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

**pom.xml (Essential dependency)**

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.32</version>

</dependency>

</dependencies>

**GreetingService.java**

package com.example.app;

public class GreetingService {

public void greet() {

System.out.println("Hello! Welcome to the Spring framework.");

}

}

**spring-config.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 id="greetingService" class="com.example.app.GreetingService"/>

</beans>

**MainApp.java**

package com.example.app;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class MainApp {

public static void main(String[] args) {

ApplicationContext context = new ClassPathXmlApplicationContext("spring-config.xml");

GreetingService service = context.getBean("greetingService", GreetingService.class);

service.greet();

}

}

**Exercise 2: Using Dependency Injection in Spring**

**NotificationService.java**

package com.example.app;

public class NotificationService {

private String type;

public NotificationService(String type) {

this.type = type;

}

public void notifyUser() {

System.out.println("Sending notification via: " + type);

}

}

**spring-config.xml (Add this block)**

<bean id="notificationService" class="com.example.app.NotificationService">

<constructor-arg value="Email"/>

</bean>

**UserProfileService.java**

package com.example.app;

public class UserProfileService {

private String username;

public void setUsername(String username) {

this.username = username;

}

public void showProfile() {

System.out.println("User Profile: " + username);

}

}

**spring-config.xml (Add this block too)**

<bean id="userProfileService" class="com.example.app.UserProfileService">

<property name="username" value="jaisurya"/>

</bean>

**Add to MainApp.java**

UserProfileService profile = context.getBean("userProfileService", UserProfileService.class);

profile.showProfile();

NotificationService notifier = context.getBean("notificationService", NotificationService.class);

notifier.notifyUser();

**Exercise 4: Creating a Maven-Based Spring Project**

**Create Maven project from terminal:**

mvn archetype:generate -DgroupId=com.example.app -DartifactId=springapp \

-DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false

**Open project folder:**

cd springapp

**Modify pom.xml to include Spring dependency:**

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.32</version>

</dependency>

</dependencies>

**Add resource folder if not present:**  
Create src/main/resources for XML configuration files.

**Compile the project:**

mvn clean install

**Run your application (if using a plugin):**

mvn exec:java -Dexec.mainClass="com.example.app.MainApp"