**WEEK – 3**

**Spring Core Maven - MANDATORY HANDS-ON EXERCISES**

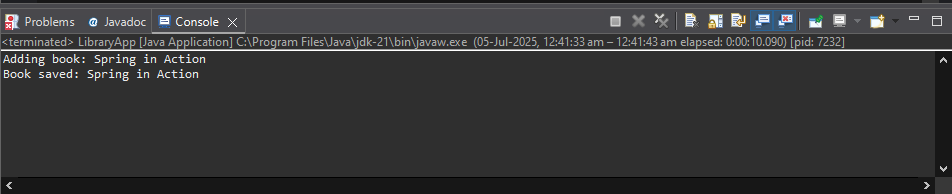
**Exercise 1: Configuring a Basic Spring Application:**

* **/LibraryManagement** –
  + /LibraryManagement/**src/main/java**

Create three packages

* + - com.library.main
    - com.library.repository
    - com.library.service
* In -> com.library.main, Create **/LibraryApp.java**
* package com.library.main;
* import org.springframework.context.ApplicationContext;
* import org.springframework.context.support.ClassPathXmlApplicationContext;
* import com.library.service.BookService;
* public class LibraryApp {
* public static void main(String[] args) {
* ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
* BookService bookService = (BookService) context.getBean("bookService");
* bookService.addBook("Spring in Action");
* }
* }
* In com.library.repository, Create **/BookRepository.java**
* package com.library.repository;
* public class BookRepository {
* public void saveBook(String title) {
* System.***out***.println("Book saved: " + title);
* }
* }
* In com.library.service, Create **/BookService.java**
* package com.library.service;
* import com.library.repository.BookRepository;
* public class BookService {
* private BookRepository bookRepository;
* // Setter Injection
* public void setBookRepository(BookRepository bookRepository) {
* this.bookRepository = bookRepository;
* }
* public void addBook(String title) {
* System.***out***.println("Adding book: " + title);
* bookRepository.saveBook(title);
* }
* }
* In **/LibraryManagement/src/main/resources -> /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">
* <!-- Repository Bean -->
* <bean id="bookRepository" class="com.library.repository.BookRepository"/>
* <!-- Service Bean -->
* <bean id="bookService" class="com.library.service.BookService">
* <property name="bookRepository" ref="bookRepository"/>
* </bean>
* </beans>
* In **/LibraryManagement/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
* https://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>
* <properties>
* <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>5.3.32</version>
* </dependency>
* <dependency>
* <groupId>commons-logging</groupId>
* <artifactId>commons-logging</artifactId>
* <version>1.2</version>
* </dependency>
* </dependencies>
* </project>
* **Right click on LibraryApp.java -> Run As -> Java Application**

**OUTPUT:**

****

**Exercise 2: Implementing Dependency Injection**

**Steps:**

Modify the XML Configuration:

1. <beans xmlns="http://www.springframework.org/schema/beans"
2. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3. xsi:schemaLocation="http://www.springframework.org/schema/beans
4. http://www.springframework.org/schema/beans/spring-beans.xsd">
5. <bean id="bookRepository" class="com.library.repository.BookRepository"/>
6. <bean id="bookService" class="com.library.service.BookService">
7. <property name="bookRepository" ref="bookRepository"/>
8. </bean>
9. </beans>

Update the BookService Class:}

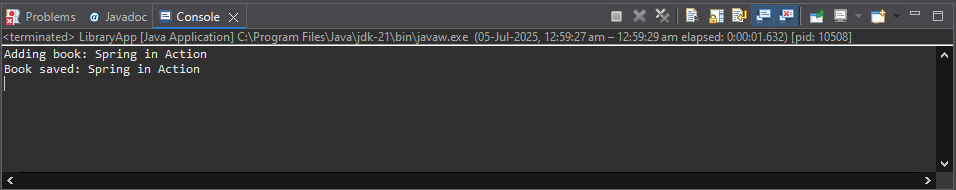
1. package com.library.service;
2. import com.library.repository.BookRepository;
3. public class BookService {
4. private BookRepository bookRepository;
5. // Setter for Dependency Injection
6. public void setBookRepository(BookRepository bookRepository) {
7. this.bookRepository = bookRepository;
8. }
9. public void addBook(String bookName) {
10. System.***out***.println("Adding book: " + bookName);
11. bookRepository.saveBook(bookName);
12. }
13. }

