**Exercise 1: Configuring a Basic Spring Application**

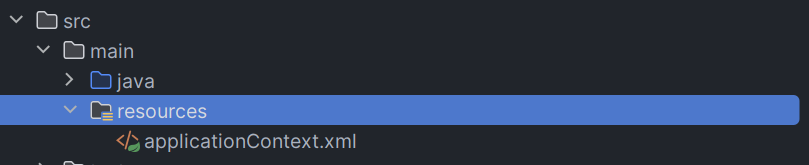
* Add Spring Core dependencies in the **pom.xml** file.

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.springcore</groupId>  
 <artifactId>LibraryManagement</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 <packaging>jar</packaging>  
  
 <name>LibraryManagement</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>  
 *<!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->* <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-core</artifactId>  
 <version>6.2.8</version>  
 </dependency>

*<!-- https://mvnrepository.com/artifact/org.springframework/spring-context -->*<dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-context</artifactId>  
 <version>6.2.8</version>  
</dependency>  
 </dependencies>  
</project>

* Create an XML configuration file named **applicationContext.xml** in the **src/main/resources** directory.



* *Define beans for* ***BookService*** *and* ***BookRepository*** *in the XML file.*

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"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xsi:schemaLocation="http://www.springframework.org/schema/beans  
 https://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 https://www.springframework.org/schema/context/spring-context.xsd">  
  
 <context:annotation-config/>  
  
 <bean class="com.library.repository.BookRepository" name="bookRepository"/>  
 <bean class="com.library.service.BookService" name="bookService"/>  
  
</beans>

* Create a package **com.library.service** and add a class **BookService**.

CODE:

package com.library.service;

import com.library.repository.BookRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.beans.factory.annotation.Qualifier;

public class BookService {

@Autowired

@Qualifier("bookRepository")

private BookRepository repo;

public void addBookByName(String name){

repo.addBook(name);

}

public void deleteBookByName(String name){

repo.deleteBook(name);

}

}

* Create a package **com.library.repository** and add a class **BookRepository**.

CODE:

package com.library.repository;

import org.springframework.stereotype.Repository;

@Repository

public class BookRepository {

public void addBook(String name){

System.out.println(name+ " Added Successfully");

}

public void deleteBook(String name){

System.out.println(name + " Deleted Successfully");

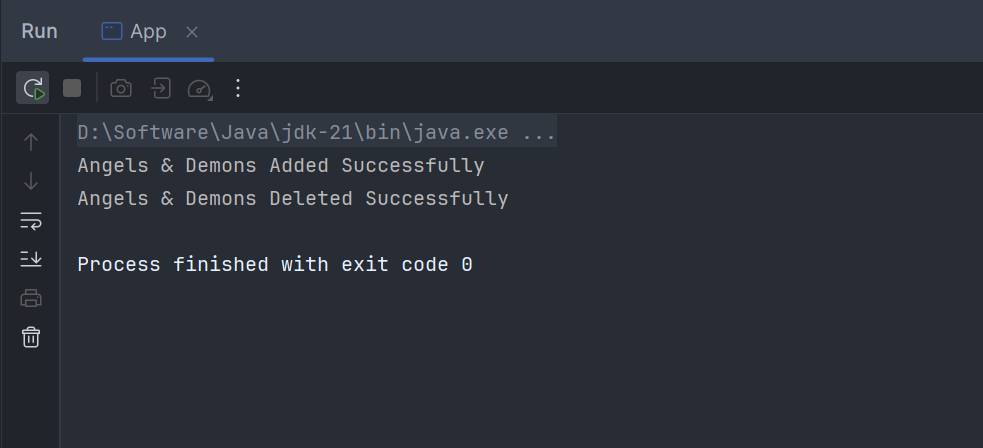
}

}

MAIN CLASS:

*package* com.springcore;  
  
  
*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 service = context.getBean("bookService",BookService.*class*);  
 service.addBookByName("Angels & Demons");  
 service.deleteBookByName("Angels & Demons");  
  
