**WEEK 4:**

**Spring Rest**

**Exercise: Hello World RESTful Web Service:**

**CODE:**

**HelloController.java:**

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

public class HelloController {

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

    @GetMapping("/hello")

    public String sayHello() {

        LOGGER.info("START: sayHello()");

        String message = "Hello World!!";

        LOGGER.info("END: sayHello()");

        return message;

    }

}

**SpringLearnApplication.java:**

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SpringLearnApplication {

    public static void main(String[] args) {

        SpringApplication.run(SpringLearnApplication.class, args);

    }

}

**application.properties:**

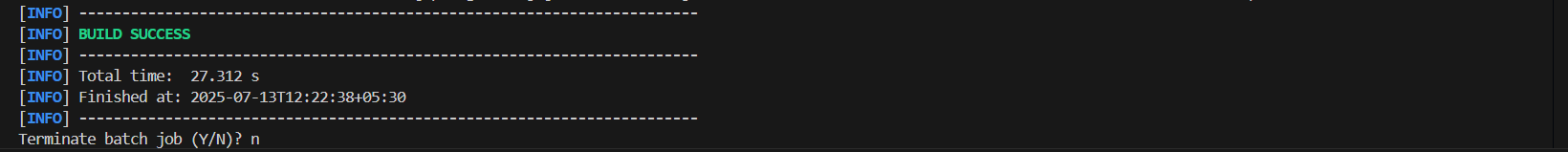
server.port=8083

spring.application.name=spring-learn

logging.level.org.springframework=INFO

logging.level.com.cognizant.spring\_learn=DEBUG

**Output:**

****

**Exercise: REST - Country Web Service:**

**CODE:**

**Country.java:**

package com.cognizant.spring\_learn.model;

public class Country {

    private String code;

    private String name;

    public Country() {}

    public Country(String code, String name) {

        this.code = code;

        this.name = name;

    }

    public String getCode() {

        return code;

    }

    public void setCode(String code) {

        this.code = code;

    }

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

}

**CountryController.java:**

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import com.cognizant.spring\_learn.model.Country;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

@RestController

public class CountryController {

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

    // Load the ApplicationContext once when the controller is created

    private final ApplicationContext context;

    public CountryController() {

        LOGGER.debug("Initializing ApplicationContext from country.xml");

        context = new ClassPathXmlApplicationContext("country.xml");

    }

    @RequestMapping("/country")

    public Country getCountryIndia() {

        LOGGER.info("START getCountryIndia");

        Country country = (Country) context.getBean("country");

        LOGGER.info("END getCountryIndia");

        return country;

    }

}

**application.properties:**

server.port=8083

spring.application.name=spring-learn

logging.level.org.springframework=INFO

logging.level.com.cognizant.spring\_learn=DEBUG

**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"

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

        https://www.springframework.org/schema/beans/spring-beans.xsd">

    <bean id="country" class="com.cognizant.spring\_learn.model.Country">

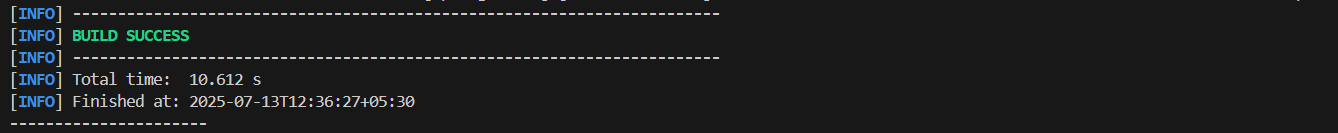
        <property name="code" value="IN"/>

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

    </bean>

</beans>

**Output:**

****

**Exercise:** **REST - Get country based on country code:**

**CODE:**

**CountryController.java:**

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.model.Country;

import com.cognizant.spring\_learn.service.CountryService;

import com.cognizant.spring\_learn.service.exception.CountryNotFoundException;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

@RestController

public class CountryController {

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

    @Autowired

    private CountryService countryService;

    @GetMapping("/countries/{code}")

    public Country getCountry(@PathVariable String code) throws CountryNotFoundException {

        LOGGER.info("START getCountry code: {}", code);

        Country country = countryService.getCountry(code);

        LOGGER.info("END getCountry");

        return country;

    }

}

**Country.java:**

package com.cognizant.spring\_learn.model;

public class Country {

    private String code;

    private String name;

    public Country() {}

    public Country(String code, String name) {

        this.code = code;

        this.name = name;

    }

    public String getCode() {

        return code;

    }

    public void setCode(String code) {

        this.code = code;

    }

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

}

**CountryNotFoundException.java:**

package com.cognizant.spring\_learn.service.exception;

import org.springframework.http.HttpStatus;

import org.springframework.web.bind.annotation.ResponseStatus;

@ResponseStatus(value = HttpStatus.NOT\_FOUND, reason = "Country not found")

public class CountryNotFoundException extends Exception {

    public CountryNotFoundException(String message) {

        super(message);

    }

}

**CountryService.java:**

package com.cognizant.spring\_learn.service;

import java.util.List;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Service;

import com.cognizant.spring\_learn.model.Country;

import com.cognizant.spring\_learn.service.exception.CountryNotFoundException;

@Service

public class CountryService {

    private final List<Country> countries;

    public CountryService() {

        try (ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext("country.xml")) {

            @SuppressWarnings("unchecked")

            List<Country> loadedCountries = (List<Country>) context.getBean("countryList");

            this.countries = loadedCountries;

        }

    }

    public Country getCountry(String code) throws CountryNotFoundException {

        return countries.stream()

                .filter(c -> c.getCode().equalsIgnoreCase(code))

                .findFirst()

                .orElseThrow(() -> new CountryNotFoundException("Country not found"));

    }

}

**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"

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

       https://www.springframework.org/schema/beans/spring-beans.xsd">

    <bean id="country" class="com.cognizant.spring\_learn.model.Country">

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

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

    </bean>

    <bean id="countryList" class="java.util.ArrayList">

        <constructor-arg>

            <list>

                <bean class="com.cognizant.spring\_learn.model.Country">

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

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

                </bean>

                <bean class="com.cognizant.spring\_learn.model.Country">

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

                    <property name="name" value="United States"/>

                </bean>

                <bean class="com.cognizant.spring\_learn.model.Country">

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

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

                </bean>

                <bean class="com.cognizant.spring\_learn.model.Country">

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

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

                </bean>

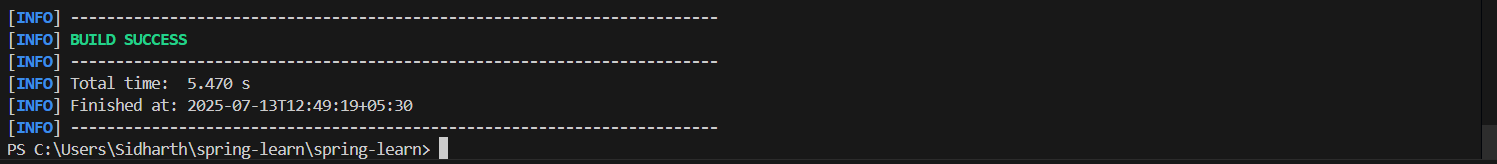
            </list>

        </constructor-arg>

    </bean>

</beans>

**Output:**

****