Test the Configuration:

1. package com.library.main;
2. import org.springframework.context.ApplicationContext;
3. import org.springframework.context.support.ClassPathXmlApplicationContext;
4. import com.library.service.BookService;
5. public class LibraryApp {
6. public static void main(String[] args) {
7. ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
8. BookService bookService = (BookService) context.getBean("bookService");
9. bookService.addBook("Spring in Action");
10. }
11. }

* **Right click on LibraryApp.java -> Run As -> Java Application**

**OUTPUT:**

****

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

* In Eclipse:
* Go to File → New → Maven Project -> **/LibraryAppSetup**
* Add Spring Dependencies in **pom.xml.**

1. <project xmlns="http://maven.apache.org/POM/4.0.0"
2. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3. xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
4. https://maven.apache.org/xsd/maven-4.0.0.xsd">
5. <modelVersion>4.0.0</modelVersion>
6. <groupId>com.library</groupId>
7. <artifactId>LibraryAppSetup</artifactId>
8. <version>0.0.1-SNAPSHOT</version>
9. <properties>
10. <maven.compiler.source>1.8</maven.compiler.source>
11. <maven.compiler.target>1.8</maven.compiler.target>
12. </properties>
13. <dependencies>
14. <!-- Spring Core / Context -->
15. <dependency>
16. <groupId>org.springframework</groupId>
17. <artifactId>spring-context</artifactId>
18. <version>5.3.32</version>
19. </dependency>
20. <!-- Spring AOP -->
21. <dependency>
22. <groupId>org.springframework</groupId>
23. <artifactId>spring-aop</artifactId>
24. <version>5.3.32</version>
25. </dependency>
26. <!-- Spring Web MVC -->
27. <dependency>
28. <groupId>org.springframework</groupId>
29. <artifactId>spring-webmvc</artifactId>
30. <version>5.3.32</version>
31. </dependency>
32. <!-- Logging (optional) -->
33. <dependency>
34. <groupId>commons-logging</groupId>
35. <artifactId>commons-logging</artifactId>
36. <version>1.2</version>
37. </dependency>
38. </dependencies>
39. <!-- Step 3: Maven Compiler Plugin -->
40. <build>
41. <plugins>
42. <plugin>
43. <groupId>org.apache.maven.plugins</groupId>
44. <artifactId>maven-compiler-plugin</artifactId>
45. <version>3.8.1</version>
46. <configuration>
47. <source>1.8</source>
48. <target>1.8</target>
49. </configuration>
50. </plugin>
51. </plugins>
52. </build>
53. </project>

* **Right-click project → Maven → Update Project**
* **Maven Dependencies:**
* commons-logging-1.2.jar
* spring-aop-5.3.32.jar
* spring-beans-5.3.32.jar
* spring-context-5.3.32.jar
* spring-core-5.3.32.jar
* spring-expression-5.3.32.jar
* spring-jcl-5.3.32.jar
* spring-web-5.3.32.jar
* spring-webmvc-5.3.32.jar

***(Checked Successfully!)***

* Confirm Compiler Plugin
* In pom.xml, confirm this section exists:

1. <properties>
2. <maven.compiler.source>1.8</maven.compiler.source>
3. <maven.compiler.target>1.8</maven.compiler.target>
4. </properties

* Also, confirm:

1. <plugin>
2. <groupId>org.apache.maven.plugins</groupId>
3. <artifactId>maven-compiler-plugin</artifactId>
4. <version>3.8.1</version>

