**EXERCISE 1**

**1.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.example</groupId>

<artifactId>spring-core-maven</artifactId>

<version>1.0-SNAPSHOT</version>

<dependencies>

<!-- Spring Context for Core -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.34</version>

</dependency>

</dependencies>

</project>

**2.src/main/java/com/example/MessageService.java**

package com.example;

public class MessageService {

private String message;

// Setter injection

public void setMessage(String message) {

this.message = message;

}

public void displayMessage() {

System.out.println("Message from Spring: " + message);

}

}

**3.src/main/resources/beans.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="messageService" class="com.example.MessageService">

<property name="message" value="Hello, Spring Core with Maven!"/>

</bean>

</beans>

**4.src/main/java/com/example/App.java**

package com.example;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class App {

public static void main(String[] args) {

// Load Spring container with XML configuration

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

// Retrieve the bean from the container

MessageService service = (MessageService) context.getBean("messageService");

// Call the method to display the message

service.displayMessage();

}

}

**Build the project**

mvn clean package

**Run the program**

java -cp target/spring-core-maven-1.0-SNAPSHOT.jar com.example.App

**Output**

Message from Spring: Hello, Spring Core with Maven!

**EXERCISE 2**

**1.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.example</groupId>

<artifactId>spring-core-maven</artifactId>

<version>1.0-SNAPSHOT</version>

<dependencies>

<!-- Spring Core -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.34</version>

</dependency>

</dependencies>

</project>

**2.src/main/java/com/example/GreetingService.java**

package com.example;

public interface GreetingService {

void greet();

}

## 3.src/main/java/com/example/GreetingServiceImpl.java

## package com.example;

## public class GreetingServiceImpl implements GreetingService {

## @Override

## public void greet() {

## System.out.println("Hello from GreetingServiceImpl!");

## }

## }

## 4. src/main/java/com/example/Person.java

## package com.example;

## public class Person {

## private GreetingService greetingService;

## // Setter Injection

## public void setGreetingService(GreetingService greetingService) {

## this.greetingService = greetingService;

## }

## public void showGreeting() {

## greetingService.greet();

## }

## }

## 5. src/main/resources/beans.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 for GreetingService implementation -->

## <bean id="greetingService" class="com.example.GreetingServiceImpl" />

## <!-- Bean for Person with dependency injected -->

## <bean id="person" class="com.example.Person">

## <property name="greetingService" ref="greetingService" />

## </bean>

## </beans>

## 6. src/main/java/com/example/App.java

## package com.example;

## import org.springframework.context.ApplicationContext;

## import org.springframework.context.support.ClassPathXmlApplicationContext;

## public class App {

## public static void main(String[] args) {

## ApplicationContext context = new ClassPathXmlApplicationContext("beans.xml");

## Person person = (Person) context.getBean("person");

## person.showGreeting();

## }

## }

## Build

## mvn clean package

## Run

## java -cp target/spring-core-maven-1.0-SNAPSHOT.jar com.example.App

## Output

## Hello from GreetingServiceImpl!

## EXERCISE 3

## 1.Project Structure

## spring-core-maven-ex4/

## ├── src/

## │ ├── main/

## │ ├── java/

## │ │ └── com/example/App.java

## │ └── resources/

## └── pom.xml

## 2.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.example</groupId>

## <artifactId>spring-core-maven-ex4</artifactId>

## <version>1.0-SNAPSHOT</version>

## <dependencies>

## <!-- Adding Spring Core dependency for future exercises -->

## <dependency>

## <groupId>org.springframework</groupId>

## <artifactId>spring-context</artifactId>

## <version>5.3.34</version>

## </dependency>

## </dependencies>

## <build>

## <plugins>

## <!-- Compiler plugin to specify Java version -->

## <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>

## </project>

## 3. src/main/java/com/example/App.java

## package com.example;

## public class App {

## public static void main(String[] args) {

## System.out.println("Maven Project Created Successfully!");

## }

## }

## Create Maven Project

## mvn archetype:generate -DgroupId=com.example -DartifactId=spring-core-maven-ex4 -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false

## Build

## mvn clean package

## Run

## java -cp target/spring-core-maven-ex4-1.0-SNAPSHOT.jar com.example.App

## Output

## Maven Project Created Successfully!