

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:

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

2. Configure the Application Context:

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

3. Define Service and Repository Classes:

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

4. Run the Application:

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

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure with files like `LibraryApp.java`, `BookService.java`, `BookRepository.java`, `applicationContext.xml`, and `pom.xml`.
- Code Editor:** Displays `LibraryApp.java` containing Java code to run a Spring application using `ClassPathXmlApplicationContext`.
- Terminal:** Shows the command line output of running `mvn exec:java`, which builds the project and runs the application, adding a book to the repository.
- Status Bar:** Shows the current file is `Current File (Spring core and Maven)` and Java is ready.

```

    package com.library;
    import org.springframework.context.ApplicationContext;
    import org.springframework.context.support.ClassPathXmlApplicationContext;
    import com.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(bookName:"Spring in Action");
        }
    }
  
```

```

PS D:\Cognizant\Spring core and Maven\demo> mvn exec:java
[INFO] Scanning for projects...
[INFO] [INFO] ----- < com.example:demo > -----
[INFO] Building demo 1.0-SNAPSHOT
[INFO] from pom.xml
[INFO] -----[ jar ]-----
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.669 s
[INFO] Finished at: 2025-07-04T12:41:49+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\demo>
  
```

```

File Edit Selection View Go Run Terminal Help < -> Spring core and Maven
EXPLORER J LibraryApp.java 1 J BookService.java X J BookRepository.java ● applicationContext.xml pom.xml
demo > src > main > java > com > service > J BookService.java > ...
1 package com.service;
2
3 import com.repository.BookRepository;
4
5 public class BookService {
6     private BookRepository bookRepository;
7
8     // Setter for DI
9     public void setBookRepository(BookRepository bookRepository) {
10         this.bookRepository = bookRepository;
11     }
12
13     public void addBook(String bookName) {
14         System.out.println("Adding book: " + bookName);
15         bookRepository.saveBook(bookName);
16     }
17
18 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 2 SQL HISTORY TASK MONITOR

PS D:\Cognizant\Spring core and Maven\demo> mvn exec:java
[INFO] Scanning for projects...
[INFO] [INFO] --- com.example:demo:1.0-SNAPSHOT
[INFO] Building demo 1.0-SNAPSHOT
[INFO] from pom.xml
[INFO] [INFO] [jar]
[INFO] [INFO] --- exec:3.1.0:java (default-cli) @ demo ---
[INFO] Adding book: Spring in Action
[INFO] Book "Spring in Action" saved to the repository.
[INFO] [INFO] BUILD SUCCESS
[INFO] [INFO] Total time: 0.669 s
[INFO] Finished at: 2025-07-04T12:41:49+05:30
[INFO] [INFO] [INFO] PS D:\Cognizant\Spring core and Maven\demo>

Ln 18, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

```

File Edit Selection View Go Run Terminal Help < -> Spring core and Maven
EXPLORER J LibraryApp.java 1 J BookService.java J BookRepository.java ● applicationContext.xml pom.xml
demo > src > main > java > com > repository > J BookRepository.java > {} com.repository
1 package com.repository;
2
3 public class BookRepository {
4     public void saveBook(String bookName) {
5         System.out.println("Book " + bookName + " saved to the repository.");
6     }
7
8 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 2 SQL HISTORY TASK MONITOR

PS D:\Cognizant\Spring core and Maven\demo> mvn exec:java
[INFO] Scanning for projects...
[INFO] [INFO] --- com.example:demo:1.0-SNAPSHOT
[INFO] Building demo 1.0-SNAPSHOT
[INFO] from pom.xml
[INFO] [INFO] [jar]
[INFO] [INFO] --- exec:3.1.0:java (default-cli) @ demo ---
[INFO] Adding book: Spring in Action
[INFO] Book "Spring in Action" saved to the repository.
[INFO] [INFO] BUILD SUCCESS
[INFO] [INFO] Total time: 0.669 s
[INFO] Finished at: 2025-07-04T12:41:49+05:30
[INFO] [INFO] [INFO] PS D:\Cognizant\Spring core and Maven\demo>

Ln 1, Col 23 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

The screenshot shows a Java Spring application named "LibraryApp" in VS Code. The project structure on the left includes files like `applicationContext.xml`, `BookService.java`, and `BookRepository.java`. The code editor on the right displays `applicationContext.xml` with annotations and XML configurations. The terminal at the bottom shows the Maven command `mvn exec:java` being run, resulting in a successful build.

```
<?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
                           https://www.springframework.org/schema/beans/spring-beans.xsd">
    <bean id="bookRepository" class="com.repository.BookRepository"/>
    <bean id="bookService" class="com.service.BookService">
        <property name="bookRepository" ref="bookRepository" />
    </bean>
</beans>
```

```
PS D:\Cognizant\Spring core and Maven\demo> PS D:\Cognizant\Spring core and Maven\demo> mvn exec:java
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.example:demo >-----
[INFO] Building demo 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time:  0.669 s
[INFO] Finished at: 2025-07-04T12:41:49+05:30
[INFO]
PS D:\Cognizant\Spring core and Maven\demo> [
```

The screenshot shows the Eclipse IDE interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Toolbar:** Includes icons for New, Open, Save, Cut, Copy, Paste, Find, Replace, and others.
- Left Sidebar (Project Explorer):**
 - SPRING CORE AND MAVEN
 - .metadata
 - .vscode
 - demo
 - src
 - main
 - java.com
 - library (LibraryApp.java)
 - repository (BookRepository.java)
 - service (BookService.java)
 - resources (applicationContext.xml)
 - test
 - target
 - pom.xml
 - LibraryManagement
 - .project
 - OUTLINE
 - TIMELINE
 - VS CODE PETS
 - JAVA PROJECTS
 - MAVEN
- Central Area (Code Editor):** The code editor displays the `pom.xml` file for the `LibraryApp` project. The file includes dependencies for Spring Core and a plugin configuration for the `exec-maven-plugin`.
- Bottom Status Bar:** Ls 36, Col 9, Spaces: 4, UTF-8, LF, XML, Go Live, Prettier.

```
Adding book: Spring in Action
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  0.669 s
[INFO] Total time:  0.669 s
[INFO] Finished at: 2025-07-04T12:41:49+05:30
[INFO] -----
```

Exercise 2: Implementing Dependency Injection

Scenario:

In the library management application, you need to manage the dependencies between the BookService and BookRepository classes using Spring's IoC and DI.

Steps:

1. **Modify the XML Configuration:**
 - Update `applicationContext.xml` to wire `BookRepository` into `BookService`.
2. **Update the BookService Class:**
 - Ensure that `BookService` class has a setter method for `BookRepository`.
3. **Test the Configuration:**
 - Run the `LibraryManagementApplication` main class to verify the dependency injection.

File Edit Selection View Go Run Terminal Help

Spring core and Maven

EXPLORER

SRC CORE AND MAVEN

- .metadata
- .vscode
- demo
- src
 - main
 - java\com
 - LibraryApp.java
 - BookRepository.java
 - BookService.java
 - resources
 - applicationContext.xml
 - test
 - pom.xml
- LibraryManagement
- .project

LibraryApp.java

```

1 package com.library;
2
3 import org.springframework.context.ApplicationContext;
4 import org.springframework.context.support.ClassPathXmlApplicationContext;
5 import com.service.BookService;
6
7 public class LibraryApp {
8     Run [Debug]
9     public static void main(String[] args) {
10         ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
11         BookService bookService = context.getBean("bookService", BookService.class);
12         bookService.addBook(bookName: "Spring in Action");
13
14         ((ClassPathXmlApplicationContext) context).close();
15     }
16 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

[INFO] [INFO] --- compiler:3.13.0:compile (default-compile) @ demo --- [INFO] Recompiling the module because of changed source code. [WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent! [INFO] Compiling 3 source files with javac [debug target 17] to target\classes [WARNING] system modules path not set in conjunction with -source 17 [INFO] [INFO] --- exec:3.1.0:java (default-cli) @ demo --- Adding book: Spring in Action Book "Spring in Action" saved to the repository. [INFO] [INFO] BUILD SUCCESS [INFO] ----- [INFO] Total time: 1.573 s [INFO] Finished at: 2025-07-04T13:13:05+05:30 [INFO] -----

PS D:\Cognizant\Spring core and Maven\demo> [Java: Ready]

In 11, Col 46 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

File Edit Selection View Go Run Terminal Help

Spring core and Maven

EXPLORER

SRC CORE AND MAVEN

- .metadata
- .vscode
- demo
- src
 - main
 - java\com
 - library
 - LibraryApp.java
 - repository
 - BookRepository.java
 - service
 - BookService.java
 - resources
 - applicationContext.xml
 - test
 - pom.xml
- LibraryManagement
- .project

BookService.java

```

1 package com.service;
2
3 import com.repository.BookRepository;
4
5 public class BookService {
6     private BookRepository bookRepository;
7
8     public void setBookRepository(BookRepository bookRepository) {
9         this.bookRepository = bookRepository;
10    }
11
12    public void addBook(String bookName) {
13        System.out.println("Adding book: " + bookName);
14        bookRepository.saveBook(bookName);
15    }
16 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

[INFO] [INFO] --- compiler:3.13.0:compile (default-compile) @ demo --- [INFO] Recompiling the module because of changed source code. [WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent! [INFO] Compiling 3 source files with javac [debug target 17] to target\classes [WARNING] system modules path not set in conjunction with -source 17 [INFO] [INFO] --- exec:3.1.0:java (default-cli) @ demo --- Adding book: Spring in Action Book "Spring in Action" saved to the repository. [INFO] [INFO] BUILD SUCCESS [INFO] ----- [INFO] Total time: 1.573 s [INFO] Finished at: 2025-07-04T13:13:05+05:30 [INFO] -----

