

Exercise 1: Configuring a Basic Spring Application

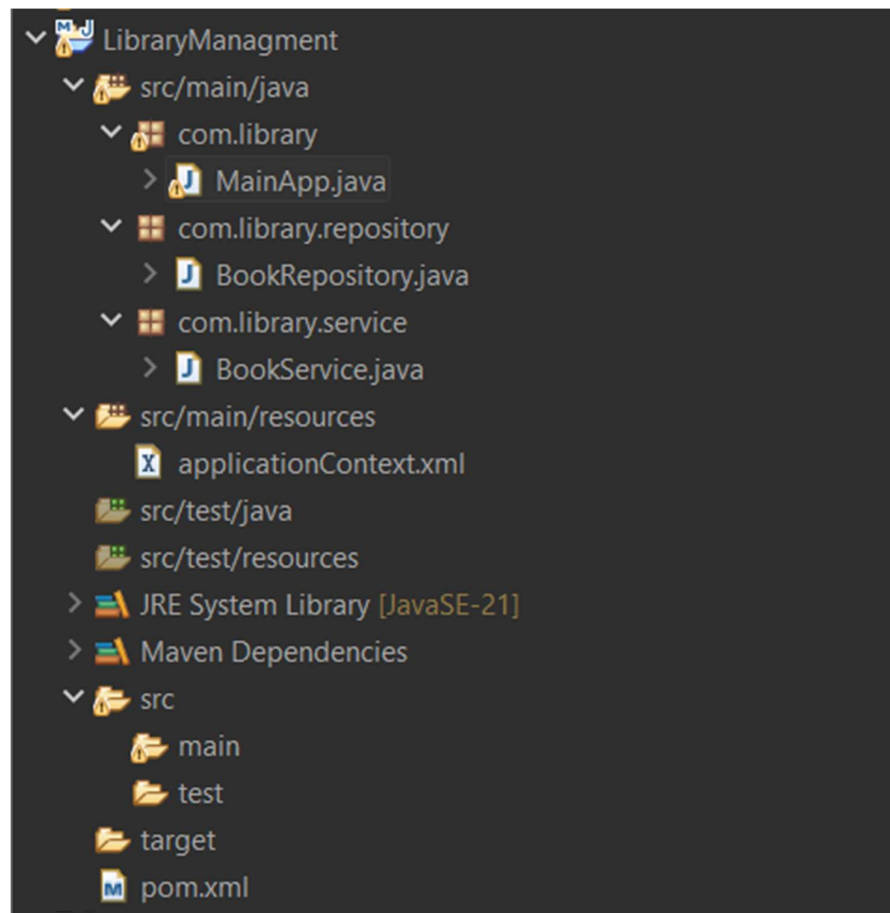
Scenario:

Your company is developing a web application for managing a library. You need to use the Spring Framework to handle the backend operations.

Steps and Implementation

1.Set Up a Spring Project

File Stucture :



- **Project Name:** LibraryManagement
- **Build Tool:** Maven

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

    <modelVersion>4.0.0</modelVersion>
    <groupId>com.library</groupId>
    <artifactId>LibraryManagement</artifactId>
    <version>0.0.1-SNAPSHOT</version>

    <properties>
        <maven.compiler.release>21</maven.compiler.release>
    </properties>

    <dependencies>
        <!-- Spring Context (which includes spring-core and spring-
beans) -->
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>5.3.31</version>
        </dependency>
    </dependencies>

    <build>
        <plugins>
            <!-- Compiler plugin for JDK 21 -->
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-compiler-plugin</artifactId>
                <version>3.11.0</version>
                <configuration>
                    <release>21</release>
                </configuration>
            </plugin>
        </plugins>
    </build>
</project>
```

```
        </plugin>
    </plugins>
</build>

</project>
```

2. Configure the Application Context

- **File:** applicationContext.xml
- **Location:** src/main/resources

```
<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>
```

3. Define Service and Repository Classes

File: BookRepository.java

```
package com.library.repository;

public class BookRepository {
    public String getBookTitle() {
        return "Varkala Diaries by Chaitanya Varma";
    }
}
```

File: 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 displayBookTitle() {
        System.out.println("Book Title: " +
            bookRepository.getBookTitle());
    }
}
```

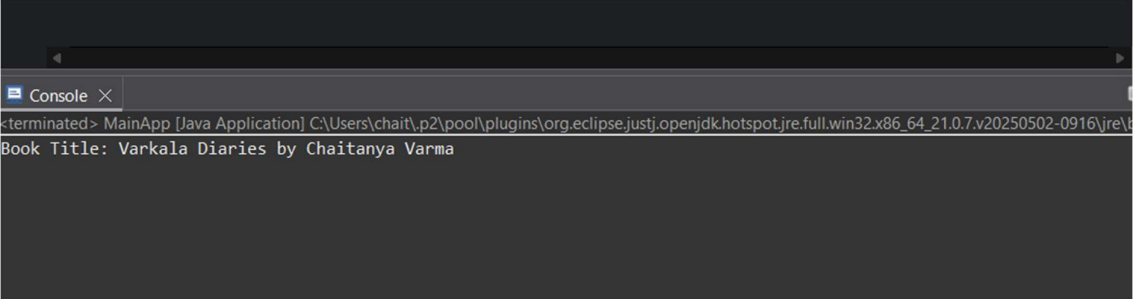
4. Run the Application**File: MainApp.java**

```
package com.library;

import com.library.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.displayBookTitle();
    }
}
```

OUTPUT :



```
<terminated> MainApp [Java Application] C:\Users\chait\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.7.v20250502-0916\jre\
Book Title: Varkala Diaries by Chaitanya Varma
```