. You need to use the Spring Framework to handle the backend operations.

Steps:

1. **Set Up a Spring Project:**

   o Create a Maven project named **LibraryManagement**.
   o Add Spring Core dependencies in the **pom.xml** file.

2. **Configure the Application Context:**

   o Create an XML configuration file named **applicationContext.xml** in the **src/main/resources** directory.
   o Define beans for **BookService** and **BookRepository** in the XML file.

3. **Define Service and Repository Classes:**

   o Create a package **com.library.service** and add a class **BookService**.
   o Create a package **com.library.repository** and add a class **BookRepository**.

4. **Run the Application:**

   o Create a main class to load the Spring context and test the configuration.

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"

    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

    http://maven.apache.org/xsd/maven-4.0.0.xsd">

  <modelVersion>4.0.0</modelVersion>


  <groupId>com.library</groupId>

  <artifactId>LibraryManagement</artifactId>

  <version>0.0.1-SNAPSHOT</version>


  <dependencies>

    <!-- Spring Core -->

    <dependency>

      <groupId>org.springframework</groupId>

      <artifactId>spring-context</artifactId>
```

```xml
        <version>5.3.34</version>
      </dependency>
   </dependencies>
</project>
```

applicationContext.xml:

```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"
    xsi:schemaLocation="
    http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans.xsd">

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


   <bean id="bookService" class="com.library.service.BookService">
     <property name="bookRepository" ref="bookRepository" />
   </bean>


</beans>
```

com.library.repository

- ♦ **BookRepository.java**

```java
package com.library.repository;


public class BookRepository {
   public void saveBook(String bookName) {
      System.out.println("Book saved: " + bookName);
   }
}
```

com.library.service

- ♦ **BookService.java**

```java
package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;
    public void setBookRepository(BookRepository bookRepository) {
        this.bookRepository = bookRepository;
    }

    public void addBook(String bookName) {
        bookRepository.saveBook(bookName);
    }
}
```

MainApp.java:

```java
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import com.library.service.BookService;

public class MainApp {
    public static void main(String[] args) {
        ApplicationContext context = new
ClassPathXmlApplicationContext("applicationContext.xml");

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

        bookService.addBook("Spring Framework Mastery");
    }
}
```

OUTPUT:

```
Book saved: Spring Framework Mastery
```