PS D:\Cognizant\Spring core and Maven\demo> [Java: Ready]

In 7, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

VS Code interface showing the BookRepository.java file:

```

1 package com.repository;
2
3 public class BookRepository {
4     public void saveBook(String bookName) {
5         System.out.println("Book '" + bookName + "' saved to the repository.");
6     }
7 }

```

The Explorer sidebar shows the project structure:

- SPRING CORE AND MAVEN
- .metadata
- .vscode
- demo
- src
 - main
 - java\com
 - library
 - J LibraryApp.java
 - repository
 - service
 - resources
 - applicationContext.xml
- test
- target
- pom.xml

The TERMINAL tab shows the build output:

```

[INFO] --- compiler:3.13.0:compile (default-compile) @ demo ---
[INFO] Recompiling the module because of changed source code.
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 3 source files with javac [debug target 17] to target\classes
[WARNING] system modules path not set in conjunction with -source 17
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.573 s
[INFO] Finished at: 2025-07-04T13:13:05+05:30
[INFO] -----

```

VS Code interface showing the applicationContext.xml file:

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="https://www.springframework.org/schema/beans"
3         xmlns:xsi="https://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
7     <bean id="bookRepository" class="com.repository.BookRepository" />
8
9     <bean id="bookService" class="com.service.BookService">
10        <property name="bookRepository" ref="bookRepository" />
11    </bean>
12
13 </beans>
14
15

```

The Explorer sidebar shows the project structure:

- SPRING CORE AND MAVEN
- .metadata
- .vscode
- demo
- src
 - main
 - java\com
 - library
 - J LibraryApp.java
 - repository
 - service
 - resources
 - applicationContext.xml
- test
- target
- pom.xml

The TERMINAL tab shows the build output:

```

[INFO] --- compiler:3.13.0:compile (default-compile) @ demo ---
[INFO] Recompiling the module because of changed source code.
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 3 source files with javac [debug target 17] to target\classes
[WARNING] system modules path not set in conjunction with -source 17
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.573 s
[INFO] Finished at: 2025-07-04T13:13:05+05:30
[INFO] -----

```

```

<?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.example</groupId>
    <artifactId>demo</artifactId>
    <version>1.0-SNAPSHOT</version>
    <properties>
        <maven.compiler.source>17</maven.compiler.source>
        <maven.compiler.target>17</maven.compiler.target>
    </properties>
    <dependencies>
        <!-- Spring Core Dependency -->
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.32</version> <!-- Use latest stable -->
        </dependency>
    </dependencies>
    <build>
        <plugins>
            <plugin>
                <groupId>org.codehaus.mojo</groupId>
                <artifactId>exec-maven-plugin</artifactId>
                <version>3.1.0</version>
                <configuration>
                    <mainClass>com.library.LibraryApp</mainClass>
                </configuration>
            </plugin>
        </plugins>
    </build>
</project>

```

```

[INFO] -----
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 3 source files with javac [debug target 17] to target\classes
[WARNING] system modules path not set in conjunction with -source 17
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO] -----
[WARNING] system modules path not set in conjunction with -source 17
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO] -----
[INFO] --- exec:3.1.0:java (default-cli) @ demo ---
Adding book: Spring in Action
Book "Spring in Action" saved to the repository.
[INFO] -----
[INFO] BUILD SUCCESS
Book "Spring in Action" saved to the repository.
[INFO] -----
[INFO] BUILD SUCCESS

```

Exercise 3: Implementing Logging with Spring AOP

Scenario:

The library management application requires logging capabilities to track method execution times.

Steps:

1. Add Spring AOP Dependency:

- Update **pom.xml** to include Spring AOP dependency.

2. Create an Aspect for Logging:

- Create a package **com.library.aspect** and add a class **LoggingAspect** with a method to log execution times.

3. Enable AspectJ Support:

- Update **applicationContext.xml** to enable **AspectJ** support and register the aspect.

4. Test the Aspect:

- Run the **LibraryManagementApplication** main class and observe the console for log messages indicating method execution times.

The screenshot shows a VS Code interface with the following details:

- Project Structure:** The Explorer view shows a project named "demo" containing "src" and "resources" folders. "src" contains "main" (with "java", "com", "library", "aspect", "LoggingAspect.java"), "service" (with "BookService.java"), and "resources" (with "applicationContext.xml"). "resources" also contains "pom.xml".
- Code Editor:** The main editor shows the content of **applicationContext.xml**:


```

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:aop="http://www.springframework.org/schema/aop"
       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
           http://www.springframework.org/schema/context
           http://www.springframework.org/schema/context/spring-context.xsd
           http://www.springframework.org/schema/aop
           http://www.springframework.org/schema/aop/spring-aop.xsd">
    <context:component-scan base-package="com.library" />
    <aop:aspectj-autoproxy />
</beans>
```
- Terminal:** The bottom terminal window shows the output of the application run:


```
[INFO] --- exec:3.1.0:java (default-cli) @ aspect ---
Book issued.
void com.library.service.BookService.issueBook() executed in 523ms
Book returned.
void com.library.service.BookService.returnBook() executed in 306ms
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.930 s
[INFO] Finished at: 2025-07-04T14:28:55+05:30
[INFO] -----
```

```

<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.example</groupId>
    <artifactId>aspect</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>17</maven.compiler.source>
        <maven.compiler.target>17</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>

    <dependencies>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.30</version>
        </dependency>

        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-aspects</artifactId>
        </dependency>
    </dependencies>

```

[INFO] --- exec:3.1.0:java (default-cli) @ aspect ---
Book issued.
void com.library.service.BookService.issueBook() executed in 523ms
Book returned.
void com.library.service.BookService.returnBook() executed in 306ms
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.930 s
[INFO] Finished at: 2025-07-04T14:28:55+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\logging>

```

package com.library.service;

import org.springframework.stereotype.Service;

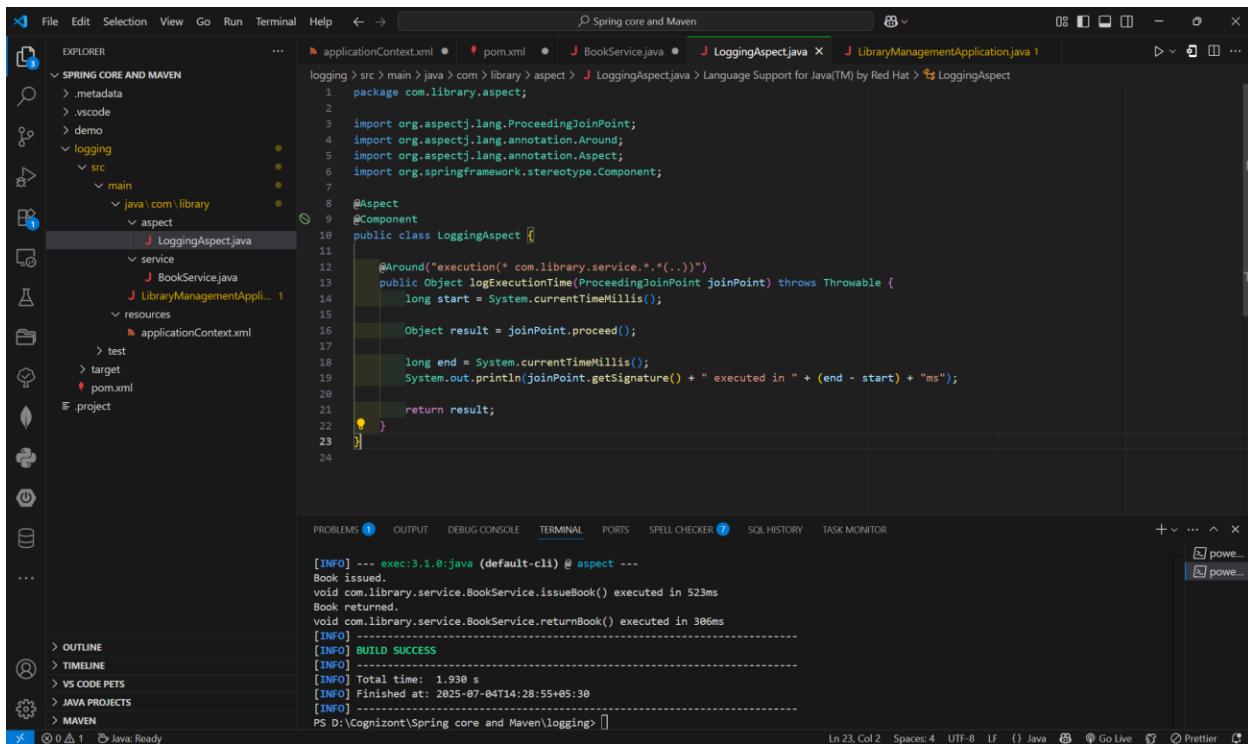
@Service
public class BookService {

    public void issueBook() throws InterruptedException {
        Thread.sleep(millis:500);
        System.out.println(x:"Book issued.");
    }

    public void returnBook() throws InterruptedException {
        Thread.sleep(millis:300);
        System.out.println(x:"Book returned.");
    }
}

```

[INFO] --- exec:3.1.0:java (default-cli) @ aspect ---
Book issued.
void com.library.service.BookService.issueBook() executed in 523ms
Book returned.
void com.library.service.BookService.returnBook() executed in 306ms
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.930 s
[INFO] Finished at: 2025-07-04T14:28:55+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\logging>



