

REST - Get Country Based on Country Code

Controller: 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.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 CountryNotFoundException
    {
        return countryService.getCountry(code);
    }
}
```

Service: CountryService.java

```
package com.cognizant.spring_learn.service;

import com.cognizant.spring_learn.model.Country;
```

```

import com.cognizant.spring_learn.service.exception.CountryNotFoundException;
import org.springframework.stereotype.Service;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import java.util.List;

@Service
public class CountryService {

    public Country getCountry(String code) throws CountryNotFoundException {

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

        List<Country> countryList = context.getBean("countryList", List.class);
        context.close();

        return countryList.stream()

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

            .findFirst()

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

```

Exception: 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);
    }
}

```

Sample country.xml

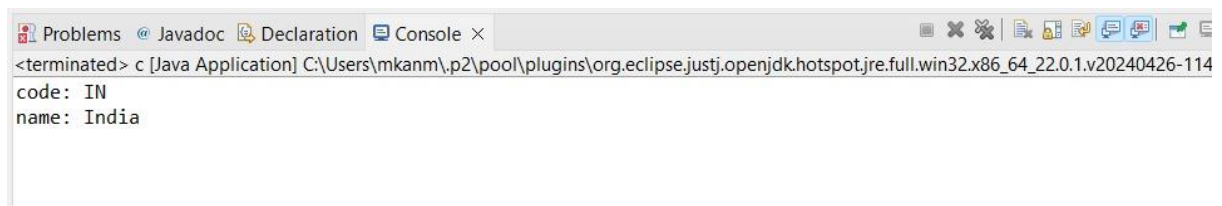
```

<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:context="http://www.springframework.org/schema/context"
    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="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>
            </list>
        </constructor-arg>
    </bean>
</beans>

```

OUTPUT:



The screenshot shows the Eclipse IDE's console window. The title bar includes tabs for 'Problems', 'Javadoc', 'Declaration', and 'Console'. The console text shows a terminated Java application with the following output:

```
<terminated> c [Java Application] C:\Users\mkanm\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-114  
code: IN  
name: India
```