**Spring Core – Load Country from Spring Configuration XML**

* **Objective:**

To demonstrate how to configure a bean in a Spring XML file and load it into a Java application using Spring’s ApplicationContext. The goal is to read country data from country.xml and log it.

* **Implementation Steps:**

**1. Create the Country Bean Class**

* A Java class Country is created in the package org.example.
* It contains two properties: code and name.
* Includes:
  + A no-arg constructor
  + Getter and setter methods
  + toString() method
  + Debug logging in constructor, getters, and setters

Purpose: This class represents the model object to be configured as a Spring bean.

**2. Create Spring Configuration File – country.xml**

* File placed in: src/main/resources
* Contains a <bean> definition for the Country object:

Purpose: Defines how Spring should create and initialize the Country object using Dependency Injection.

**3. Create the Main Class – SpringLearnApplication.java**

* A main method is added in SpringLearnApplication class under the same package.
* It contains the displayCountry() method that:
  + Loads country.xml using ClassPathXmlApplicationContext
  + Retrieves the country bean
  + Logs the object using toString()

**Purpose**: To demonstrate how Spring loads beans and manages object creation using XML configuration.

**4. Configure Logging**

* Added slf4j-api and slf4j-simple in pom.xml for logging
* Created simplelogger.properties in resources to set log level:

Purpose: To view the constructor and method invocation steps during Spring bean lifecycle.

**5. Run the Application**

* Run the main class
* Console output confirms:
  + Bean was created
  + Properties were injected
  + Log messages were triggered