```

logging > src > main > java > com > library > aspect > LoggingAspect.java > Language Support for Java(TM) by Red Hat > LoggingAspect
1 package com.library.aspect;
2
3 import org.aspectj.lang.ProceedingJoinPoint;
4 import org.aspectj.lang.annotation.Around;
5 import org.aspectj.lang.annotation.Aspect;
6 import org.springframework.stereotype.Component;
7
8 @Aspect
9 @Component
10 public class LoggingAspect {
11
12     @Around("execution(* com.library.service.*(..))")
13     public Object logExecutionTime(ProceedingJoinPoint joinPoint) throws Throwable {
14         long start = System.currentTimeMillis();
15
16         Object result = joinPoint.proceed();
17
18         long end = System.currentTimeMillis();
19         System.out.println(joinPoint.getSignature() + " executed in " + (end - start) + "ms");
20
21         return result;
22     }
23 }

```

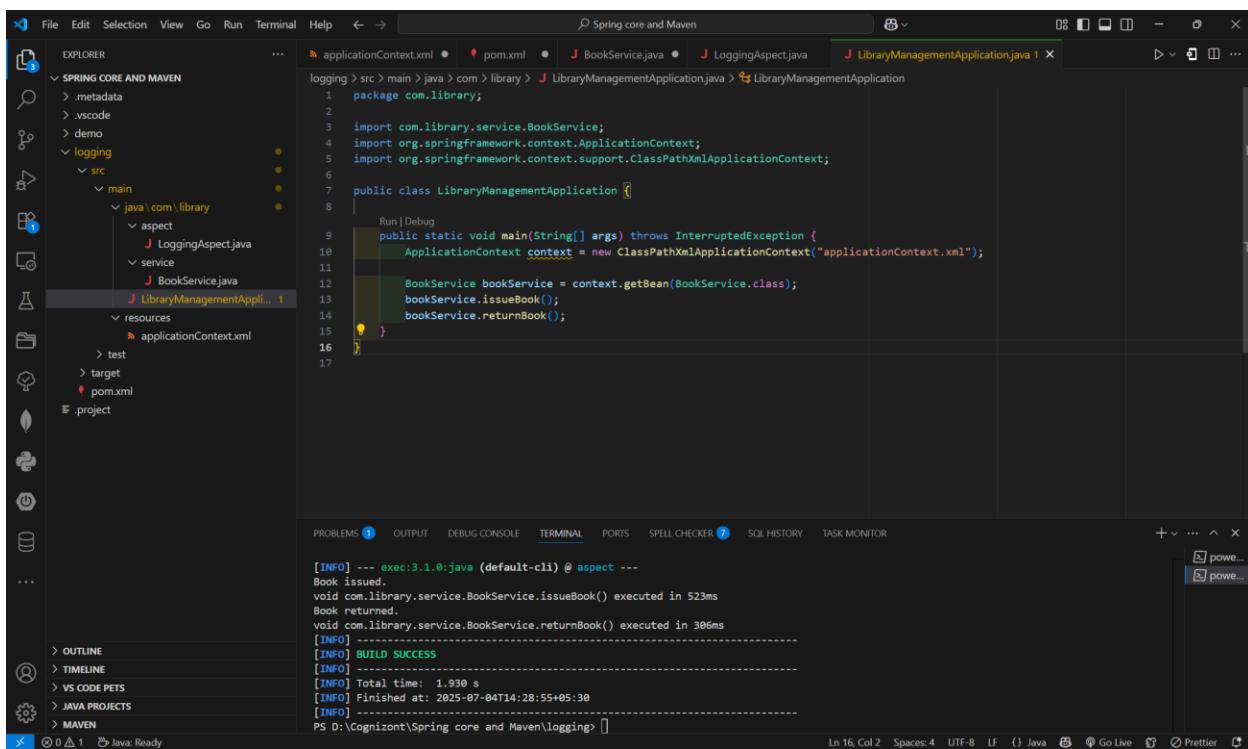
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 7 SQL HISTORY TASK MONITOR

```

[INFO] --- exec:3.1.0:java (default-cli) @ aspect ---
Book issued.
void com.library.service.BookService.issueBook() executed in 523ms
Book returned.
void com.library.service.BookService.returnBook() executed in 306ms
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.930 s
[INFO] Finished at: 2025-07-04T14:28:55+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\logging> []

```

Ln 23, Col 2 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡



```

logging > src > main > java > com > library > LibraryManagementApplication.java > Language Support for Java(TM) by Red Hat > LibraryManagementApplication
1 package com.library;
2
3 import com.library.service.BookService;
4 import org.springframework.context.ApplicationContext;
5 import org.springframework.context.support.ClassPathXmlApplicationContext;
6
7 public class LibraryManagementApplication {
8
9     Run|Debug
10    public static void main(String[] args) throws InterruptedException {
11        ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
12
13        BookService bookService = context.getBean(BookService.class);
14        bookService.issueBook();
15        bookService.returnBook();
16    }
17

```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 7 SQL HISTORY TASK MONITOR

```

[INFO] --- exec:3.1.0:java (default-cli) @ aspect ---
Book issued.
void com.library.service.BookService.issueBook() executed in 523ms
Book returned.
void com.library.service.BookService.returnBook() executed in 306ms
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.930 s
[INFO] Finished at: 2025-07-04T14:28:55+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\logging> []

```

Ln 16, Col 2 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

```

PS D:\Cognizont\Spring core and Maven\logging> mvn exec:java
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.example:aspect >-----
[INFO] Building aspect 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ aspect ---
Book issued.
void com.library.service.BookService.issueBook() executed in 523ms
Book returned.
void com.library.service.BookService.returnBook() executed in 306ms
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  1.930 s
[INFO] Finished at: 2025-07-04T14:28:55+05:30
[INFO] -----
PS D:\Cognizont\Spring core and Maven\logging> █

```

Exercise 4: Creating and Configuring a Maven Project

Scenario:

You need to set up a new Maven project for the library management application and add Spring dependencies.

Steps:

1. **Create a New Maven Project:**
 - Create a new Maven project named **LibraryManagement**.
2. **Add Spring Dependencies in pom.xml:**
 - Include dependencies for Spring Context, Spring AOP, and Spring WebMVC.
3. **Configure Maven Plugins:**
 - Configure the Maven Compiler Plugin for Java version 1.8 in the pom.xml file.

```

4. <?xml version="1.0" encoding="UTF-8"?>
5. <project xmlns="http://maven.apache.org/POM/4.0.0"
6.           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
7.           xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
8.                               http://maven.apache.org/xsd/maven-4.0.0.xsd">
9.

```

```
10.    <modelVersion>4.0.0</modelVersion>
11.
12.    <groupId>com.library</groupId>
13.    <artifactId>LibraryManagement</artifactId>
14.    <version>1.0-SNAPSHOT</version>
15.
16.    <properties>
17.        <maven.compiler.source>1.8</maven.compiler.source>
18.        <maven.compiler.target>1.8</maven.compiler.target>
19.    </properties>
20.
21.    <dependencies>
22.        <dependency>
23.            <groupId>org.springframework</groupId>
24.            <artifactId>spring-context</artifactId>
25.            <version>5.3.30</version>
26.        </dependency>
27.
28.        <dependency>
29.            <groupId>org.springframework</groupId>
30.            <artifactId>spring-aspects</artifactId>
31.            <version>5.3.30</version>
32.        </dependency>
33.
34.        <dependency>
35.            <groupId>org.springframework</groupId>
36.            <artifactId>spring-webmvc</artifactId>
37.            <version>5.3.30</version>
38.        </dependency>
39.
40.        <dependency>
41.            <groupId>org.aspectj</groupId>
42.            <artifactId>aspectjweaver</artifactId>
43.            <version>1.9.21</version>
44.        </dependency>
45.    </dependencies>
46.
47.    <build>
48.        <plugins>
49.            <plugin>
50.                <groupId>org.apache.maven.plugins</groupId>
51.                <artifactId>maven-compiler-plugin</artifactId>
52.                <version>3.11.0</version>
53.                <configuration>
54.                    <source>1.8</source>
```

```

55.          <target>1.8</target>
56.      </configuration>
57.  </plugin>
58.
59.  <plugin>
60.      <groupId>org.codehaus.mojo</groupId>
61.      <artifactId>exec-maven-plugin</artifactId>
62.      <version>3.1.0</version>
63.      <configuration>
64.          <mainClass>com.library.LibraryManagementApplication</m
ainClass>
65.      </configuration>
66.  </plugin>
67. </plugins>
68. </build>
69.
70.</project>
71.

```

