

Exercise 1: Configuring a Basic Spring Application

Main.java

```
package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;

    public void setBookRepository(BookRepository bookRepository) {

        this.bookRepository = bookRepository;

    }

    public void displayBook() {

        System.out.println("Inside BookService");

        bookRepository.fetchBook();

    }

}
```

BookService.java

```
package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;

    public void setBookRepository(BookRepository bookRepository) {

        this.bookRepository = bookRepository;

    }

    public void displayBook() {

        System.out.println("Inside BookService");

        bookRepository.fetchBook();

    }

}
```

```
}
```

BookRepository.java

```
package com.library.repository;

public class BookRepository {

    public void fetchBook() {

        System.out.println("Fetching book from repository...");

    }

}
```

applicationContext.xml

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

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

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

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

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

    </bean>

</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.maven</groupId>`

`<artifactId>LibManagement</artifactId>`

`<version>0.0.1-SNAPSHOT</version>`

`<name>LibManagement</name>`

`<url>http://www.example.com</url>`

`<properties>`

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

`<maven.compiler.release>17</maven.compiler.release>`

`</properties>`

`<dependencies>`

`<dependency>`

`<groupId>org.springframework</groupId>`

`<artifactId>spring-context</artifactId>`

`<version>5.3.29</version> <!-- You can change to latest stable if needed -->`

`</dependency>`

`<dependency>`

`<groupId>org.junit.jupiter</groupId>`

`<artifactId>junit-jupiter-api</artifactId>`

`<version>5.11.0</version>`

`<scope>test</scope>`

`</dependency>`

`<dependency>`

`<groupId>org.junit.jupiter</groupId>`

```

    <artifactId>junit-jupiter-params</artifactId>

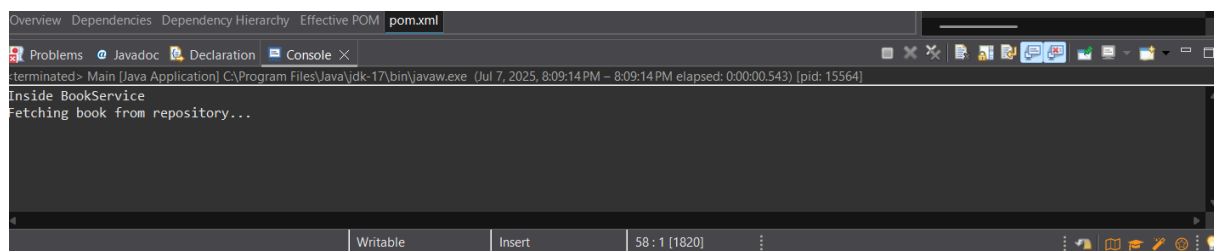
    <version>5.11.0</version>

    <scope>test</scope>
  </dependency>
</dependencies>

<build>
  <plugins>
    <plugin>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.13.0</version>
      <configuration>
        <release>17</release>
      </configuration>
    </plugin>
    <plugin>
      <artifactId>maven-surefire-plugin</artifactId>
      <version>3.3.0</version>
    </plugin>
  </plugins>
</build>
</project>

```

OUTPUT



Exercise 2: Implementing Dependency Injection

BookRepository.java

```
package com.library.repository;

import java.util.Arrays;
import java.util.List;

public class BookRepository {

    public List<String> getAllBooks() {

        return Arrays.asList("Java Fundamentals", "Spring in Action", "Effective Java");

    }

}
```

BookService.java

```
package com.library.service;

import com.library.repository.BookRepository;

public class BookService {

    private BookRepository bookRepository;

    public void setBookRepository(BookRepository bookRepository) {

        this.bookRepository = bookRepository;

    }

    public void listBooks() {

        bookRepository.getAllBooks().forEach(System.out::println);

    }

}
```

Main.java

```
package main;

import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
```

```

import com.library.service.BookService;

public class Main {

    public static void main(String[] args) {

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

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

        bookService.listBooks();

    }

}

```

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: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">

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

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

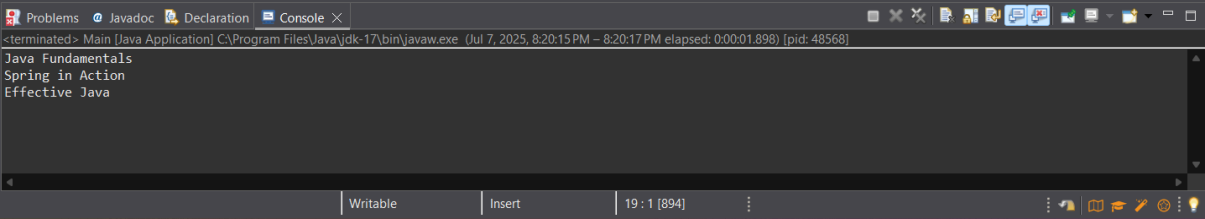
        <property name="bookRepository" ref="bookRepository" />

    </bean>

</beans>

```

OUTPUT



The screenshot shows an IDE console window with a dark theme. The title bar includes tabs for 'Problems', 'Javadoc', 'Declaration', and 'Console'. The console output displays the termination of a Java application, followed by a list of books: 'Java Fundamentals', 'Spring in Action', and 'Effective Java'. The status bar at the bottom indicates the cursor is at line 19, column 1, with a character count of 894. Various icons for file operations and settings are visible in the bottom right corner.

```
<terminated> Main [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (Jul 7, 2025, 8:20:15 PM - 8:20:17 PM elapsed: 0:00:01.898) [pid: 48568]  
Java Fundamentals  
Spring in Action  
Effective Java
```

Exercise 4: Creating and Configuring a Maven Project

Updated 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.maven</groupId>
    <artifactId>LibManagement</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>LibManagement</name>
    <properties>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>
    <dependencies>
        <!-- Spring Context -->
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.29</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-aop</artifactId>
            <version>5.3.29</version>
```



```
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-webmvc</artifactId>
  <version>5.3.29</version>
</dependency>
<dependency>
  <groupId>org.junit.jupiter</groupId>
  <artifactId>junit-jupiter-api</artifactId>
  <version>5.11.0</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.junit.jupiter</groupId>
  <artifactId>junit-jupiter-engine</artifactId>
  <version>5.11.0</version>
  <scope>test</scope>
</dependency>
</dependencies>
<build>
  <plugins>
    <plugin>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.10.1</version>
      <configuration>
        <source>1.8</source>
        <target>1.8</target>
      </configuration>
    </plugin>
  </plugins>
</build>
```

</plugin>

<plugin>

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

<version>3.3.0</version>

</plugin>

</plugins>

</build>

</project>

OUTPUT

