**Exercise 3: Implementing Logging with Spring AOP**

**Scenario:**

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

**BookServiceEx3.java**

package com.library.ex3;

import org.springframework.stereotype.Component;

@Component

public class BookServiceEx3 {

    public void addBook(String title) {

        System.out.println("ADDING BOOK: " + title);

        try {

            Thread.sleep(400); // simulate delay

        } catch (InterruptedException e) {

            e.printStackTrace();

        }

    }

    public String findBook(String title) {

        System.out.println("FINDING BOOK: " + title);

        try {

            Thread.sleep(250); // simulate delay

        } catch (InterruptedException e) {

            e.printStackTrace();

        }

        return title;

    }

}

**LibraryManagementApplication.java**

package com.library.ex3;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class LibraryManagementApplication {

    public static void main(String[] args) {

        ApplicationContext context =

                new ClassPathXmlApplicationContext("applicationContextEx3.xml");

        BookServiceEx3 bookService = context.getBean(BookServiceEx3.class);

        bookService.addBook("Clean Code");

        bookService.findBook("Effective Java");

    }

}

**LoggingAspect.java**

package com.library.ex3;

import org.aspectj.lang.ProceedingJoinPoint;

import org.aspectj.lang.annotation.Around;

import org.aspectj.lang.annotation.Aspect;

import org.springframework.stereotype.Component;

@Aspect

@Component

public class LoggingAspect {

    @Around("execution(\* com.library..\*(..))")

    public Object logExecutionTime(ProceedingJoinPoint joinPoint) throws Throwable {

        String methodName = joinPoint.getSignature().toShortString();

        System.out.println("STARTED EXECUTION OF: " + methodName);

        long start = System.currentTimeMillis();

        Object result = joinPoint.proceed();

        long end = System.currentTimeMillis();

        System.out.println("COMPELETED EXECUTION OF: " + methodName);

        System.out.println("EXECUTION TIME: " + (end - start) + " ms");

        return result;

    }

}

**applicationContextEx3.xml**

<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>

**pom.xml**

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

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

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

  <modelVersion>4.0.0</modelVersion>

  <groupId>com.spring</groupId>

  <artifactId>example1</artifactId>

  <version>1.0-SNAPSHOT</version>

  <name>example1</name>

  <!-- FIXME change it to the project's website -->

  <url>http://www.example.com</url>

  <properties>

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

    <maven.compiler.source>17</maven.compiler.source>

    <maven.compiler.target>17</maven.compiler.target>

  </properties>

  <dependencies>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-context</artifactId>

        <version>5.3.33</version>

    </dependency>

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-beans</artifactId>

        <version>5.3.33</version>

    </dependency>

    <dependency>

        <groupId>junit</groupId>

        <artifactId>junit</artifactId>

        <version>4.11</version>

        <scope>test</scope>

    </dependency>

    <!-- Spring AOP -->

    <dependency>

        <groupId>org.springframework</groupId>

        <artifactId>spring-aspects</artifactId>

        <version>5.3.30</version>

    </dependency>

    <!-- AspectJ Runtime -->

    <dependency>

        <groupId>org.aspectj</groupId>

        <artifactId>aspectjweaver</artifactId>

        <version>1.9.20</version>

    </dependency>

</dependencies>

  <build>

    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->

      <plugins>

        <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean\_Lifecycle -->

        <plugin>

          <artifactId>maven-clean-plugin</artifactId>

          <version>3.1.0</version>

        </plugin>

        <!-- default lifecycle, jar packaging: see https://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_jar\_packaging -->