```

[INFO] Changes detected - recompiling the module! :dependency
[INFO] Changes detected - recompiling the module! :dependency
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO]
[INFO] --- surefire:3.2.5:test (default-test) @ LibraryManagement ---
[INFO]
[INFO]
[INFO] --- jar:3.4.1:jar (default-jar) @ LibraryManagement ---
[INFO] Building jar: D:\Cognizont\Spring core and Maven\librarymanagement\target\LibraryManagement-1.0-SNAPSHOT.jar
[INFO]
[INFO] --- install:3.1.2:install (default-install) @ LibraryManagement ---
[INFO]
[INFO] --- install:3.1.2:install (default-install) @ LibraryManagement ---
[INFO] --- install:3.1.2:install (default-install) @ LibraryManagement ---
[INFO] Installing D:\Cognizont\Spring core and Maven\librarymanagement\pom.xml to C:\Users\Del1\.m2\repository\com\library\LibraryManagement
[INFO] Installing D:\Cognizont\Spring core and Maven\librarymanagement\pom.xml to C:\Users\Del1\.m2\repository\com\library\LibraryManagement
\1.0-SNAPSHOT\LibraryManagement-1.0-SNAPSHOT.pom
[INFO] Installing D:\Cognizont\Spring core and Maven\librarymanagement\target\LibraryManagement-1.0-SNAPSHOT.jar to C:\Users\Del1\.m2\reposito
ry\com\library\LibraryManagement\1.0-SNAPSHOT\LibraryManagement-1.0-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  6.831 s
[INFO] Finished at: 2025-07-04T15:14:40+05:30
[INFO] Total time:  6.831 s
[INFO] Total time:  6.831 s
[INFO] Finished at: 2025-07-04T15:14:40+05:30

```

Exercise 5: Configuring the Spring IoC Container

Scenario:

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

Steps:

1. Create Spring Configuration File:

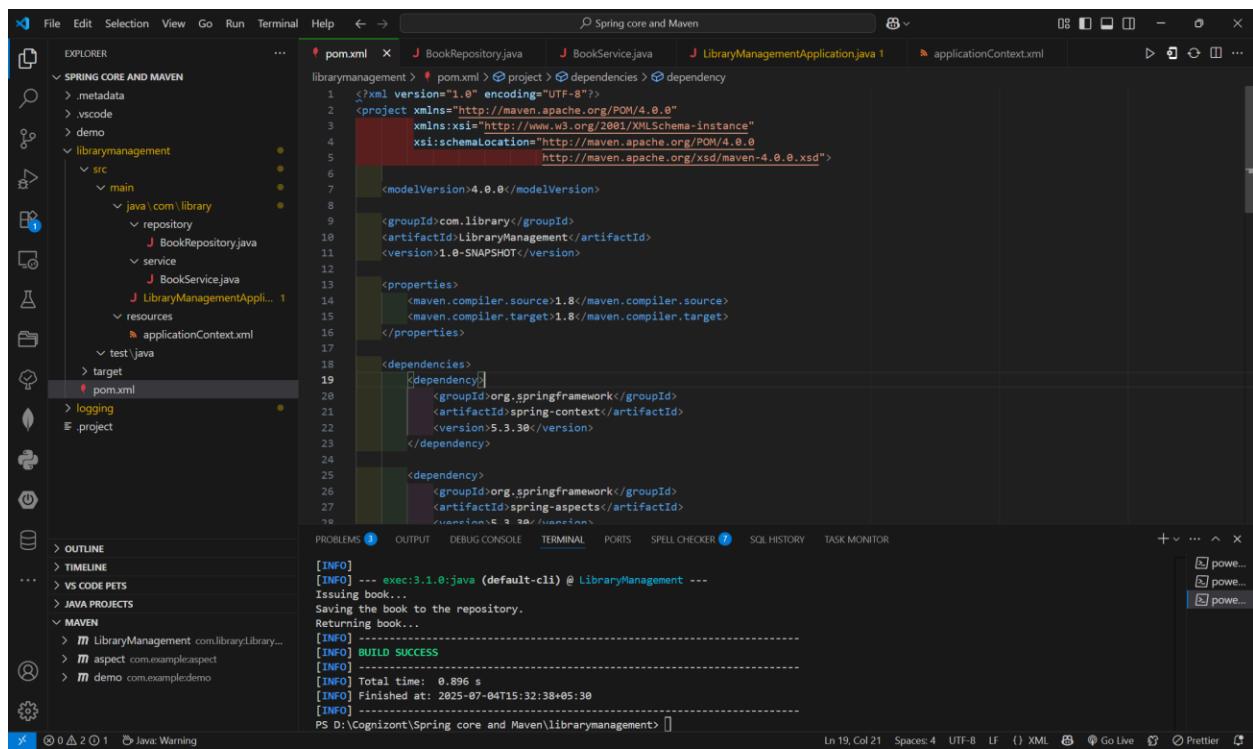
- Create an XML configuration file named **applicationContext.xml** in the **src/main/resources** directory.
- Define beans for **BookService** and **BookRepository** in the XML file.

2. Update the BookService Class:

- Ensure that the **BookService** class has a setter method for **BookRepository**.

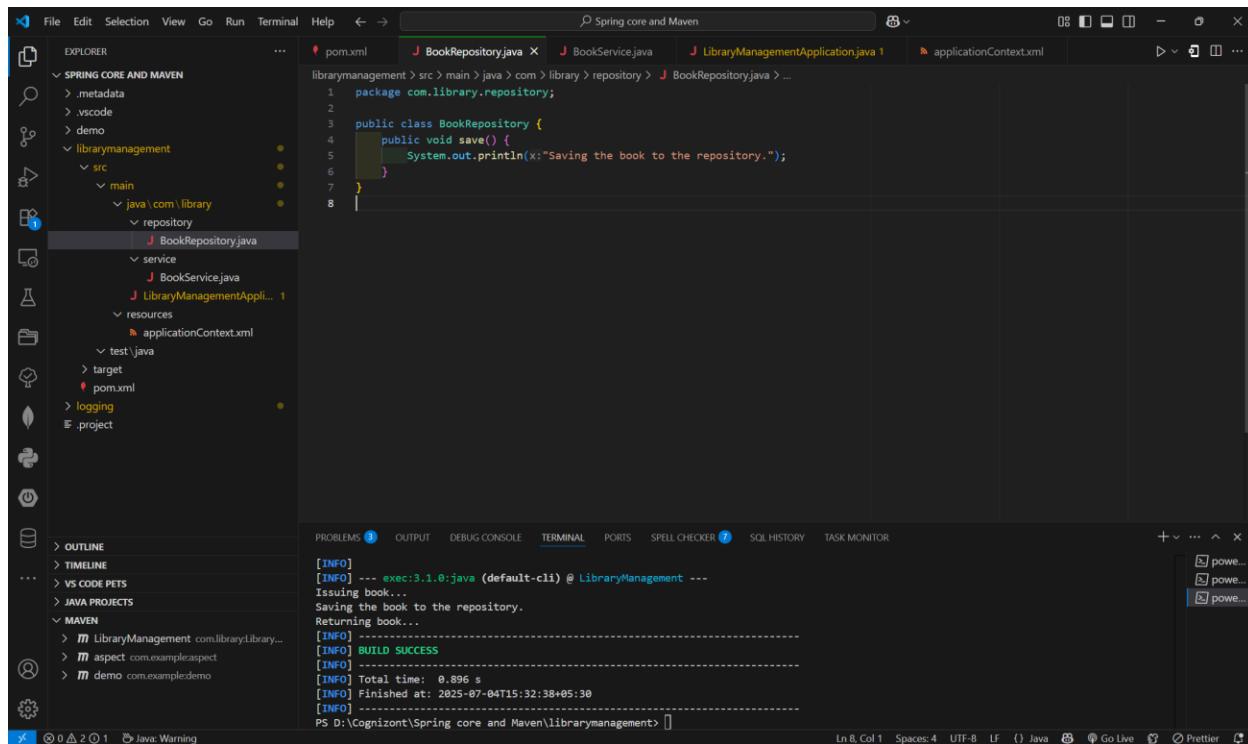
3. Run the Application:

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



The screenshot shows the VS Code interface with the following details:

- Project Explorer:** Shows the project structure under "librarymanagement". It includes a "src" folder containing "main" (with "java" and "resources" subfolders), "test", and "target". "pom.xml" is also listed.
- Maven View:** Shows the current build configuration.
- Code Editor:** Displays the content of "pom.xml". The XML code defines a Maven project with group ID "com.library", artifact ID "libraryManagement", version "1.0-SNAPSHOT", and properties for compiler versions. It lists dependencies for Spring Framework (version 5.3.30) and Spring Aspects.
- Terminal:** Shows the command-line output of the Maven build process. It includes the command "exec:3.1.0:java (default-cli) @ LibraryManagement", the message "Issuing book...", "Saving the book to the repository.", "Returning book...", and the final message "BUILD SUCCESS". The total build time is 0.896 seconds, and it finished at 2025-07-04T15:32:38+05:30.
- Status Bar:** Shows the path "PS D:\Cognizant\Spring core and Maven\librarymanagement>" and various status indicators like "In 19, Col 21", "Spaces: 4", "UTF-8", "LF", "XML", "Go Live", "Prettier", and "Java: Warning".



Spring core and Maven

