WEEK-3

**SPRING CORE\_MAVEN**

**Exercise 1: Configuring a Basic Spring Application**

LibraryManagement

├── pom.xml

└── src

└── main

├── java

│ └── com

│ └── library

│ ├── MainApp.java

│ ├── repository

│ │ └── BookRepository.java

│ └── service

│ └── BookService.java

└── resources

└── applicationContext.xml

**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>1.0-SNAPSHOT</version>

    <dependencies>

        <!-- Spring Context -->

        <dependency>

            <groupId>org.springframework</groupId>

            <artifactId>spring-context</artifactId>

            <version>5.3.34</version>

        </dependency>

    </dependencies>

    <build>

        <plugins>

            <plugin>

                <groupId>org.codehaus.mojo</groupId>

                <artifactId>exec-maven-plugin</artifactId>

                <version>3.1.0</version>

                <configuration>

                    <mainClass>com.library.MainApp</mainClass>

                </configuration>

            </plugin>

        </plugins>

    </build>

</project>

**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"

       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>

**MainApp.java:**

package com.library;

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

        bookService.addBook("Spring in Action");

    }

}

**BookRepository.java:**

package com.library.repository;

public class BookRepository {

    public void saveBook(String bookName) {

        System.out.println("Book '" + bookName + "' saved to the database.");

    }

}

**BookService.java:**

package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;

    // Setter for Dependency Injection

    public void setBookRepository(BookRepository bookRepository) {

        this.bookRepository = bookRepository;

    }

    public void addBook(String bookName) {

        System.out.println("BookService: Adding book...");

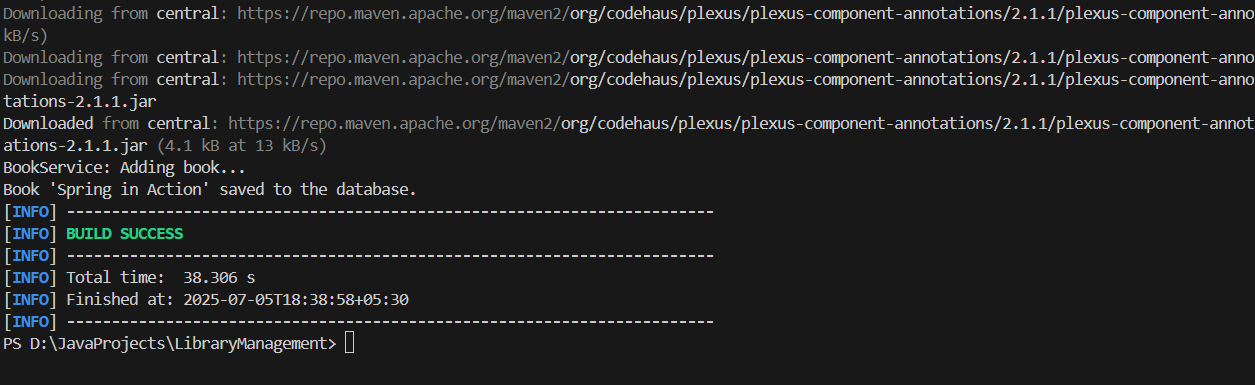
        bookRepository.saveBook(bookName);

    }

}

**Output:**





**Exercise 2: Implementing Dependency Injection**

**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"

       xsi:schemaLocation="http://www.springframework.org/schema/beans

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

    <!-- Bean for Repository -->

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

    <!-- Bean for Service with DI -->

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

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

    </bean>

</beans>

**BookService.java:**

package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;

    // Setter for Dependency Injection

    public void **setBookRepository**(BookRepository bookRepository) {

        this.bookRepository = bookRepository;

    }

    public void addBook(String bookName) {

        System.out.println("BookService: Adding book...");

        bookRepository.saveBook(bookName);

    }

}

**MainApp.java:**

package com.library;

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");

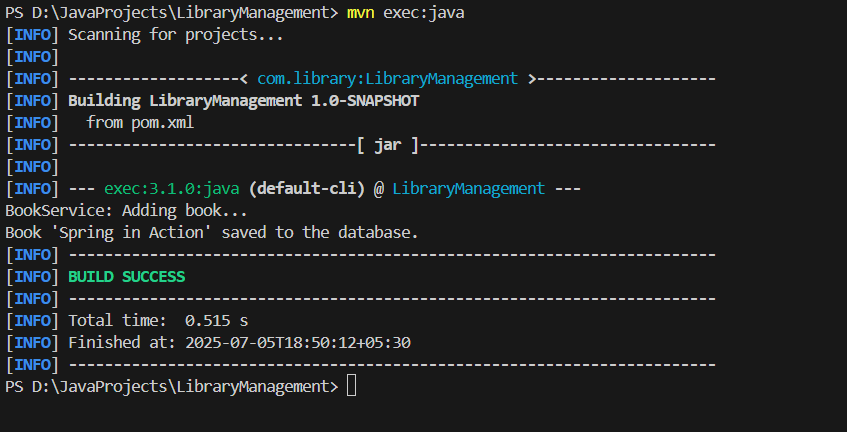
        BookService bookService = context.getBean("bookService", BookService.class);

        bookService.addBook("Spring in Action");

    }

}

**Output:**



**Exercise 4: Creating and Configuring a Maven Project**

Library\_Management/

├── pom.xml

└── src/

├── main/java/com/library/App.java

└── test/java/com/library/AppTest.java

**pom.xml:**

<?xml version="1.0" encoding="UTF-8"?>

<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>Library\_Management</artifactId>

<version>1.0-SNAPSHOT</version>

<name>Library\_Management</name>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.compiler.source>1.8</maven.compiler.source>

<maven.compiler.target>1.8</maven.compiler.target>

</properties>

<dependencies>

<!-- Spring Core/Context -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.34</version>

</dependency>

<!-- Spring AOP -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

<version>5.3.34</version>

</dependency>

<!-- Spring Web MVC -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.3.34</version>

</dependency>

<!-- JUnit (for testing) -->

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<!-- Compiler Plugin -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.1</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

</plugins>

</build>

</project>

**App.java:**

package com.library;

/\*\*

 \* Hello world!

\*\*/

public class App

{

    public static void main( String[] args )

    {

        System.out.println( "Hello World!" );

    }

}

**AppTest.java:**  
package com.library;

import static org.junit.Assert.assertTrue;

import org.junit.Test;

/\*\*

 \* Unit test for simple App.

 \*/

public class AppTest

{

    /\*\*

     \* Rigorous Test :-)

     \*/

    @Test

    public void shouldAnswerWithTrue()

    {

        assertTrue( true );

    }

}

**Output:**