 }  
}

**OUTPUT:**



**Exercise 2: Implementing Dependency Injection**

Using the same example from last exercise, instead of using @Autowired, we wire the object of **BookRepository** into **BookService**

**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"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xsi:schemaLocation="http://www.springframework.org/schema/beans  
 https://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 https://www.springframework.org/schema/context/spring-context.xsd">  
  
 <context:annotation-config/>  
   
 <bean class="com.library.repository.BookRepository" name="bookRepository"/>  
 *<!--Wiring object using ref-->* <bean class="com.library.service.BookService" name="bookService">  
 <property name="repo" ref="bookRepository"/>  
 </bean>  
</beans>

**BookRepository:**

*package* com.library.repository;  
  
*import* org.springframework.stereotype.*Repository*;  
  
*@Repository  
public class* BookRepository {  
 *public void* addBook(String name){  
 System.out.println(name+ " Added Successfully");  
 }  
 *public void* deleteBook(String name){  
 System.out.println(name + " Deleted Successfully");  
 }  
}

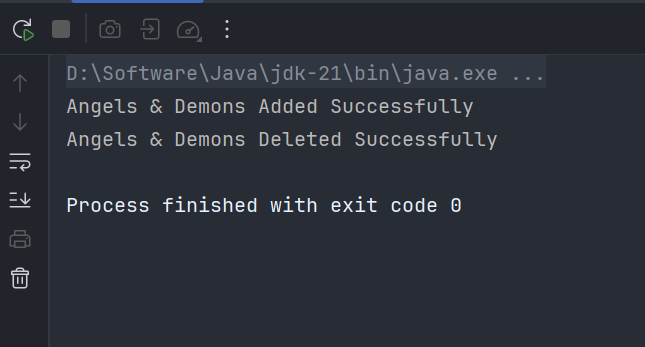
**BookService:**

*package* com.library.service;  
  
*import* com.library.repository.BookRepository;  
*import* org.springframework.beans.factory.annotation.*Autowired*;  
*import* org.springframework.beans.factory.annotation.*Qualifier*;  
  
*public class* BookService {  
  
 *private* BookRepository repo;  
  
 *//setter  
 public void* setRepo(BookRepository repo) {  
 *this*.repo = repo;  
 }  
  
 *public void* addBookByName(String name){  
 repo.addBook(name);  
 }  
  
 *public void* deleteBookByName(String name){  
 repo.deleteBook(name);  
 }  
  
}

**LibraryManagementApplication:**

*package* com.springcore;  
  
  
*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 service = context.getBean("bookService",BookService.*class*);  
 service.addBookByName("Angels & Demons");  
 service.deleteBookByName("Angels & Demons");  
  
 }  
}

**OUTPUT:**

****

**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.springcore</groupId>  
 <artifactId>LibraryManagement</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 <packaging>jar</packaging>  
  
 <name>LibraryManagement</name>  
 <url>http://maven.apache.org</url>  
  
 <properties>  
 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 </properties>  
  
 <build>  
 <plugins>  
 <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>  
 </plugins>  
 </build>  
  
 <dependencies>  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>3.8.1</version>  
 <scope>test</scope>  
 </dependency>  
 *<!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->* <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-core</artifactId>  
 <version>6.2.8</version>  
 </dependency>  
 *<!-- https://mvnrepository.com/artifact/org.springframework/spring-context -->* <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-context</artifactId>  
 <version>6.2.8</version>  
 </dependency>  
 *<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->* <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-webmvc</artifactId>  
 <version>6.2.8</version>  
 </dependency>  
 *<!-- https://mvnrepository.com/artifact/org.springframework/spring-aop -->* <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-aop</artifactId>  
 <version>6.2.8</version>  
 </dependency>  
  
  
 </dependencies>  
</project>

**The above code has dependencies for Spring Context, Spring AOP, and Spring WebMVC. The Maven Compiler Plugin is set to Java version 1.8.**