```

1 package com.library.repository;
2
3 public class BookRepository {
4     public void save() {
5         System.out.println("Saving the book to the repository.");
6     }
7 }

```

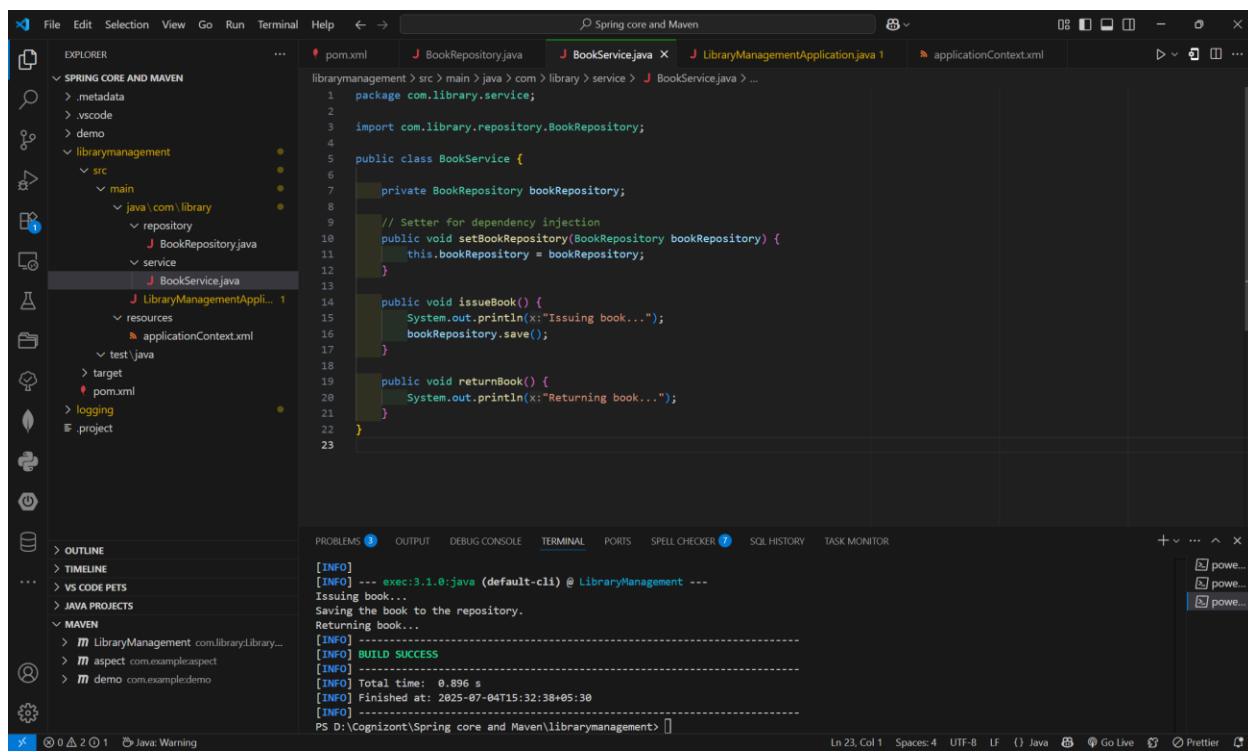
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

```

[INFO] [INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Issuing book...
Saving the book to the repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:32:38+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> []

```

In 8, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡



Spring core and Maven

```

1 package com.library.service;
2
3 import com.library.repository.BookRepository;
4
5 public class BookService {
6     private BookRepository bookRepository;
7
8     // Setter for dependency injection
9     public void setBookRepository(BookRepository bookRepository) {
10         this.bookRepository = bookRepository;
11     }
12
13     public void issueBook() {
14         System.out.println("Issuing book...");
15         bookRepository.save();
16     }
17
18     public void returnBook() {
19         System.out.println("Returning book...");
20     }
21 }
22
23

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

```

[INFO] [INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Issuing book...
Saving the book to the repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:32:38+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> []

```

In 23, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

VS Code interface showing the LibraryManagementApplication.java file:

```

    package com.library;
    import com.library.service.BookService;
    import org.springframework.context.ApplicationContext;
    import org.springframework.context.support.ClassPathXmlApplicationContext;
    public class LibraryManagementApplication {
        public static void main(String[] args) {
            ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
            BookService bookService = context.getBean(BookService.class);
            bookService.issueBook();
            bookService.returnBook();
        }
    }
  
```

Output terminal shows:

```

[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Issuing book...
Saving the book to the repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:32:38+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> 
  
```

VS Code interface showing the applicationContext.xml file:

```

<?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
                           https://www.springframework.org/schema/beans/spring-beans.xsd">
    <!-- Define BookRepository Bean -->
    <bean id="bookRepository" class="com.library.repository.BookRepository" />
    <!-- Define BookService Bean with setter injection -->
    <bean id="bookService" class="com.library.service.BookService">
        <property name="bookRepository" ref="bookRepository" />
    </bean>
</beans>
  
```

Output terminal shows:

```

[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Issuing book...
Saving the book to the repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:32:38+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> 
  
```

```

[INFO] -----
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Issuing book...
Saving the book to the repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  0.896 s
[INFO] Finished at: 2025-07-04T15:32:38+05:30
[INFO] -----
PS D:\Cognizont\Spring core and Maven\librarymanagement> []

```

Exercise 6: Configuring Beans with Annotations

Scenario:

You need to simplify the configuration of beans in the library management application using annotations.

Steps:

1. **Enable Component Scanning:**
 - Update **applicationContext.xml** to include component scanning for the **com.library** package.
2. **Annotate Classes:**
 - Use **@Service** annotation for the **BookService** class.
 - Use **@Repository** annotation for the **BookRepository** class.
3. **Test the Configuration:**
 - Run the **LibraryManagementApplication** main class to verify the annotation-based configuration.

```

<?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>libraryManagement</artifactId>
    <version>1.0-SNAPSHOT</version>
    <properties>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.0</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-aspects</artifactId>
            <version>5.3.0</version>
        </dependency>
    </dependencies>
</project>

```

BookService: Returning book..
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:35:55+05:30
[INFO] -----

```

package com.library.repository;
import org.springframework.stereotype.Repository;
@Repository
public class BookRepository {
    public void save() {
        System.out.println("BookRepository: Saving the book.");
    }
}

```

BookService: Returning book..
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:35:55+05:30
[INFO] -----

File Edit Selection View Go Run Terminal Help ← → Spring core and Maven pom.xml BookRepository.java BookService.java LibraryManagementApplication.java applicationContext.xml

EXPLORER

- SPRING CORE AND MAVEN
 - .metadata
 - .vscode
 - demo
 - librarymanagement
 - src
 - main
 - java\com\library
 - repository BookRepository.java
 - service BookService.java
 - resources applicationContext.xml
 - test.java
 - target
 - pom.xml
 - logging
 - .project

OUTLINE

TIMELINE

VS CODE PETS

JAVA PROJECTS

MAVEN

- M LibraryManagement com.library.Library...
- M aspect com.exampleaspect
- M demo com.exampledemo

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

```
BookService: Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:35:55+05:30
[INFO] -----
```

PS D:\Cognizant\Spring core and Maven\librarymanagement> Ln 22, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

Java: Warning

File Edit Selection View Go Run Terminal Help ← → Spring core and Maven pom.xml BookRepository.java BookService.java LibraryManagementApplication.java applicationContext.xml

EXPLORER

- SPRING CORE AND MAVEN
 - .metadata
 - .vscode
 - demo
 - librarymanagement
 - src
 - main
 - java\com\library
 - repository BookRepository.java
 - service BookService.java
 - LibraryManagementApplication.java
 - resources applicationContext.xml
 - test.java
 - target
 - pom.xml
 - logging
 - .project

OUTLINE

TIMELINE

VS CODE PETS

JAVA PROJECTS

MAVEN

- M LibraryManagement com.library.Library...
- M aspect com.exampleaspect
- M demo com.exampledemo

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

```
BookService: Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.896 s
[INFO] Finished at: 2025-07-04T15:35:55+05:30
[INFO] -----
```

PS D:\Cognizant\Spring core and Maven\librarymanagement> Ln 16, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

Java: Warning

The screenshot shows the VS Code interface with the following details:

- Explorer View:** Shows the project structure under "SPRING CORE AND MAVEN". It includes ".metadata", ".vscode", "demo", "librarymanagement" (with "src" and "resources" subfolders), and "test" (with "java" and "resources" subfolders).
- Editor View:** Displays the `applicationContext.xml` file with code like:


```

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
                           https://www.springframework.org/schema/beans/spring-beans.xsd
                           http://www.springframework.org/schema/context
                           https://www.springframework.org/schema/context/spring-context.xsd">
    <!-- Enable annotation-based component scanning -->
    <context:component-scan base-package="com.library" />
</beans>
```
- Terminal View:** Shows the command `mvn exec:java` being run and its output:


```

PS D:\Cognizont\Spring core and Maven\librarymanagement> mvn exec:java
[INFO] Scanning for projects...
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
BookService: Issuing book...
BookRepository: Saving the book.
BookService: Returning book...
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time:  0.896 s
[INFO] Finished at: 2025-07-04T15:35:55+05:30
[INFO]
```

```

PS D:\Cognizont\Spring core and Maven\librarymanagement> mvn exec:java
[INFO] Scanning for projects...
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
BookService: Issuing book...
BookRepository: Saving the book.
BookService: Returning book...
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] -----[ jar ]-----
[INFO] -----[ jar ]-----
[INFO]
[INFO] Total time:  0.896 s
[INFO] Finished at: 2025-07-04T15:35:55+05:30
[INFO]
```

Exercise 7: Implementing Constructor and Setter Injection

Scenario:

The library management application requires both constructor and setter injection for better control over bean initialization.

Steps:

1. Configure Constructor Injection:

- Update **applicationContext.xml** to configure constructor injection for **BookService**.

2. Configure Setter Injection:

- Ensure that the **BookService** class has a setter method for **BookRepository** and configure it in **applicationContext.xml**.

3. Test the Injection:

- Run the **LibraryManagementApplication** main class to verify both constructor and setter injection.

