WEEK – 3

Exercise 7: Implementing Constructor and Setter Injection

Nipuna.A

Superset id:6432842

**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.

Solution:

1)Book.java

package com.library.model;

public class Book {

    private String title;

    public Book(String title) {

        this.title = title;

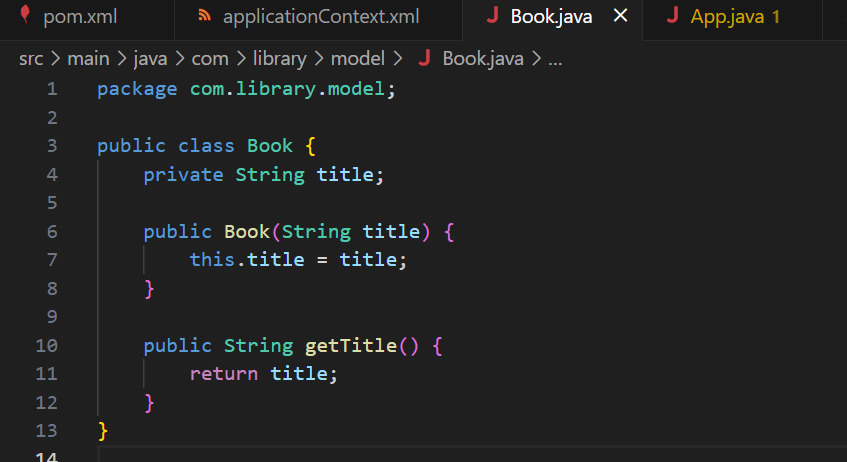
    }

    public String getTitle() {

        return title;

    }

}



2)BookRepository.java

package com.library.repository;

import com.library.model.Book;

public class BookRepository {

    public Book getBook() {

        return new Book("Spring in Action");

    }

}



3)BookService.java

package com.library.service;

import com.library.model.Book;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;

    public BookService(BookRepository bookRepository) {

        System.out.println("Constructor Injection used");

        this.bookRepository = bookRepository;

    }

    public void setBookRepository(BookRepository bookRepository) {

        System.out.println("Setter Injection used");

        this.bookRepository = bookRepository;

    }

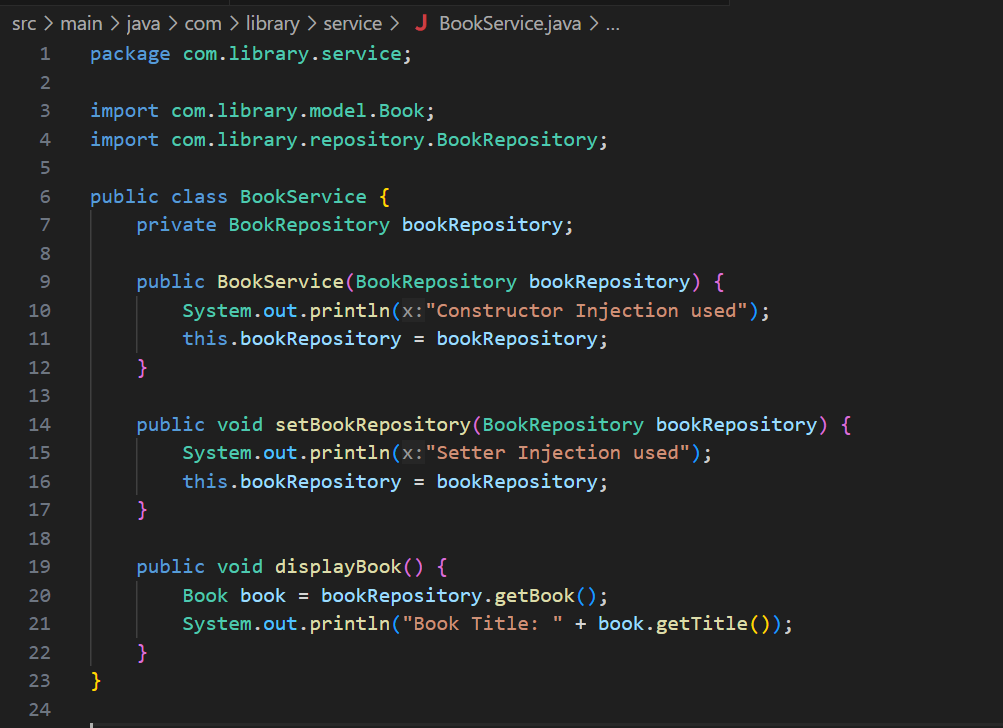
    public void displayBook() {

        Book book = bookRepository.getBook();

