**SPRING CORE\_MAVEN**

**Exercise 2: Implementing Dependency Injection**

**Description:**

This class will be the dependency. It provides a message that another class will use.

**Code:**

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

package com.example;

public class MessageService {

public String getMessage() {

return "Hello from MessageService!";

}

}

**Description:**

This is the target bean that depends on MessageService. Spring will inject the dependency via a setter method.

**Code:**

// src/main/java/com/example/UserService.java

package com.example;

public class UserService {

private MessageService messageService;

// Setter method for Spring to inject the dependency

public void setMessageService(MessageService messageService) {

this.messageService = messageService;

}

public void processMessage() {

System.out.println(messageService.getMessage());

}

}

**Description:**

Tell Spring how to create the beans and inject one into the other.

**Code:**

<!-- src/main/resources/applicationContext.xml -->

<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 MessageService -->

<bean id="messageService" class="com.example.MessageService"/>

<!-- Bean for UserService with dependency injected -->

<bean id="userService" class="com.example.UserService">

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

</bean>

</beans>

**Description:**

Write a main() method to load the Spring container and test if the dependency was injected correctly.

**Code:**

// 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 the Spring configuration file

ApplicationContext context =

new ClassPathXmlApplicationContext("applicationContext.xml");

// Get the bean from Spring

UserService userService = context.getBean("userService", UserService.class);

// Call method to use the injected dependency

userService.processMessage();

}

}

**OUTPUT:**