The screenshot shows the VS Code interface with the following details:

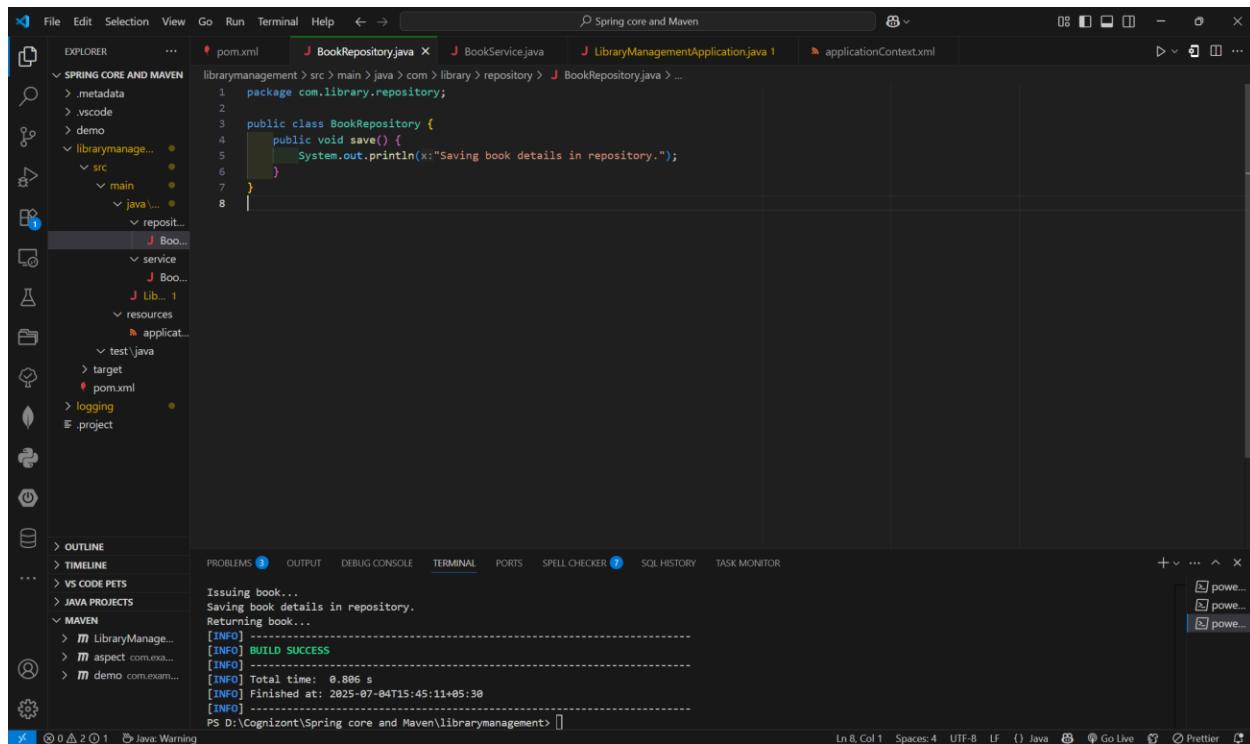
- EXPLORER** view: Shows the project structure under "SPRING CORE AND MAVEN". It includes ".metadata", ".vscode", "demo", and "librarymanagement" (which is expanded) containing "src" (expanded), "main" (expanded), "java" (expanded), "repository" (expanded), "service" (expanded), "resources" (expanded), "application" (expanded), and "test" (expanded). Inside "src/main/java/librarymanagement/repository", there is a file named "BookRepository.java". Inside "src/main/java/librarymanagement/service", there is a file named "BookService.java". Inside "src/test/java", there is a file named "LibraryManagementApplication.java". Below these, there are "pom.xml" files and a ".project" file.
- MIDDLE AREA**: The main workspace shows the content of the "pom.xml" file. The code is as follows:

```

<?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>LibraryManagement</artifactId>
    <version>1.0-SNAPSHOT</version>
    <properties>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.30</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-aspects</artifactId>
            <version>5.3.30</version>
        </dependency>
    </dependencies>

```

- TERMINAL** view: Shows the output of the Maven build command. It includes messages like "Issuing book...", "Saving book details in repository.", "Returning book...", and the final "BUILD SUCCESS" message. The total build time is listed as 0.806s, and it finished at 2025-07-04T15:45:11+05:30.
- STATUS BAR**: Shows the path "PS D:\Cognizont\Spring core and Maven\librarymanagement", file statistics (0△ 2○ 1), Java warning icon, and status indicators for Lint, Spell Checker, and Task Monitor.



File Edit Selection View Go Run Terminal Help

Spring core and Maven

EXPLORER pom.xml BookRepository.java BookService.java LibraryManagementApplication.java applicationContext.xml

```

librarymanagement > src > main > java > com > library > repository > BookRepository.java > ...
1 package com.library.repository;
2
3 public class BookRepository {
4     public void save() {
5         System.out.println("Saving book details in repository.");
6     }
7 }

```

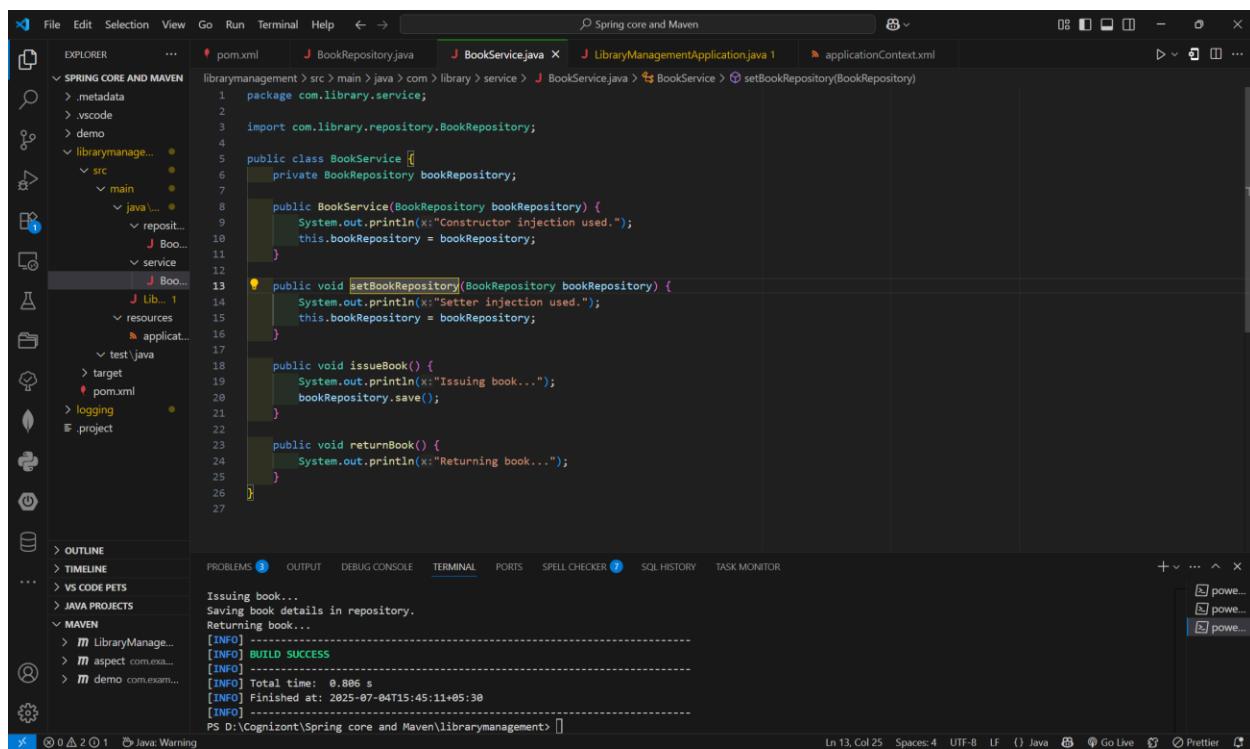
OUTLINE

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

Issuing book...
Saving book details in repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.806 s
[INFO] Finished at: 2025-07-04T15:45:11+05:30
[INFO] -----

PS D:\Cognizant\Spring core and Maven\librarymanagement> |

In 8, Col 1 Spaces: 4 UTF-8 LF () Java Go Live Prettier



File Edit Selection View Go Run Terminal Help

Spring core and Maven

EXPLORER pom.xml BookRepository.java BookService.java LibraryManagementApplication.java applicationContext.xml

```

librarymanagement > src > main > java > com > library > service > BookService.java > BookService > setBookRepository(BookRepository)
1 package com.library.service;
2
3 import com.library.repository.BookRepository;
4
5 public class BookService {
6     private BookRepository bookRepository;
7
8     public BookService(BookRepository bookRepository) {
9         System.out.println("Constructor injection used.");
10        this.bookRepository = bookRepository;
11    }
12
13    public void setBookRepository(BookRepository bookRepository) {
14        System.out.println("Setter injection used.");
15        this.bookRepository = bookRepository;
16    }
17
18    public void issueBook() {
19        System.out.println("Issuing book...");
20        bookRepository.save();
21    }
22
23    public void returnBook() {
24        System.out.println("Returning book...");
25    }
26 }

```

OUTLINE

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

Issuing book...
Saving book details in repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.806 s
[INFO] Finished at: 2025-07-04T15:45:11+05:30
[INFO] -----

PS D:\Cognizant\Spring core and Maven\librarymanagement> |

In 13, Col 25 Spaces: 4 UTF-8 LF () Java Go Live Prettier

```

File Edit Selection View Go Run Terminal Help < > Spring core and Maven pom.xml BookRepository.java BookService.java LibraryManagementApplication.java applicationContext.xml

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

public class LibraryManagementApplication {
    public static void main(String[] args) {
        ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
        BookService bookService = context.getBean(BookService.class);
        bookService.issueBook();
        bookService.returnBook();
    }
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

