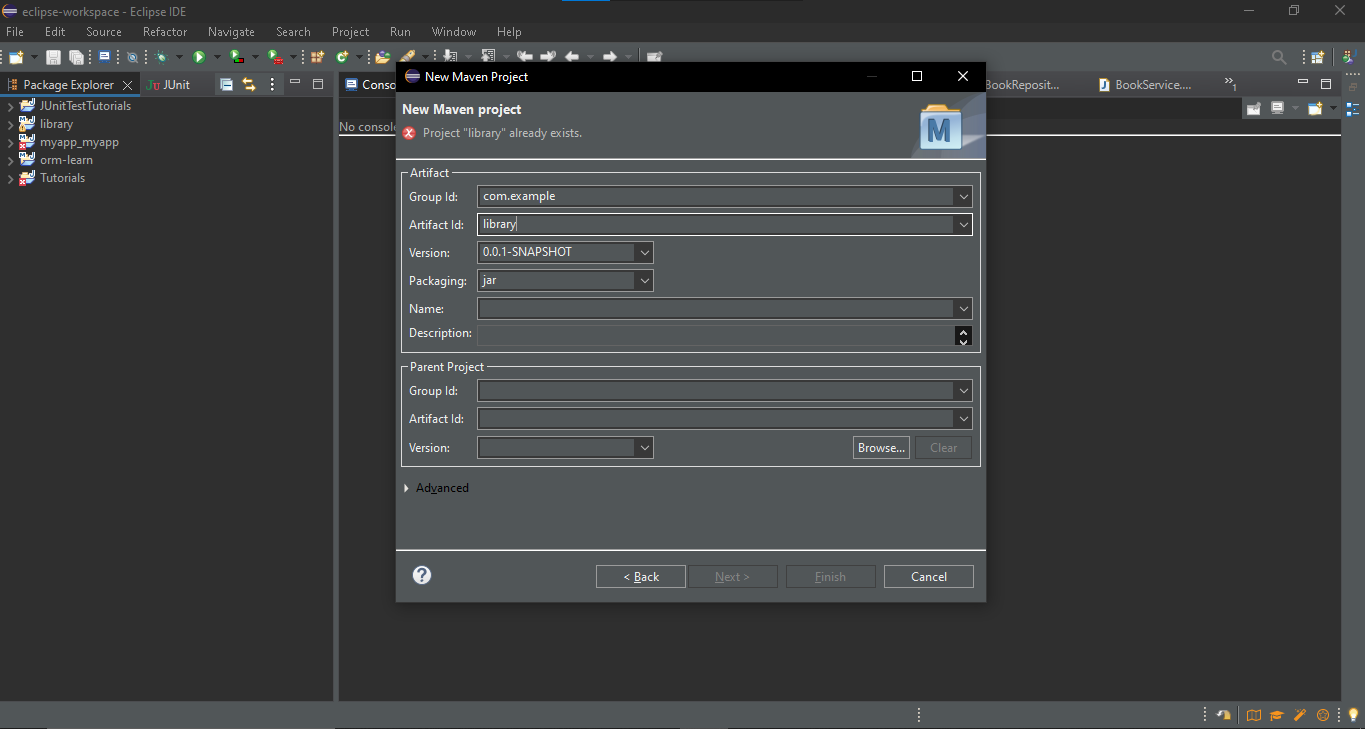
**Exercise 1: Configuring a Basic Spring Application**

1. **Setting up project:**

****

1. **Adding Dependencies:**

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>6.2.8</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.33</version> <!-- Or any stable latest version -->

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-beans</artifactId>

<version>5.3.33</version>

</dependency>

1. **Application Context Configuration:**

<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

[https://www.springframework.org/beans/spring-beans.xsd](http://www.springframework.org/schema/beans/spring-beans.xsd)">

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

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

</bean>

</beans>

1. **Repository Class:**

package com.example.library.repository;

public class BookRepository {

public void display() {

System.out.println("Hello world");

}

}

**Service Class:**

package com.example.library.service;

public class BookService {

public void display() {

System.out.println("Book Has been Added");

}

}

**MainApp Class:**

package com.example.library;

import com.example.library.repository.BookRepository;

import com.example.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");

BookRepository obj = (BookRepository) context.getBean("bookRepository");

obj.display();

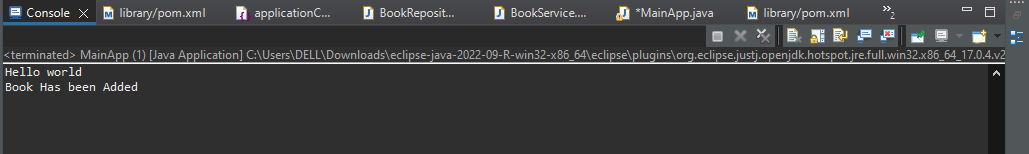
BookService obj1 = (BookService) context.getBean("bookService");

obj1.display();

}

}

**OUTPUT:**

****

**Exercise 2: Implementing Dependency Injection**

**1.XML Configuration:**

<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="bookrepo" class="com.example.library.repository.BookRepository" />

