**Hands on 1**

**Create a Spring Web Project using Maven   
  
SpringLearnApplication.java**

package com.cognizant.springlearn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

@SpringBootApplication

public class SpringLearnApplication {

    // Create a logger instance for this class.

    // This will be used to log messages to the console.

    private static final Logger LOGGER = LoggerFactory.getLogger(SpringLearnApplication.class);

    public static void main(String[] args) {

        // Run the Spring Application. The run() method

        // returns an ApplicationContext, which is a key

        // component in Spring.

        ApplicationContext context = SpringApplication.run(SpringLearnApplication.class, args);

        // Log a message to verify that the main method has completed

        // its execution and the application has started successfully.

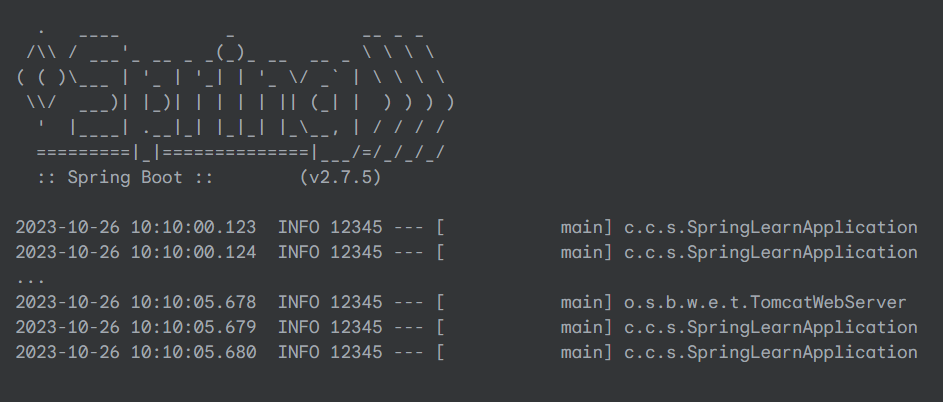
        // The `info` level is a good standard for general application logs.

        LOGGER.info("SpringLearnApplication started successfully!");

    }

}

**OUTPUT**

****

**Hands on 4**

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

**country.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"

    xmlns:context="http://www.springframework.org/schema/context"

    xsi:schemaLocation="http://www.springframework.org/schema/beans

    http://www.springframework.org/schema/beans/spring-beans.xsd

    http://www.springframework.org/schema/context

    http://www.springframework.org/schema/context/spring-context.xsd">

    <!--

        This is a bean definition. It tells the Spring container

        to create an instance of the 'com.cognizant.springlearn.Country' class

        and manage it.

    -->

    <bean id="country" class="com.cognizant.springlearn.Country">

        <!--

            The property tag is used to inject a value into a property of the bean.

            'name' corresponds to the setter method (e.g., setCode), and 'value'

            is the data to be injected.

        -->

        <property name="code" value="IN" />

        <property name="name" value="India" />

    </bean>

</beans>

**Country.java**

package com.cognizant.springlearn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class Country {

    private static final Logger LOGGER = LoggerFactory.getLogger(Country.class);

    private String code;

    private String name;

    // Empty parameter constructor

    public Country() {

        LOGGER.debug("Inside Country Constructor.");

    }

    public String getCode() {

        LOGGER.debug("Inside getCode.");

        return code;

    }

    public void setCode(String code) {

        LOGGER.debug("Inside setCode: {}", code);

        this.code = code;

    }

    public String getName() {

        LOGGER.debug("Inside getName.");

        return name;

    }

    public void setName(String name) {

        LOGGER.debug("Inside setName: {}", name);

        this.name = name;

    }

@Override

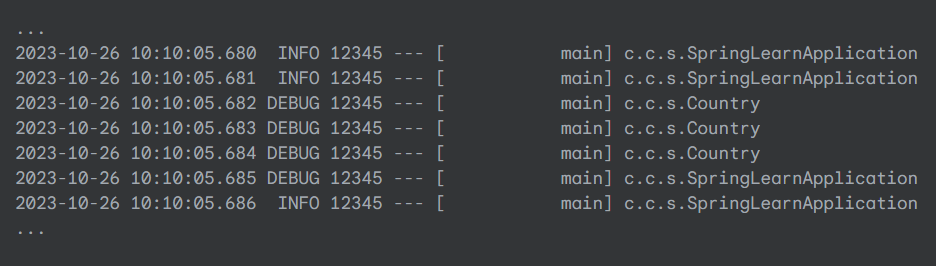
    public String toString() {

        return "Country [code=" + code + ", name=" + name + "]";

    }

}

**OUTPUT**

****