```

Issuing book...
Saving book details in repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.806 s
[INFO] Finished at: 2025-07-04T15:45:11+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> []

```

In 11, Col 43 Spaces: 4 UTF-8 LF () Java Go Live Prettier

```

File Edit Selection View Go Run Terminal Help < > Spring core and Maven pom.xml BookRepository.java BookService.java LibraryManagementApplication.java 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
                           https://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 ref="bookRepository" />
        <property name="bookRepository" ref="bookRepository" />
    </bean>
</beans>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

```

Issuing book...
Saving book details in repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.806 s
[INFO] Finished at: 2025-07-04T15:45:11+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> []

```

In 12, Col 1 Spaces: 4 UTF-8 LF () XML Go Live Prettier

```

PS D:\Cognizont\Spring core and Maven\librarymanagement> mvn exec:java
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Constructor injection used.
Setter injection used.
Issuing book...
Saving book details in repository.
Returning book...
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  0.806 s
[INFO] Finished at: 2025-07-04T15:45:11+05:30
[INFO] -----
PS D:\Cognizont\Spring core and Maven\librarymanagement> []

```

Exercise 8: Implementing Basic AOP with Spring

Scenario:

The library management application requires basic AOP functionality to separate cross-cutting concerns like logging and transaction management.

Steps:

1. Define an Aspect:

- Create a package **com.library.aspect** and add a class **LoggingAspect**.
- 2. Create Advice Methods:**
- Define advice methods in **LoggingAspect** for logging before and after method execution.
- 3. Configure the Aspect:**
- Update **applicationContext.xml** to register the aspect and enable **AspectJ** auto-proxying.
- 4. Test the Aspect:**
- Run the **LibraryManagementApplication** main class to verify the AOP functionality.

```

<?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>LibraryManagement</artifactId>
    <version>1.0-SNAPSHOT</version>
    <properties>
        <maven.compiler.sources>1.8</maven.compiler.sources>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.30</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-aspects</artifactId>
            <version>5.3.30</version>
        </dependency>
    </dependencies>
</project>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

[INFO] -----
[INFO] Total time: 1.296 s
[INFO] Finished at: 2025-07-04T15:49:16+05:30
[INFO] -----

PS D:\Cognizont\Spring core and Maven\librarymanagement>

File Edit Selection View Go Run Terminal Help

Spring core and Maven

EXPLORER pom.xml BookRepository.java BookService.java LibraryManagementApplication.java applicationContext.xml LoggingAspect.java

```

librarymanagement > src > main > java > com > library > repository > BookRepository.java > ...
1 package com.library.repository;
2
3 public class BookRepository {
4     public void save() {
5         System.out.println("Saving book details in repository.");
6     }
7 }
8

```

OUTLINE TIMELINE VS CODE PETS JAVA PROJECTS MAVEN

MAVEN

- M LibraryManagement com.library...
- M aspect com.exampleaspect
- M demo com.exampledemo

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

[INFO] -----
[INFO] Total time: 1.296 s
[INFO] Finished at: 2025-07-04T15:49:16+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> []

In 8, Col 1 Spaces: 4 UTF-8 LF () Java Go Live Prettier

File Edit Selection View Go Run Terminal Help

Spring core and Maven

EXPLORER pom.xml BookRepository.java BookService.java LibraryManagementApplication.java applicationContext.xml LoggingAspect.java

```

librarymanagement > src > main > java > com > library > service > BookService.java > ...
1 package com.library.service;
2
3 import org.springframework.stereotype.Service;
4
5 @Service
6 public class BookService {
7
8     public void issueBook() {
9         System.out.println("Issuing book...");
10    }
11
12     public void returnBook() {
13         System.out.println("Returning book...");
14    }
15 }
16

```

OUTLINE TIMELINE VS CODE PETS JAVA PROJECTS MAVEN

MAVEN

- M LibraryManagement com.library...
- M aspect com.exampleaspect
- M demo com.exampledemo

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR

[INFO] -----
[INFO] Total time: 1.296 s
[INFO] Finished at: 2025-07-04T15:49:16+05:30
[INFO] -----
PS D:\Cognizant\Spring core and Maven\librarymanagement> []

In 16, Col 1 Spaces: 4 UTF-8 LF () Java Go Live Prettier

VS Code Screenshot showing the LibraryManagementApplication.java file:

```

librarymanagement > src > main > java > com > library > LibraryManagementApplication.java > ...
1 package com.library;
2
3 import com.library.service.BookService;
4 import org.springframework.context.ApplicationContext;
5 import org.springframework.context.support.ClassPathXmlApplicationContext;
6
7 public class LibraryManagementApplication {
8
9     Run | Debug
10    public static void main(String[] args) {
11        ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");
12
13        BookService bookService = context.getBean(BookService.class);
14        bookService.issueBook();
15        bookService.returnBook();
16    }
17

```

The code highlights the XML configuration file reference ("applicationContext.xml") in line 11.

VS Code Screenshot showing the applicationContext.xml file:

```

librarymanagement > src > main > resources > applicationContext.xml > ...
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3         xmlns:context="http://www.springframework.org/schema/context"
4         xmlns:aop="http://www.springframework.org/schema/aop"
5         xmlns: xsi="http://www.w3.org/2001/XMLSchema-instance"
6         xsi:schemaLocation="
7             http://www.springframework.org/schema/beans
8             https://www.springframework.org/schema/beans/spring-beans.xsd
9             http://www.springframework.org/schema/aop
10            https://www.springframework.org/schema/aop/spring-aop.xsd
11            http://www.springframework.org/schema/context
12            https://www.springframework.org/schema/context/spring-context.xsd"
13
14     <context:component-scan base-package="com.library" />
15     <aop:aspectj-autoproxy />
16
17 </beans>
18

```

The XML schema location and component scan configurations are highlighted in red.

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "SPRING CORE AND MAVEN". The "src/main/java/com/library/aspect" folder contains the "LoggingAspect.java" file, which is selected.
- Code Editor:** Displays the content of LoggingAspect.java:


```

1 package com.library.aspect;
2
3 import org.aspectj.lang.JoinPoint;
4 import org.aspectj.lang.annotation.After;
5 import org.aspectj.lang.annotation.Aspect;
6 import org.aspectj.lang.annotation.Before;
7 import org.springframework.stereotype.Component;
8
9 @Aspect
10 @Component
11 public class LoggingAspect {
12
13     @Before("execution(* com.library.service.*.*(..))")
14     public void logBefore(JoinPoint joinPoint) {
15         System.out.println("Before method: " + joinPoint.getSignature().getName());
16     }
17
18     @After("execution(* com.library.service.*.*(..))")
19     public void logAfter(JoinPoint joinPoint) {
20         System.out.println("After method: " + joinPoint.getSignature().getName());
21     }
22 }
```
- Terminal:** Shows the command line output of the Maven build and execution:


```

PS D:\Cognizont\Spring core and Maven\librarymanagement>
mvn exec:java
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Before method: issueBook
Issuing book...
After method: issueBook
Before method: returnBook
Returning book...
After method: returnBook
[INFO]
[INFO] -----
[INFO] Total time: 1.296 s
[INFO] Finished at: 2025-07-04T15:49:16+05:30
[INFO]
```
- Status Bar:** Shows the current file is "LoggingAspect.java", line 23, column 1, and other status indicators like "Spaces: 4", "UTF-8", "LF", "Java", "Go Live", and "Prettier".

```

PS D:\Cognizont\Spring core and Maven\librarymanagement>
mvn exec:java
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.library:LibraryManagement >-----
[INFO] Building LibraryManagement 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ LibraryManagement ---
Before method: issueBook
Issuing book...
After method: issueBook
Before method: returnBook
Returning book...
After method: returnBook
[INFO]
[INFO] -----
[INFO] Total time: 1.296 s
[INFO] Finished at: 2025-07-04T15:49:16+05:30
[INFO]
```

Exercise 9: Creating a Spring Boot Application

Scenario:

You need to create a Spring Boot application for the library management system to simplify configuration and deployment.

Steps:

1. **Create a Spring Boot Project:**
 - Use **Spring Initializr** to create a new Spring Boot project named **LibraryManagement**.
2. **Add Dependencies:**
 - Include dependencies for **Spring Web**, **Spring Data JPA**, and **H2 Database**.
3. **Create Application Properties:**
 - Configure database connection properties in **application.properties**.
4. **Define Entities and Repositories:**
 - Create **Book** entity and **BookRepository** interface.
5. **Create a REST Controller:**
 - Create a **BookController** class to handle CRUD operations.
6. **Run the Application:**
 - Run the Spring Boot application and test the REST endpoints.

The screenshot shows the VS Code interface with the following details:

- Project Structure:** The Explorer view shows a project named "librarymanagement" with packages like ".metadata", ".vscode", "demo", and "librarymanagement". Under "librarymanagement", there are "src" and "target" folders. "src" contains "main" (with "java", "controller", "entity", "repository", "service", and "test" subfolders), "resources" (with "application.properties" and "applicationContext.xml"), and "pom.xml". "target" contains "logging" and ".project".