* Test with a Small Java Class: (Optional)
  + create a test class:
  + **📁 com.library.test.HelloTest**
* package com.library.test;

public class HelloTest {

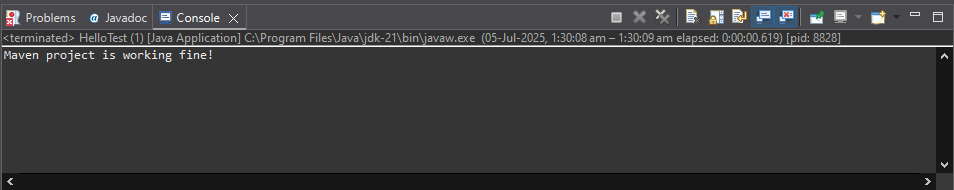
public static void main(String[] args) {

System.out.println("Maven project is working fine!");

}

}

**OUTPUT:**



**ADDITIONAL HANDS-ON EXERCISES**

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

* + **Create a New Maven Project:**

File → New → Maven Project -> **/LibraryIoCConfig**

* + Add Spring Dependencies in **pom.xml**

1. <project xmlns="http://maven.apache.org/POM/4.0.0"
2. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3. xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
4. https://maven.apache.org/xsd/maven-4.0.0.xsd">
5. <modelVersion>4.0.0</modelVersion>
6. <groupId>com.library</groupId>
7. <artifactId>LibraryIoCConfig</artifactId>
8. <version>0.0.1-SNAPSHOT</version>
9. <properties>
10. <maven.compiler.source>1.8</maven.compiler.source>
11. <maven.compiler.target>1.8</maven.compiler.target>
12. </properties>
13. <dependencies>
14. <!-- Spring Core Context -->
15. <dependency>
16. <groupId>org.springframework</groupId>
17. <artifactId>spring-context</artifactId>
18. <version>5.3.32</version>
19. </dependency>
20. <!-- Logging -->
21. <dependency>
22. <groupId>commons-logging</groupId>
23. <artifactId>commons-logging</artifactId>
24. <version>1.2</version>
25. </dependency>
26. </dependencies>
27. <!-- Java compiler plugin -->
28. <build>
29. <plugins>
30. <plugin>
31. <groupId>org.apache.maven.plugins</groupId>
32. <artifactId>maven-compiler-plugin</artifactId>
33. <version>3.8.1</version>
34. <configuration>
35. <source>1.8</source>
36. <target>1.8</target>
37. </configuration>
38. </plugin>
39. </plugins>
40. </build>
41. </project>

✔️ **Right-click project → Maven → Update Project**  
 Make sure dependencies are downloaded (spring-context, commons-logging)

* + Create Java Packages and Classes

Create the following inside **src/main/java**:

* + com.library.repository.BookRepository
  + com.library.service.BookService
  + com.library.repository.BookRepository

1. package com.library.repository;
2. public class BookRepository {
3. public void saveBook(String title) {
4. System.***out***.println("Book saved: " + title);
5. }
6. }
   * com.library.service.BookService
7. package com.library.service;
8. import com.library.repository.BookRepository;
9. public class BookService {
10. private BookRepository bookRepository;
11. // Setter injection (required for Spring IoC)
12. public void setBookRepository(BookRepository bookRepository) {
13. this.bookRepository = bookRepository;
14. }
15. public void addBook(String title) {
16. System.***out***.println("Adding book: " + title);
17. bookRepository.saveBook(title);
18. }
19. }
    * **Create Spring XML Configuration**

* In src/main/resources → create **applicationContext.xml**

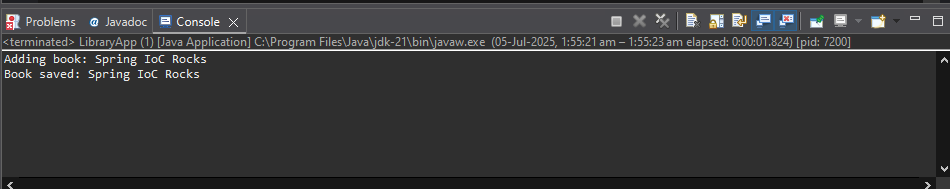
1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans xmlns="http://www.springframework.org/schema/beans"
3. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4. xsi:schemaLocation="http://www.springframework.org/schema/beans
5. http://www.springframework.org/schema/beans/spring-beans.xsd">
6. <!-- BookRepository Bean -->
7. <bean id="bookRepository" class="com.library.repository.BookRepository" />
8. <!-- BookService Bean with Setter Injection -->
9. <bean id="bookService" class="com.library.service.BookService">
10. <property name="bookRepository" ref="bookRepository" />
11. </bean>
12. </beans>
    * **Create Main Class to Run**

