**Spring REST using Spring Boot 3**

**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);

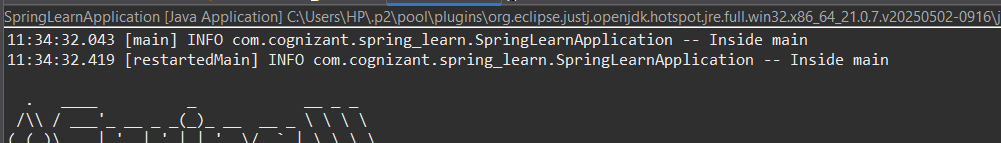
public static void main(String[] args) {

***LOGGER***.info("Inside main");

SpringApplication.*run*(SpringLearnApplication.class, args);

}

}



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

SpringLearnApplication.java:

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

*@SpringBootApplication*

public class SpringLearnApplication {

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

public static void main(String[] args) {

***LOGGER***.info("Inside main");

SpringApplication.*run*(SpringLearnApplication.class, args);

*displayCountry*();

}

public static void displayCountry() {

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

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

***LOGGER***.debug("Country : {}", country.toString());

}

}

Country.java:

package com.cognizant.spring\_learn;

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;

}

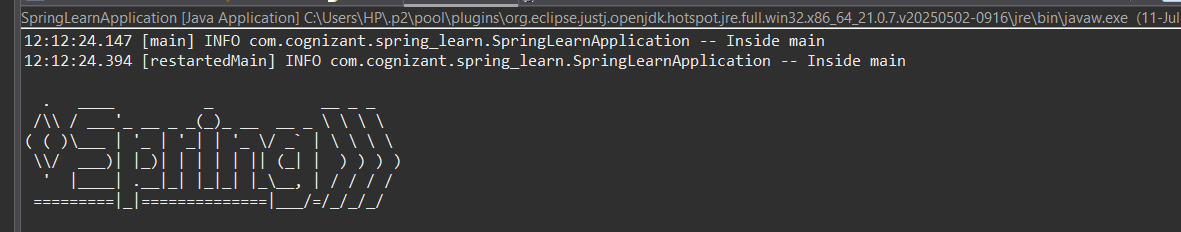
*@Override*

public String toString() {

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

}

}



**Hello World RESTful Web Service**

SpringLearnApplication.java:

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("Inside main");

SpringApplication.*run*(SpringLearnApplication.class, args);

}

}

HelloController.java:

package com.cognizant.spring\_learn.controller;

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()");

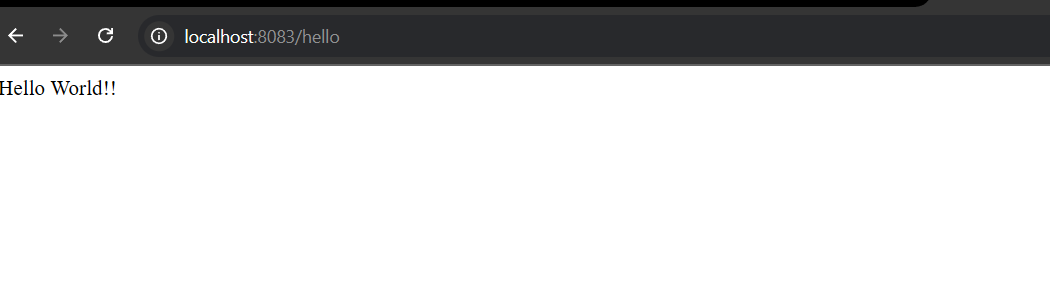
String response = "Hello World!!";

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

return response;

}

}



**REST - Country Web Service**CountryController.java:

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.RequestMapping;

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

*@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 country = context.getBean("country", Country.class);

***LOGGER***.info("END - getCountryIndia()");

return country;

}

}

Country.java:

package com.cognizant.spring\_learn.model;

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("Getting code");

return code;

}

public void setCode(String code) {

***LOGGER***.debug("Setting code");

this.code = code;

}

public String getName() {

***LOGGER***.debug("Getting name");

return name;

}

public void setName(String name) {

***LOGGER***.debug("Setting name");

this.name = name;

}

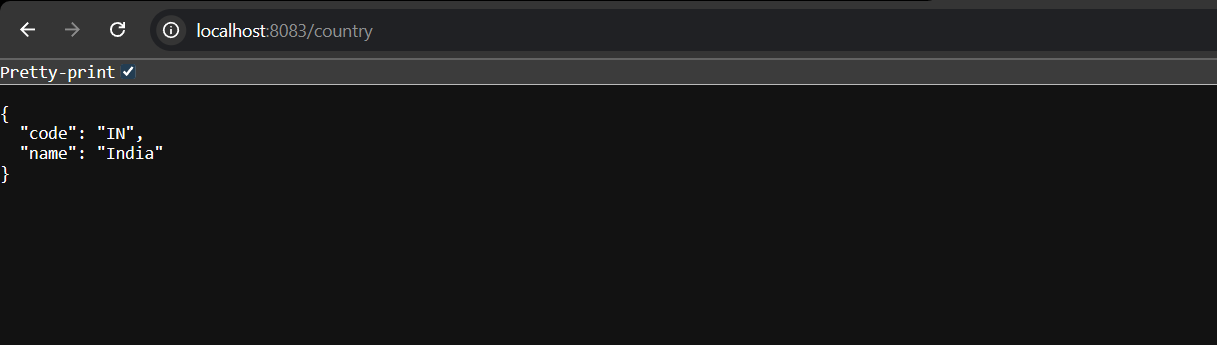
*@Override*

public String toString() {

return "Country{" + "code='" + code + '\'' + ", name='" + name + '\'' + '}';

}

}



**REST - Get country based on country code**CountryController.java:

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*("/country/{code}")

public Country getCountry(*@PathVariable* String code) {

***LOGGER***.info("Start getCountry with code: {}", code);

Country country = countryService.getCountry(code);

***LOGGER***.info("End getCountry");

return country;

}

}

CountryService.java:

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> countries = context.getBean("countryList", List.class);

return countries.stream()

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

.findFirst()

.orElse(null);

}

}

country.xml:

<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

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

<bean id="country1" class="com.cognizant.spring\_learn.model.Country">

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

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

</bean>

<bean id="country2" class="com.cognizant.spring\_learn.model.Country">

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

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

</bean>

<bean id="country3" class="com.cognizant.spring\_learn.model.Country">

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

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

</bean>

<bean id="country4" class="com.cognizant.spring\_learn.model.Country">

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

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

</bean>

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

<constructor-arg>

<list>

<ref bean="country1" />

<ref bean="country2" />

<ref bean="country3" />

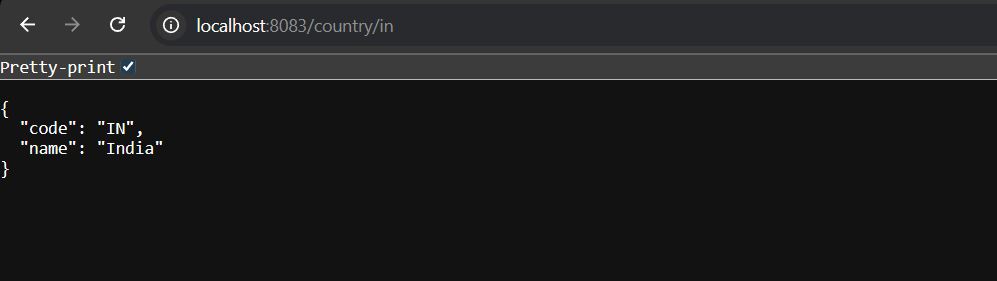
<ref bean="country4" />

</list>

</constructor-arg>

</bean>

</beans>



**Create authentication service that returns JWT**

SecurityConfig.java:

package com.cognizant.spring\_learn.config;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;

import org.springframework.security.config.annotation.web.builders.HttpSecurity;

import org.springframework.security.core.userdetails.User;

import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.UserDetailsService;

import org.springframework.security.provisioning.InMemoryUserDetailsManager;

import org.springframework.security.web.SecurityFilterChain;

import org.springframework.security.config.Customizer;

*@Configuration*

*@EnableWebSecurity*

public class SecurityConfig {

*@Bean*

public UserDetailsService userDetailsService() {

UserDetails user = User

.*withUsername*("user")

.password("{noop}pwd")

.roles("USER")

.build();

return new InMemoryUserDetailsManager(user);

}

*@Bean*

public SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {

http

.authorizeHttpRequests(auth -> auth

.requestMatchers("/authenticate").permitAll()

.anyRequest().authenticated()

)

.httpBasic(Customizer.*withDefaults*())

.csrf(csrf -> csrf.disable());

return http.build();

}

}

AuthenticationController.java:

package com.cognizant.spring\_learn.controller;

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

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

import org.springframework.http.ResponseEntity;

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

*@RestController*

public class AuthenticationController {

*@Autowired*

private JwtService jwtService;

*@GetMapping*("/authenticate")

public ResponseEntity<String> authenticate(*@RequestHeader*("Authorization") String authHeader) {

String token = jwtService.generateTokenFromHeader(authHeader);

return ResponseEntity.*ok*(token);

}

}

JwtService.java:

package com.cognizant.spring\_learn.service;

import io.jsonwebtoken.Jwts;

import io.jsonwebtoken.SignatureAlgorithm;

import io.jsonwebtoken.security.Keys;

import org.springframework.stereotype.Service;

import javax.crypto.SecretKey;

*@Service*

public class JwtService {

private final SecretKey key;

public JwtService() {

// Generate a secure random key for HS256

this.key = Keys.*secretKeyFor*(*SignatureAlgorithm*.***HS256***);

}

public String generateTokenFromHeader(String username) {

return Jwts.*builder*()

.setSubject(username)

.signWith(key)

.compact();

}

}