```

      SPRING CORE AND MAVEN
      .metadata
      .vscode
      demo
      librarymanagement
        src
          main
            java\com\library
              controller
                BookController.java
              entity
                Book.java
              repository
                BookRepository.java
              service
                BookService.java
            test.java
          target
            pom.xml
        resources
          application.properties
          applicationContext.xml
        test.java
      target
        logging
        .project
      
```
- Code Editor:** The main editor shows `LibraryManagementApplication.java` with the following code:


```

      package com.library;
      import org.springframework.boot.SpringApplication;
      import org.springframework.boot.autoconfigure.SpringBootApplication;
      @SpringBootApplication
      public class LibraryManagementApplication {
        public static void main(String[] args) {
          SpringApplication.run(LibraryManagementApplication.class, args);
        }
      }
      
```
- Terminal:** The terminal tab shows the build process and success message:


```

      [INFO] from pom.xml
      [INFO] -----[ jar ]-----
      [INFO]
      [INFO] --- clean:3.3.2:clean (default-clean) @ LibraryManagement ---
      [INFO] Deleting D:\Cognizant\Spring core and Maven\librarymanagement\target
      [INFO]
      [INFO] BUILD SUCCESS
      [INFO] -----
      [INFO] Total time: 0.507 s
      [INFO] Finished at: 2025-07-04T16:46:51+05:30
      [INFO] -----
      
```
- Status Bar:** The bottom status bar indicates "Java Ready".

The screenshot shows the Eclipse IDE interface with the following details:

- File Explorer (Left):** Shows the project structure:
 - SPRING CORE AND MAVEN**: .metadata, .vscode, demo, librarymanagement (selected), src (containing main, java\com\library, controller, BookController.java, entity, Book.java, repository, BookRepository.java, service, BookService.java, LibraryManagementApplication.java, resources, application.properties, applicationContext.xml).
 - test\java, target, pom.xml (with a red error icon), logging, .project.
- Editor (Center):** Displays `applicationContext.xml` content:

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:aop="http://www.springframework.org/schema/aop"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="
           http://www.springframework.org/schema/beans
           https://www.springframework.org/schema/beans/spring-beans.xsd
           http://www.springframework.org/schema/aop
           https://www.springframework.org/schema/aop/spring-aop.xsd
           http://www.springframework.org/schema/context
           https://www.springframework.org/schema/context/spring-context.xsd">

    <context:component-scan base-package="com.library" />
    <aop:aspectj-autoproxy />
</beans>
```
- Terminal (Bottom):** Shows Maven build output:

```
[INFO] from pom.xml
[INFO] -----
[INFO] --- [ jar ] ---
[INFO] --- clean:3.3.2:clean (default-clean) @ LibraryManagement ---
[INFO] Deleting D:\Cognizant\Spring core and Maven\librarymanagement\target
[INFO] 
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.507 s
[INFO] Finished at: 2025-07-04T16:46:51+05:30
[INFO] -----
```

The screenshot shows the VS Code interface with the following details:

- Project Explorer:** Shows the project structure under "LIBRARYMANAGEMENT".
 - src
 - main
 - java\com\library
 - controller
 - J BookController.java
 - entity
 - J Book.java
 - repository
 - J BookRepository.java
 - service
 - J BookService.java
 - J LibraryManagementAp...
 - resources
 - application.properties
 - applicationContext.xml
 - test\java
 - pom.xml
 - target
 - logging
 - .project
 - Editor:** The "application.properties" file is open, showing Spring configuration for H2 database and Hibernate dialect.

```
spring.datasource.url=jdbc:h2:mem:testdb
spring.datasource.driver-class-name=org.h2.Driver
spring.datasource.username=sa
spring.datasource.password=
spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
spring.h2.console.enabled=true
spring.jpa.show-sql=true
spring.jpa.hibernate.ddl-auto=create
```
 - Terminal:** Maven clean and build logs are displayed.

```
[INFO] from pom.xml
[INFO] -----
[INFO] --- [ jar ] ---
[INFO] --- clean:3.3.2:clean (default-clean) @ LibraryManagement ---
[INFO] Deleting D:\Cognizant\Spring core and Maven\librarymanagement\target
[INFO] 
[INFO] BUILD SUCCESS
[INFO] 
[INFO] Total time: 0.507 s
[INFO] Finished at: 2025-07-04T16:46:51+05:30
[INFO] -----
```
 - Bottom Status Bar:** Shows "Java: Ready".

Spring core and Maven

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Spring core and Maven
EXPLORER J LibraryManagementApplication.java applicationContext.xml application.properties J BookController.java J Book.java J BookRepository.java ...
    > .metadata
    > .vscode
    > demo
    > librarymanagement
        > src
            > main
                > java.com.library
                    > controller
                        J BookController.java
                    > entity
                        J Book.java
                    > repository
                        J BookRepository.java
                    > service
                        J BookService.java
                        J LibraryManagementAp...
                    > resources
                        & application.properties
                            & applicationContext.xml
                        > test.java
                    > target
                        pom.xml
                    > logging
                    & .project
    > OUTLINE
    > TIMELINE
    ... VS CODE PETS
    > JAVA PROJECTS
    > MAVEN
        > M LibraryManagement com.library...
        > M aspect com.exampleaspect
        > M demo com.exampledemo
    PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR
[INFO] from pom.xml
[INFO] -----[ jar ]-----
[INFO] --- clean:3.3.2:clean (default-clean) @ LibraryManagement ---
[INFO] Deleting D:\Cognizant\Spring core and Maven\librarymanagement\target
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.507 s
[INFO] Finished at: 2025-07-04T16:46:51+05:30
[INFO]
Ln 26, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

```

Spring core and Maven

```

File Edit Selection View Go Run Terminal Help ← → ⌘ Spring core and Maven
EXPLORER J LibraryManagementApplication.java applicationContext.xml application.properties J BookController.java J Book.java J BookRepository.java ...
    > .metadata
    > .vscode
    > demo
    > librarymanagement
        > src
            > main
                > java.com.library
                    > controller
                        J BookController.java
                    > entity
                        J Book.java
                    > repository
                        J BookRepository.java
                    > service
                        J BookService.java
                        J LibraryManagementAp...
                    > resources
                        & application.properties
                            & applicationContext.xml
                        > test.java
                    > target
                        pom.xml
                    > logging
                    & .project
    > OUTLINE
    > TIMELINE
    ... VS CODE PETS
    > JAVA PROJECTS
    > MAVEN
        > M LibraryManagement com.library...
        > M aspect com.exampleaspect
        > M demo com.exampledemo
    PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER SQL HISTORY TASK MONITOR
[INFO] from pom.xml
[INFO] -----[ jar ]-----
[INFO] --- clean:3.3.2:clean (default-clean) @ LibraryManagement ---
[INFO] Deleting D:\Cognizant\Spring core and Maven\librarymanagement\target
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.507 s
[INFO] Finished at: 2025-07-04T16:46:51+05:30
[INFO]
Ln 30, Col 1 Spaces: 4 UTF-8 LF () Java ⚡ Go Live ⚡ Prettier ⚡

```

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "LIBRARYMANAGEMENT".
 - src: controller (BookController.java), entity (Book.java), repository (BookRepository.java), service (BookService.java), and test (BookTest.java).
 - resources: application.properties and applicationContext.xml.
- Terminal:** Displays the build log:

```
[INFO]  from pom.xml
[INFO]  [ jar ]
[INFO]
[INFO] ... clean:3.3.2:clean (default-clean) @ LibraryManagement ...
[INFO] Deleting D:\Cognizant\Spring core and Maven\librarymanagement\target
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 0.507 s
[INFO] Finished at: 2025-07-04T16:46:51+05:30
[INFO]
```
- Bottom Status Bar:** Shows "Ln 8, Col 1" and "Java Ready".

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "SPRING CORE AND MAVEN". The "librarymanagement" package contains "BookService.java", which is currently selected.
- Code Editor:** Displays the content of BookService.java. The code defines a BookService class with methods for getting all books, getting a book by ID, creating a book, and deleting a book. It uses annotations like @Service, @Autowired, and @Transactional.
- Bottom Bar:** Includes tabs for OUTLINE, PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, PORTS, SPELL CHECKER, SQL HISTORY, and TASK MONITOR. The TERMINAL tab is active, showing build logs for Maven.

```
librarymanagement > src > main > java \ com \ library > service > BookService.java > Language Support for Java(TM) by Red Hat > BookService > returnBook()
```

```
1 package com.library.service;
2
3 import com.library.entity.Book;
4 import com.library.repository.BookRepository;
5 import org.springframework.beans.factory.annotation.Autowired;
6 import org.springframework.stereotype.Service;
7
8 import java.util.List;
9
10 @Service
11 public class BookService {
12
13     @Autowired
14     private BookRepository bookRepository;
15
16     public List<Book> getAllBooks() {
17         return bookRepository.findAll();
18     }
19
20     public Book getBookById(Long id) {
21         return bookRepository.findById(id).orElse(null);
22     }
23
24     public Book createBook(Book book) {
25         return bookRepository.save(book);
26     }
27
28     public void deleteBook(Long id) {
```

```
[INFO] from pom.xml
[INFO] -----
[INFO] [ jar ]
[INFO] [INFO] --- clean:3.3.2:clean (default-clean) @ LibraryManagement ---
[INFO] [INFO] Deleting D:\Cognizont\Spring core and Maven\librarymanagement\target
[INFO] [INFO] BUILD SUCCESS
[INFO] [INFO] -----
[INFO] Total time: 0.507 s
[INFO] Finished at: 2025-07-04T16:46:51+05:30
[INFO] -----
```