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

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

</bean>

</beans>

**2.BookService Class:**

package com.example.library.service;

import com.example.library.repository.BookRepository;

public class BookService {

private BookRepository bookrepo;

public void setBookrepo(BookRepository bookrepo) {

this.bookrepo = bookrepo;

}

public void display() {

System.out.println(bookrepo.addBook("Lif of Pi"));

System.out.println("Book Has been Added");

}

}

**3.Main App:**

package com.example.library;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.example.library.repository.BookRepository;

import com.example.library.service.BookService;

public class MainApp {

public static void main(String[] args) {

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

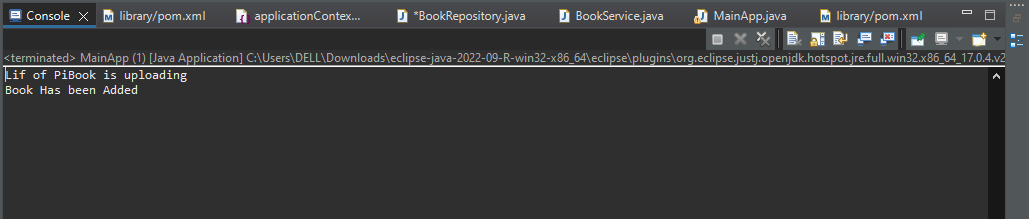
BookService obj1 = (BookService) context.getBean("bookService");

obj1.display();

}

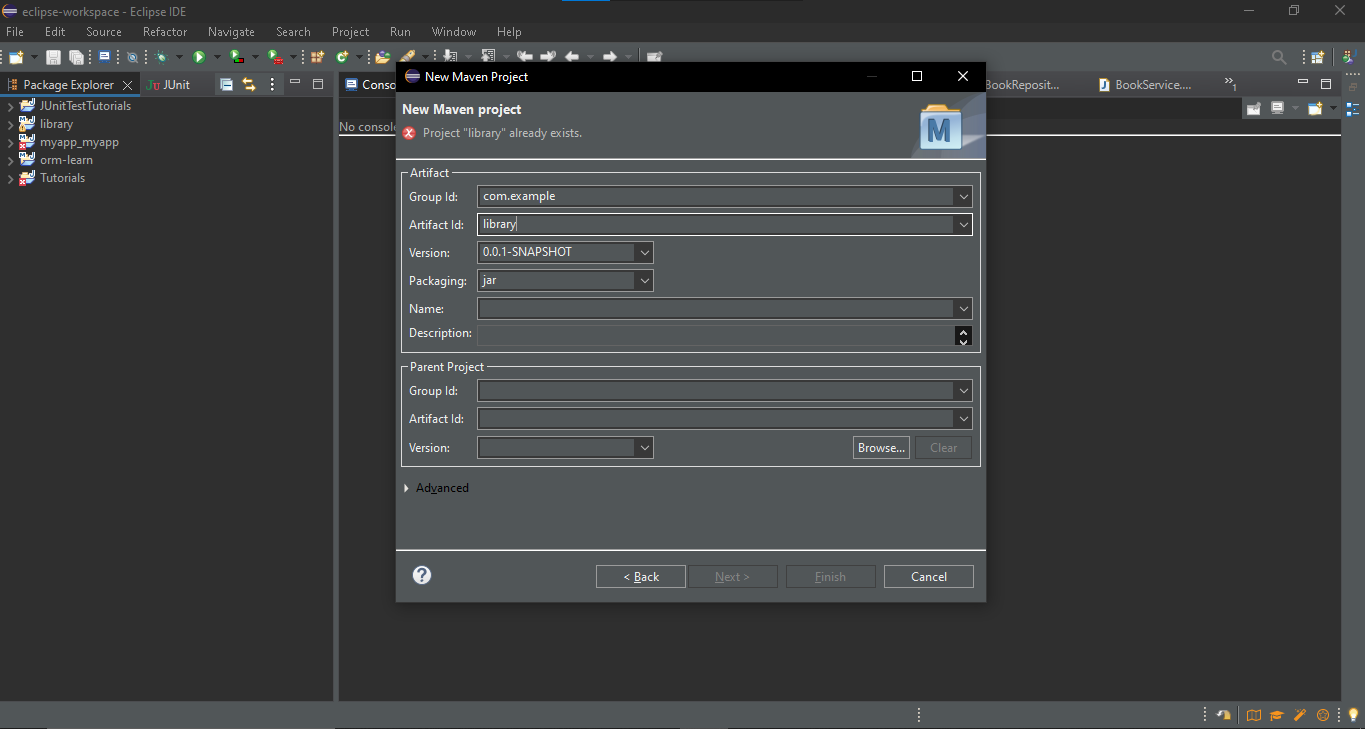
}

**OUTPUT:**

****

**Exercise 4: Creating and Configuring a Maven Project**

**1.Maven Project:(Same in the 1st exercise)**

****

**2.Adding Spring Dependencies :**

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.33</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

<version>5.3.33</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.3.33</version>

</dependency>

</dependencies>

**3.Configure Maven Plugins:**

<build>

<plugins>

<plugin>

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

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

<version>3.8.1</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

</plugins>

</build>