Exercise 1: Configuring a Basic Spring Application:

```
package com.library.LibraryManagemnet.repository;

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

```
package com.library.LibraryManagemnet.Service;

import com.library.LibraryManagemnet.repository.BookRepository;

public class BookService {
      private BookRepository bookRepository;

    // Setter for Spring to inject
    public void setBookRepository(BookRepository bookRepository) {
        this.bookRepository = bookRepository;
    }

    public void addBook(String bookName) {
        System.out.println("Adding book: " + bookName);
        bookRepository.saveBook(bookName);
    }
}
```

applicationContext.xml:

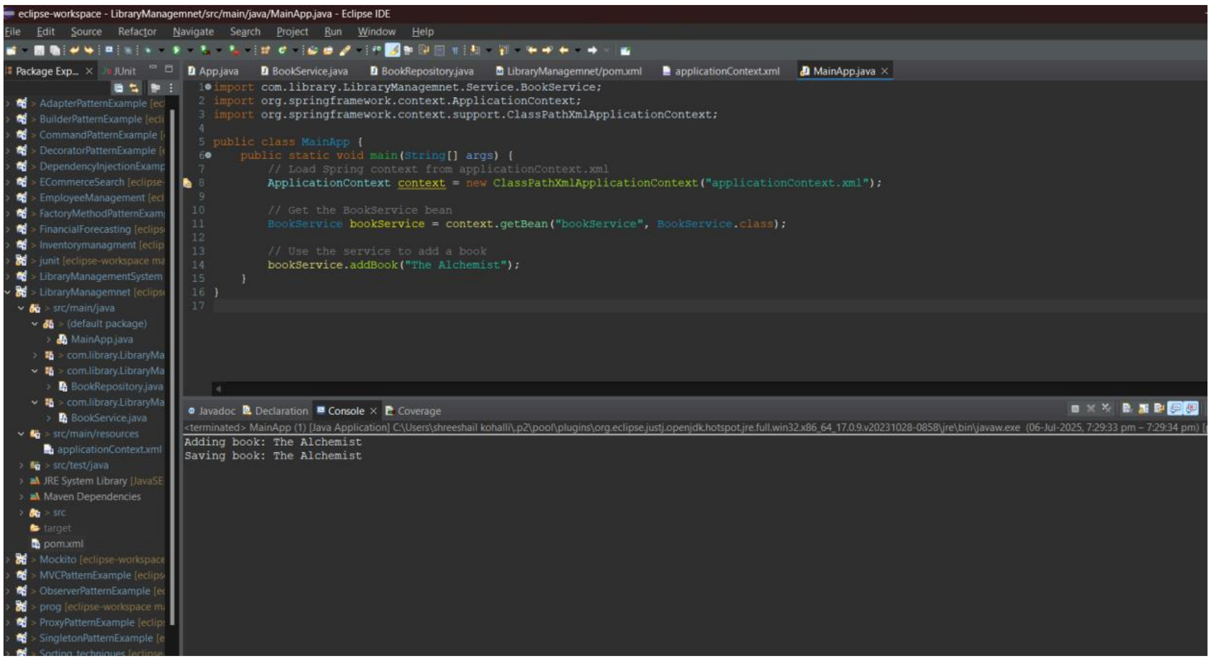
```
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="
           http://www.springframework.org/schema/beans
           http://www.springframework.org/schema/beans/spring-beans.xsd">

    <!-- Repository Bean -->
    <bean id="bookRepository"
class="com.library.LibraryManagemnet.repository.BookRepository"/>

    <!-- Service Bean with dependency injection -->
    <bean id="bookService"
class="com.library.LibraryManagemnet.Service.BookService">
        <property name="bookRepository" ref="bookRepository"/>
    </bean>
</beans>
```

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>LibraryManagemnet</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>
  <name>LibraryManagemnet</name>
  <url>http://maven.apache.org</url>

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

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>

        <!-- Spring Core -->
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.30</version>
        </dependency>

  </dependencies>
</project>
```

```
MainApp.java
import com.library.LibraryManagemnet.Service.BookService;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

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

        // Get the BookService bean
        BookService bookService = context.getBean("bookService",
BookService.class);

        // Use the service to add a book
        bookService.addBook("The Alchemist");
    }
}
```

Output:



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

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

        // Get the BookService bean
        BookService bookService = context.getBean("bookService", BookService.class);

        // Use the service to add a book
        bookService.addBook("The Alchemist");
    }
}
```