        <plugin>

          <artifactId>maven-resources-plugin</artifactId>

          <version>3.0.2</version>

        </plugin>

        <plugin>

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

          <version>3.8.0</version>

        </plugin>

        <plugin>

          <artifactId>maven-surefire-plugin</artifactId>

          <version>2.22.1</version>

        </plugin>

        <plugin>

          <artifactId>maven-jar-plugin</artifactId>

          <version>3.0.2</version>

        </plugin>

        <plugin>

          <artifactId>maven-install-plugin</artifactId>

          <version>2.5.2</version>

        </plugin>

        <plugin>

          <artifactId>maven-deploy-plugin</artifactId>

          <version>2.8.2</version>

        </plugin>

        <!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site\_Lifecycle -->

        <plugin>

          <artifactId>maven-site-plugin</artifactId>

          <version>3.7.1</version>

        </plugin>

        <plugin>

          <artifactId>maven-project-info-reports-plugin</artifactId>

          <version>3.0.0</version>

        </plugin>

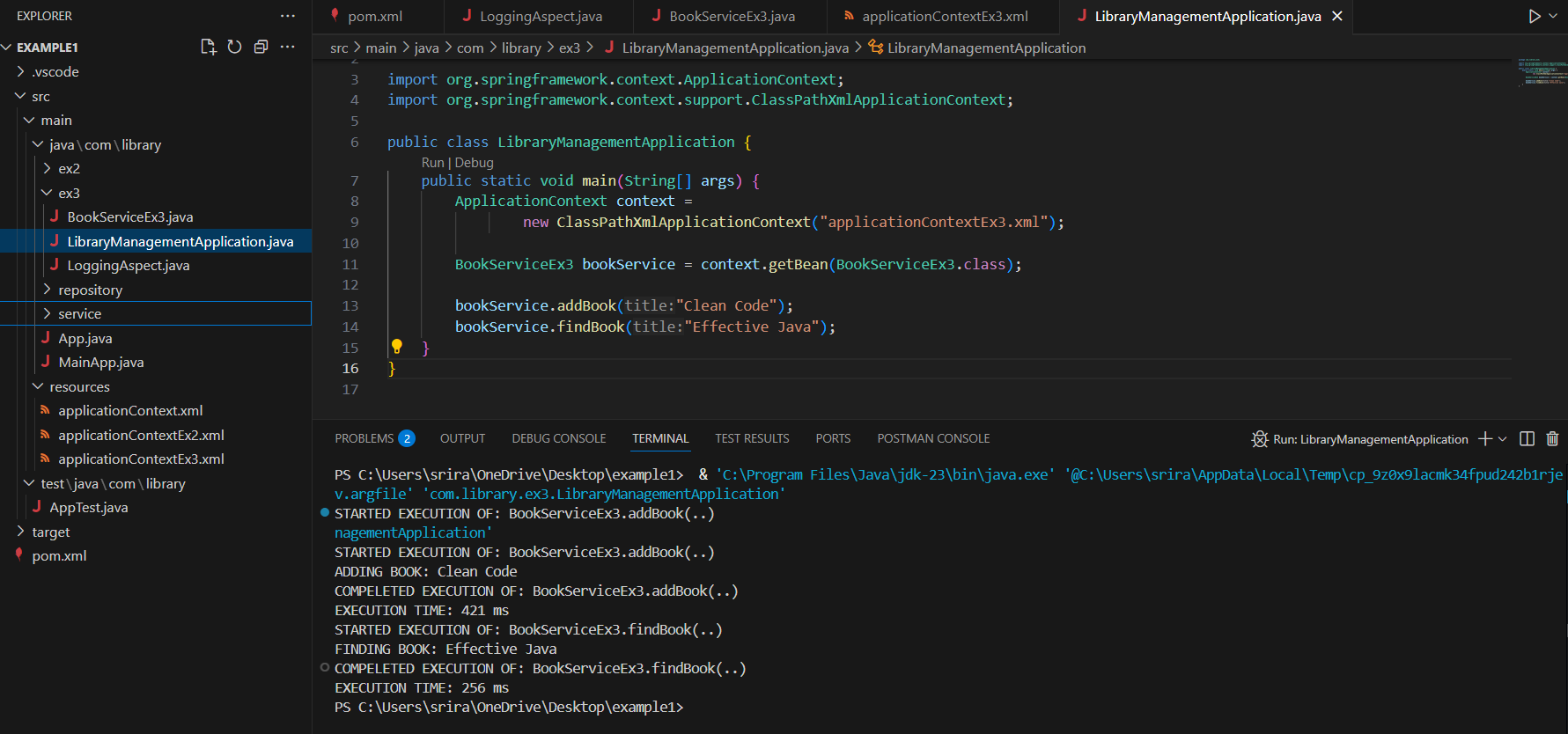
      </plugins>

    </pluginManagement>

  </build>

</project>

**Output:**

****