📁 com.library.main.**LibraryApp**

1. package com.library.main;
2. import com.library.service.BookService;
3. import org.springframework.context.ApplicationContext;
4. import org.springframework.context.support.ClassPathXmlApplicationContext;
5. public class LibraryApp {
6. public static void main(String[] args) {
7. ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
8. BookService bookService = (BookService) context.getBean("bookService");
9. bookService.addBook("Spring IoC Rocks");
10. }
11. }
    * **Run the Application**

Right-click on LibraryApp.java → Run As → Java Application

* + **OUTPUT:**



**Exercise 7: Implementing Constructor and Setter Injection**

**In Eclipse:**

* File → New → Maven Project
  + Package name **-> /LibraryComboInjection**
  + Add Spring Dependencies to **pom.xml**

1. <project xmlns="http://maven.apache.org/POM/4.0.0"
2. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3. xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
4. https://maven.apache.org/xsd/maven-4.0.0.xsd">
5. <modelVersion>4.0.0</modelVersion>
6. <groupId>com.library</groupId>
7. <artifactId>LibraryComboInjection</artifactId>
8. <version>0.0.1-SNAPSHOT</version>
9. <properties>
10. <maven.compiler.source>1.8</maven.compiler.source>
11. <maven.compiler.target>1.8</maven.compiler.target>
12. </properties>
13. <dependencies>
14. <dependency>
15. <groupId>org.springframework</groupId>
16. <artifactId>spring-context</artifactId>
17. <version>5.3.32</version>
18. </dependency>
19. <dependency>
20. <groupId>commons-logging</groupId>
21. <artifactId>commons-logging</artifactId>
22. <version>1.2</version>
23. </dependency>
24. </dependencies>
25. </project>

***✅ Right-click the project → Maven → Update Project  
 Make sure dependencies are downloaded***

* + **Create Packages and Classes:**
* ***Create 3 packages:***
* Right-click src/main/java → New → Package
* Create:
* com.library.repository
* com.library.service
* com.library.main
  + BookRepository.java 🡪 com.library.repository

package com.library.repository;

public class BookRepository {

public void saveBook(String title) {

System.***out***.println("Book saved: " + title);

}

}

* + BookService.java 🡪 com.library.service

package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

private BookRepository bookRepository;

private String serviceName;

// Constructor Injection

public BookService(String serviceName) {

this.serviceName = serviceName;

System.***out***.println("Constructor Injection: " + serviceName);

}

// Setter Injection

public void setBookRepository(BookRepository bookRepository) {

this.bookRepository = bookRepository;

}

public void addBook(String title) {

System.***out***.println(serviceName + " - Adding book: " + title);

bookRepository.saveBook(title);

}

}

* + LibraryApp.java 🡪 com.library.main

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("Spring Combo Injection");

}

}

* + Create XML Configuration File.

* src/main/resources → Create **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">

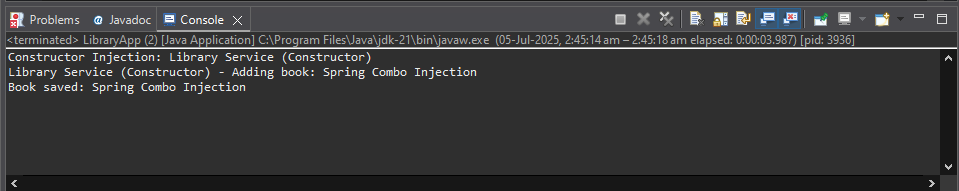
<constructor-arg value="Library Service (Constructor)" />

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

</bean>

</beans>

* + Run the Application.
* Right-click LibraryApp.java → **Run As → Java Application**
  + **OUTPUT:**

****