**Handson 4 :REST - Get country based on country code**   
  
Write a REST service that returns a specific country based on country code. The country code should be case insensitive.  
  
**Controller**: com.cognizant.spring-learn.controller.CountryController  
**Method Annotation:** @GetMapping("/countries/{code}")  
**Method Name**: getCountry(String code)  
**Method Implemetation**: Invoke countryService.getCountry(code)   
**Service Method:**com.cognizant.spring-learn.service.CountryService.getCountry(String code)  
  
**Service Method Implementation**:

* Get the country code using @PathVariable
* Get country list from country.xml
* Iterate through the country list
* Make a case insensitive matching of country code and return the country.
* Lambda expression can also be used instead of iterating the country list

**Sample Request**: http://localhost:8083/country/in  
  
**Sample Response**:

{

  "code": "IN",

  "name": "India"

}

Solution:

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;

}

}

Country.xml

<countries>

<country>

<code>IN</code>

<name>India</name>

</country>

<country>

<code>US</code>

<name>United States</name>

</country>

<country>

<code>JP</code>

<name>Japan</name>

</country>

</countries>

CountryService.java

package com.cognizant.spring\_learn.service;

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

import org.springframework.core.io.ClassPathResource;

import org.springframework.stereotype.Service;

import org.w3c.dom.\*;

import javax.xml.parsers.DocumentBuilder;

import javax.xml.parsers.DocumentBuilderFactory;

import java.util.ArrayList;

import java.util.List;

@Service

public class CountryService {

public Country getCountry(String code) throws Exception {

List<Country> countryList = loadCountries();

return countryList.stream()

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

.findFirst()

.orElseThrow(() -> new Exception("Country Not Found"));

}

private List<Country> loadCountries() throws Exception {

List<Country> countries = new ArrayList<>();

DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();

DocumentBuilder db = dbf.newDocumentBuilder();

Document doc = db.parse(new ClassPathResource("country.xml").getInputStream());

NodeList nodeList = doc.getElementsByTagName("country");

for (int i = 0; i < nodeList.getLength(); i++) {

Node nNode = nodeList.item(i);

if (nNode.getNodeType() == Node.ELEMENT\_NODE) {

Element element = (Element) nNode;

String code = element.getElementsByTagName("code").item(0).getTextContent();

String name = element.getElementsByTagName("name").item(0).getTextContent();

countries.add(new Country(code, name));

}

}

return countries;

}

}

CountryController.java

package com.cognizant.spring\_learn.controller;

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

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

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

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

@RestController

public class CountryController {

@Autowired

private CountryService countryService;

@GetMapping("/countries/{code}")

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

return countryService.getCountry(code);

}

}

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);

}

}

Output:

