**Create a Spring Web Project using Maven:**

package com.cognizant.spring\_learn;  
  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
  
@SpringBootApplication  
public class SpringLearnApplication {  
  
 private static final Logger LOGGER = LoggerFactory.getLogger(SpringLearnApplication.class);  
  
 public static void main(String[] args) {  
 LOGGER.info("START");  
 SpringApplication.run(SpringLearnApplication.class, args);  
 LOGGER.info("END");  
 }  
}

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

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;  
  
 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()");  
 this.code = code;  
 }  
  
 public String getName() {  
 *LOGGER*.debug("Inside getName()");  
 return name;  
 }  
  
 public void setName(String name) {  
 *LOGGER*.debug("Inside setName()");  
 this.name = name;  
 }  
  
 @Override  
 public String toString() {  
 return "Country [code=" + code + ", name=" + name + "]";  
 }  
}

country.xml

<?xml version="1.0" encoding="UTF-8"?>  
<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   
https://www.springframework.org/schema/beans/spring-beans.xsd  
http://www.springframework.org/schema/context   
https://www.springframework.org/schema/context/spring-context.xsd">  
  
 <!-- Configure country bean -->  
 <bean id="country" class="com.cognizant.springlearn.Country">  
 <property name="code" value="IN" />  
 <property name="name" value="India" />  
 </bean>  
  
</beans>

package com.cognizant.springlearn;  
  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
public class SpringLearnApplication {  
  
 private static final Logger *LOGGER* = LoggerFactory.getLogger(SpringLearnApplication.class);  
  
 public static void main(String[] args) {  
 *LOGGER*.info("START main");  
 *displayCountry*();  
 *LOGGER*.info("END main");  
 }  
  
 public static void displayCountry() {  
 ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");  
 Country country = context.getBean("country", Country.class);  
 *LOGGER*.debug("Country : {}", country.toString());  
 }  
}

pom.xml

<dependencies>  
 <!-- Spring Core -->  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-context</artifactId>  
 <version>5.3.30</version>  
 </dependency>  
  
 <!-- SLF4J API -->  
 <dependency>  
 <groupId>org.slf4j</groupId>  
 <artifactId>slf4j-api</artifactId>  
 <version>1.7.36</version>  
 </dependency>  
  
 <!-- SLF4J Simple Logger -->  
 <dependency>  
 <groupId>org.slf4j</groupId>  
 <artifactId>slf4j-simple</artifactId>  
 <version>1.7.36</version>  
 </dependency>  
</dependencies>

**Hello World RESTful Web Service**

package com.cognizant.spring-learn.controller;

import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.web.bind.annotation.\*;  
  
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;  
 }  
}

**Response in browser:**

Hello World!!

**REST - Country Web Service**

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

<?xml version="1.0" encoding="UTF-8"?>  
<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   
 https://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context   
 https://www.springframework.org/schema/context/spring-context.xsd">  
  
 <!-- India bean -->  
 <bean id="in" class="com.cognizant.spring-learn.model.Country">  
 <property name="code" value="IN"/>  
 <property name="name" value="India"/>  
 </bean>  
  
</beans>

package com.cognizant.spring-learn.controller;  
  
import com.cognizant.spring-learn.model.Country;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
import org.springframework.web.bind.annotation.\*;  
  
@RestController  
public class CountryController {  
  
 private static final Logger *LOGGER* = LoggerFactory.getLogger(CountryController.class);  
  
 @RequestMapping("/country")  
 public Country getCountryIndia() {  
 *LOGGER*.info("START: getCountryIndia()");  
 ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");  
 Country india = (Country) context.getBean("in");  
 *LOGGER*.info("END: getCountryIndia()");  
 return india;  
 }  
}

**Sample Response:**

{

"code": "IN",

"name": "India"

}

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

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;  
 }  
}

<?xml version="1.0" encoding="UTF-8"?>  
<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   
https://www.springframework.org/schema/beans/spring-beans.xsd  
http://www.springframework.org/schema/context   
https://www.springframework.org/schema/context/spring-context.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>  
 <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>

package com.cognizant.spring-learn.service;  
  
import com.cognizant.spring-learn.model.Country;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
import org.springframework.stereotype.Service;  
  
import java.util.List;  
  
@Service  
public class CountryService {  
  
 public Country getCountry(String code) {  
 ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");  
 List<Country> countryList = (List<Country>) context.getBean("countryList");  
  
 // Using lambda for case-insensitive match  
 return countryList.stream()  
 .filter(country -> country.getCode().equalsIgnoreCase(code))  
 .findFirst()  
 .orElse(null); // or throw a custom CountryNotFoundException  
 }  
}

package com.cognizant.spring-learn.controller;  
  
import com.cognizant.spring-learn.model.Country;  
import com.cognizant.spring-learn.service.CountryService;  
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) {  
 *LOGGER*.info("START: getCountry()");  
 Country country = countryService.getCountry(code);  
 *LOGGER*.info("END: getCountry()");  
 return country;  
 }  
}

**Sample response:**

{

"code": "IN",

"name": "India"

}

**Create authentication service that returns JWT:**

<dependencies>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-security</artifactId>  
 </dependency>  
 <dependency>  
 <groupId>io.jsonwebtoken</groupId>  
 <artifactId>jjwt</artifactId>  
 <version>0.9.0</version>  
 </dependency>  
</dependencies>

package com.cognizant.springlearn.controller;  
  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.web.bind.annotation.\*;  
  
import java.util.\*;  
import io.jsonwebtoken.\*;  
  
@RestController  
public class AuthenticationController {  
  
 private static final Logger *LOGGER* = LoggerFactory.getLogger(AuthenticationController.class);  
  
 @GetMapping("/authenticate")  
 public Map<String, String> authenticate(@RequestHeader("Authorization") String authHeader) {  
 *LOGGER*.info("START - /authenticate");  
 *LOGGER*.debug("Authorization Header: {}", authHeader);  
  
 String user = getUser(authHeader);  
 *LOGGER*.debug("Decoded User: {}", user);  
  
 String token = generateJwt(user);  
  
 Map<String, String> map = new HashMap<>();  
 map.put("token", token);  
  
 *LOGGER*.info("END - /authenticate");  
 return map;  
 }  
  
 private String getUser(String authHeader) {  
 *LOGGER*.debug("START - getUser()");  
 String base64Credentials = authHeader.substring("Basic ".length());  
 byte[] decodedBytes = Base64.*getDecoder*().decode(base64Credentials);  
 String decodedCredentials = new String(decodedBytes);  
  
 *LOGGER*.debug("Decoded Credentials: {}", decodedCredentials);  
  
 String username = decodedCredentials.split(":")[0];  
 *LOGGER*.debug("Extracted Username: {}", username);  
  
 return username;  
 }  
  
 private String generateJwt(String user) {  
 *LOGGER*.debug("START - generateJwt()");  
 JwtBuilder builder = Jwts.builder();  
 builder.setSubject(user);  
 builder.setIssuedAt(new Date());  
 builder.setExpiration(new Date((new Date()).getTime() + 20 \* 60 \* 1000)); // 20 mins  
 builder.signWith(SignatureAlgorithm.HS256, "secretkey");  
  
 String token = builder.compact();  
 *LOGGER*.debug("Generated Token: {}", token);  
 return token;  
 }  
}

package com.cognizant.springlearn.config;  
  
import org.springframework.context.annotation.Configuration;  
import org.springframework.security.config.annotation.authentication.builders.AuthenticationManagerBuilder;  
import org.springframework.security.config.annotation.web.builders.HttpSecurity;  
import org.springframework.security.config.annotation.web.configuration.\*;  
  
@Configuration  
@EnableWebSecurity  
public class SecurityConfig extends WebSecurityConfigurerAdapter {  
  
 @Override  
 protected void configure(AuthenticationManagerBuilder auth) throws Exception {  
 auth.inMemoryAuthentication()  
 .withUser("user").password("{noop}pwd").roles("USER")  
 .and()  
 .withUser("admin").password("{noop}adminpwd").roles("ADMIN");  
 }  
  
 @Override  
 protected void configure(HttpSecurity http) throws Exception {  
 http.httpBasic()  
 .and()  
 .authorizeRequests()  
 .antMatchers("/countries").hasRole("USER")  
 .antMatchers("/authenticate").hasAnyRole("USER", "ADMIN")  
 .and()  
 .csrf().disable();  
 }  
}

package com.cognizant.springlearn;  
  
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);  
 }  
}

**CURL command:**

curl -s -u user:pwd http://localhost:8090/authenticate