```
<terminated> MainApp (1) [Java Application] C:\Users\shreeshail kohalli\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.9.v20231028-0858\jre\bin\javaw.exe  (06-Jul-2025, 7:29:33 pm – 7:29:34 pm)
Adding book: The Alchemist
Saving book: The Alchemist
```

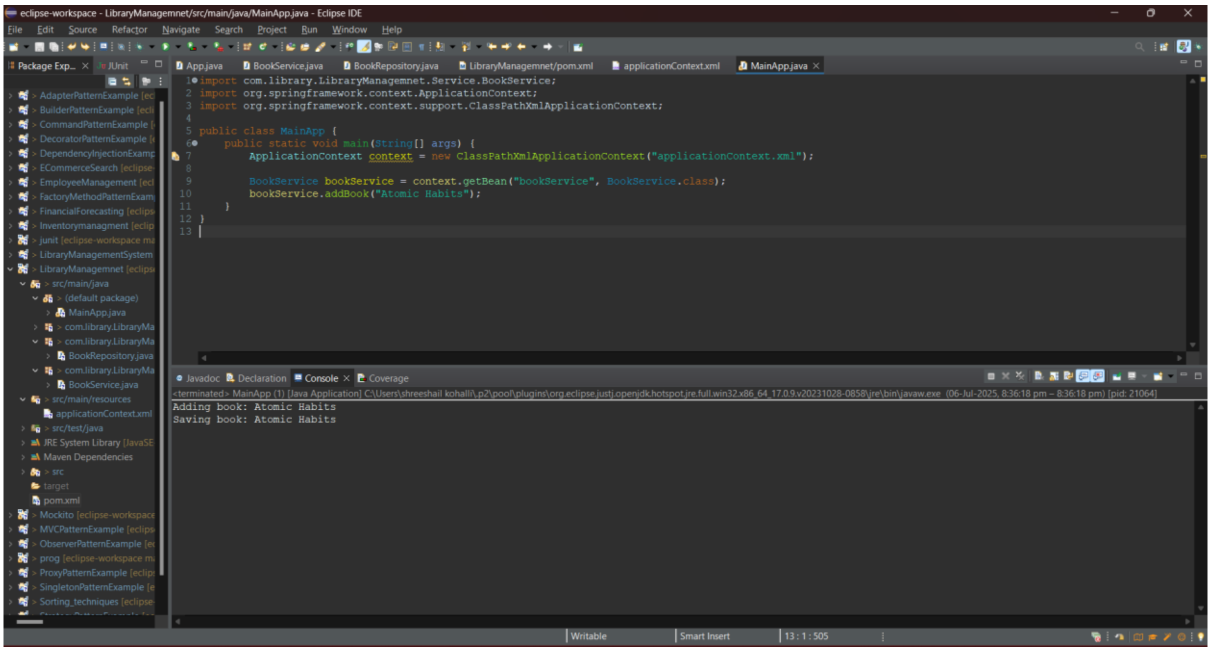
Exercise 2: Implementing Dependency Injection:

applicationContext.xml

```xml
<!-- src/main/resources/applicationContext.xml -->
<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.LibraryManagemnet.repository.BookRepository"/>

    <!-- Service Bean with DI (Setter Injection) -->
    <bean id="bookService"
class="com.library.LibraryManagemnet.Service.BookService">
        <property name="bookRepository" ref="bookRepository"/>
    </bean>
</beans>
```

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

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

        BookService bookService = context.getBean("bookService",
BookService.class);
        bookService.addBook("Atomic Habits");
    }
}
```

Output:



```
import com.library.LibraryManagemnet.Service.BookService;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

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

        BookService bookService = context.getBean("bookService", BookService.class);
        bookService.addBook("Atomic Habits");
    }
}
```

```
<terminated> MainApp (1) [Java Application] C:\Users\shreeshail kohalli\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.9.v20231028-0858\jre\bin\javaw.exe  (06-Jul-2025, 8:36:18 pm – 8:36:18 pm) [pid: 21064]
Adding book: Atomic Habits
Saving book: Atomic Habits
```

Exercise 4: Creating and Configuring a Maven Project:

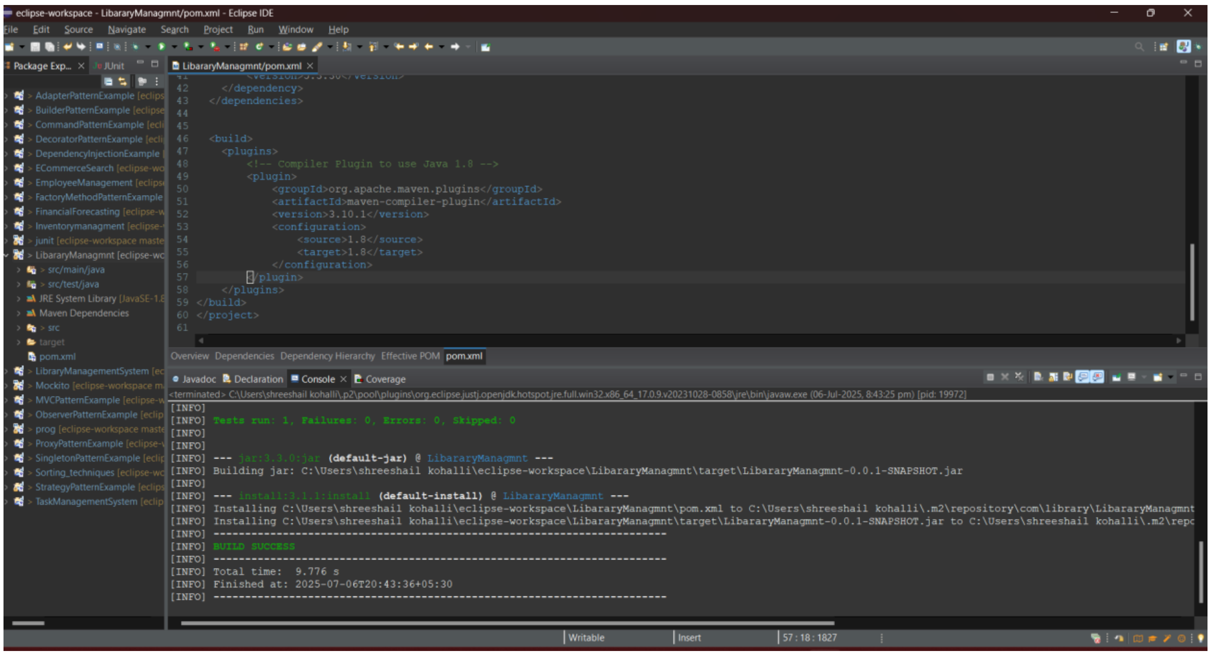
Pom.xml:

```
<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>LibararyManagmnt</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>LibararyManagmnt</name>
  <url>http://maven.apache.org</url>

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

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
    <dependency>
```

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

    <!-- Spring AOP (for aspect-oriented programming) -->
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-aop</artifactId>
        <version>5.3.30</version>
    </dependency>

    <!-- Spring Web MVC (if you plan to add web controllers later) -->
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-webmvc</artifactId>
        <version>5.3.30</version>
    </dependency>
  </dependencies>


  <build>
    <plugins>
        <!-- Compiler Plugin to use Java 1.8 -->
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.10.1</version>
            <configuration>
                <source>1.8</source>
                <target>1.8</target>
            </configuration>
        </plugin>
    </plugins>
</build>
</project>
```

Output:

Maven clean install



```xml
    </dependency>
  </dependencies>


  <build>
    <plugins>
        <!-- Compiler Plugin to use Java 1.8 -->
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.10.1</version>
            <configuration>
                <source>1.8</source>
                <target>1.8</target>
            </configuration>
        </plugin>
    </plugins>
</build>
</project>
```

```
<terminated> C:\Users\shreeshail kohalli\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.9.v20231028-0858\jre\bin\javaw.exe (06-Jul-2025, 8:43:25 pm) [pid: 19972]
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- jar:3.3.0:jar (default-jar) @ LibararyManagmnt ---
[INFO] Building jar: C:\Users\shreeshail kohalli\eclipse-workspace\LibararyManagmnt\target\LibararyManagmnt-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- install:3.1.1:install (default-install) @ LibararyManagmnt ---
[INFO] Installing C:\Users\shreeshail kohalli\eclipse-workspace\LibararyManagmnt\pom.xml to C:\Users\shreeshail kohalli\.m2\repository\com\library\LibararyManagmnt
[INFO] Installing C:\Users\shreeshail kohalli\eclipse-workspace\LibararyManagmnt\target\LibararyManagmnt-0.0.1-SNAPSHOT.jar to C:\Users\shreeshail kohalli\.m2\repo
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  9.776 s
[INFO] Finished at: 2025-07-06T20:43:36+05:30
[INFO] ------------------------------------------------------------------------
```