        System.out.println("Book Title: " + book.getTitle());

    }

}



4)App.java

package com.library;

import com.library.service.BookService;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class App {

    public static void main(String[] args) {

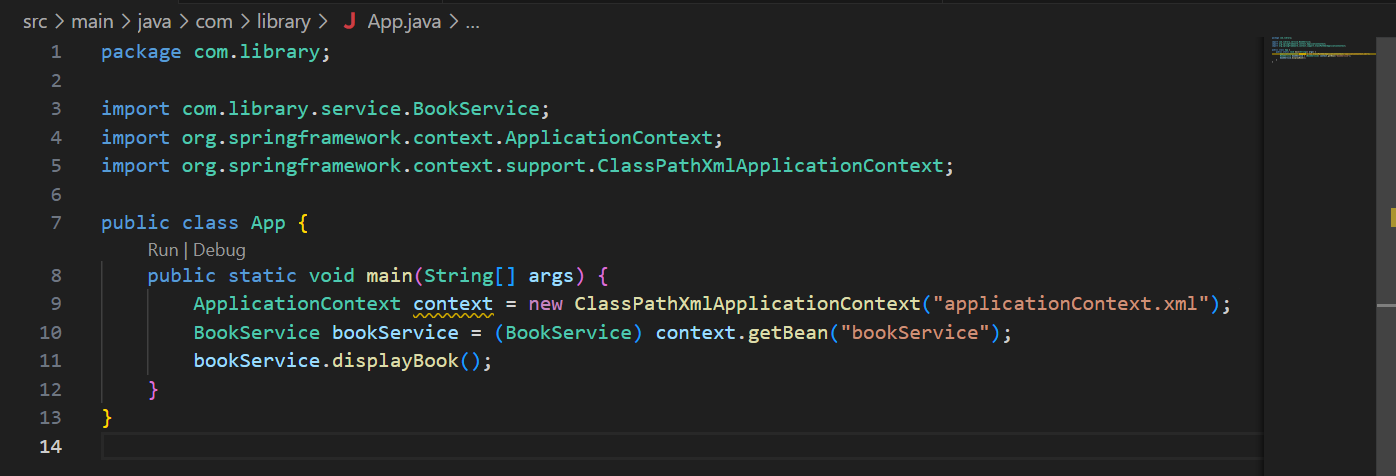
        ApplicationContext context = new ClassPathXmlApplicationContext("applicationContext.xml");

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

        bookService.displayBook();

    }

}



5)ApplicationContext.java

<?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">

    <!-- BookRepository bean -->

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

    <!-- BookService with both constructor and setter injection -->

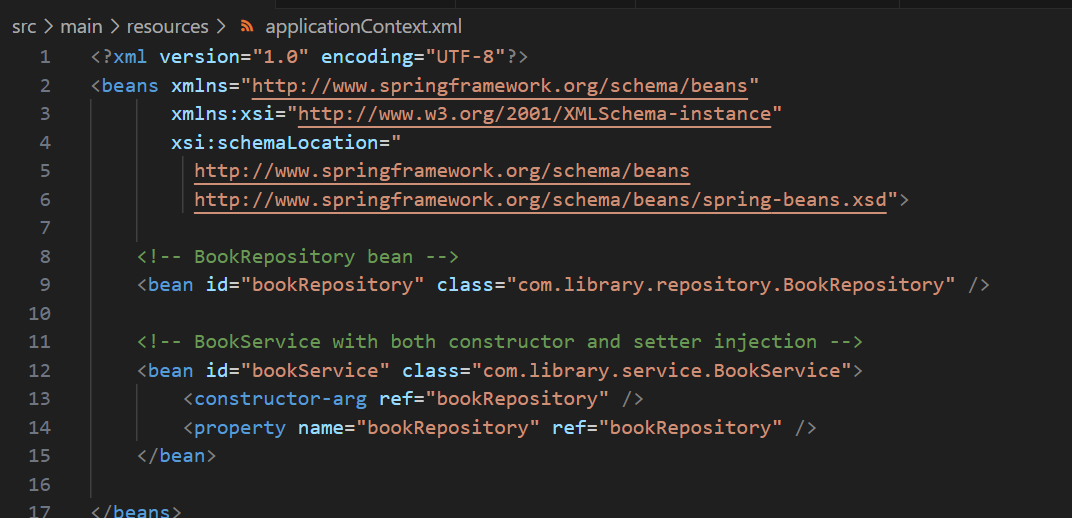
    <bean id="bookService" class="com.library.service.BookService">

        <constructor-arg ref="bookRepository" />

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

    </bean>

</beans>



6)Pom.xml

<project xmlns="http://maven.apache.org/POM/4.0.0"

         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

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

                             http://maven.apache.org/xsd/maven-4.0.0.xsd">

    <modelVersion>4.0.0</modelVersion>

    <groupId>com.library</groupId>

    <artifactId>LibraryManagement</artifactId>

    <version>1.0-SNAPSHOT</version>

    <properties>

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

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

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

    </properties>

    <dependencies>

        <!-- Spring Context Dependency -->

        <dependency>

            <groupId>org.springframework</groupId>

            <artifactId>spring-context</artifactId>

            <version>5.3.32</version>

        </dependency>

    </dependencies>

    <build>

        <plugins>

            <!-- Maven Compiler Plugin -->

            <plugin>

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

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

                <version>3.11.0</version>

                <configuration>

                    <source>1.8</source>

                    <target>1.8</target>

                </configuration>

            </plugin>

            <!-- Maven Exec Plugin to run the application -->

            <plugin>

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

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

                <version>3.1.0</version>

                <configuration>

                    <mainClass>com.library.App</mainClass>

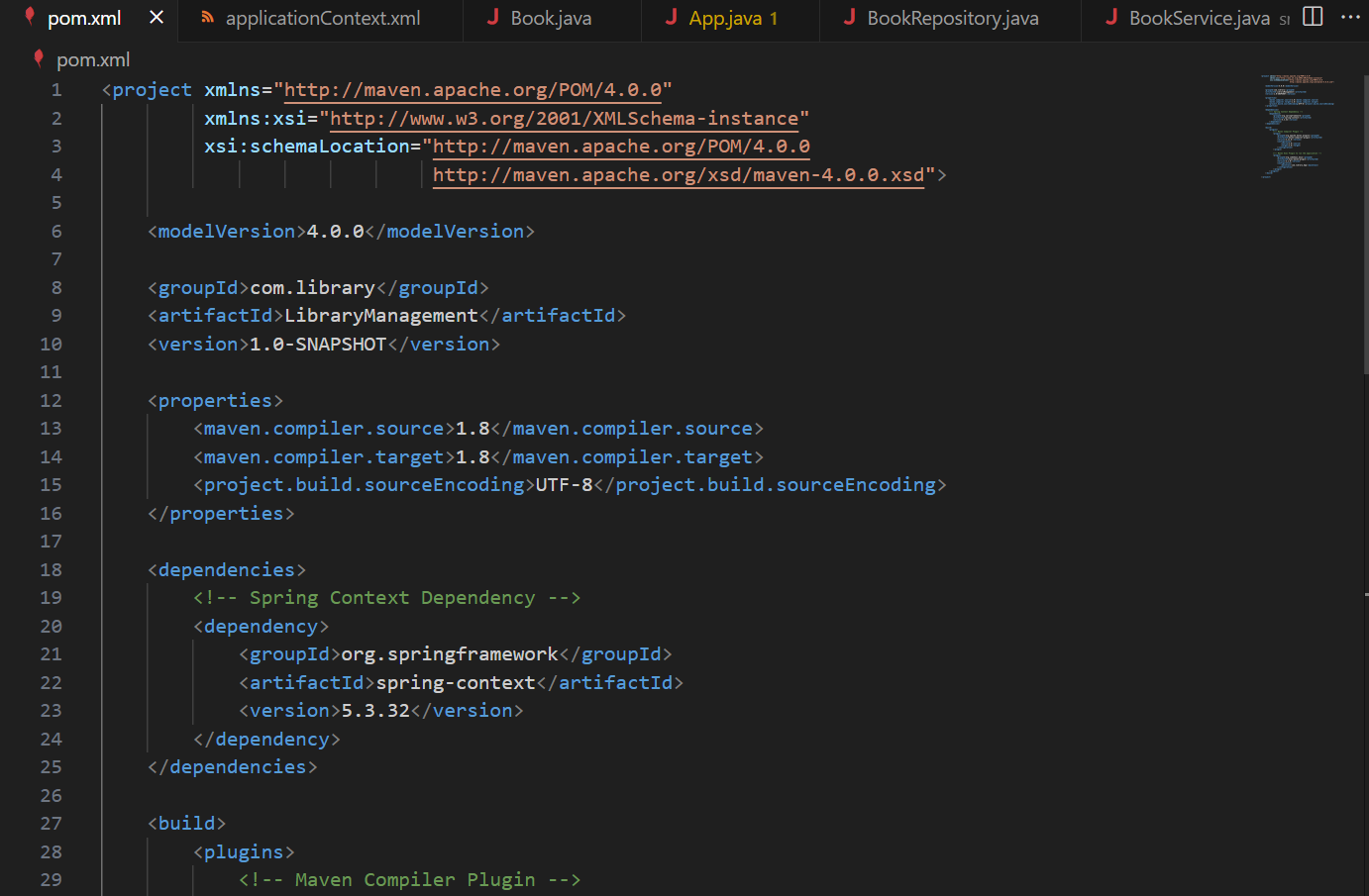
                </configuration>

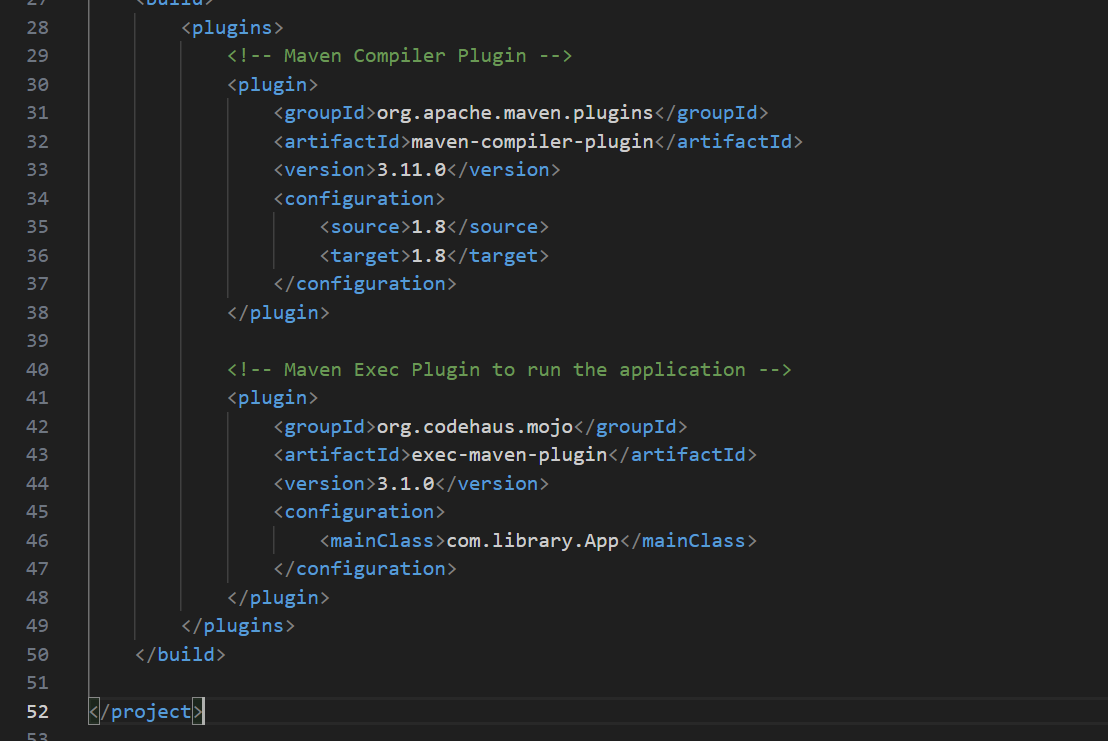
            </plugin>

        </plugins>

    </build>

</project>





Output:

