**SuperSet ID: 6376594**

Spring REST Using Spring Boot(Hands-On)

**Exercise-1: Create a Spring Web Project using Maven**

Project Name:spring-learn

SpringLearnApplication.java Code:

**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);

System.***out***.println("✅ Main method executed - SpringLearnApplication started successfully.");

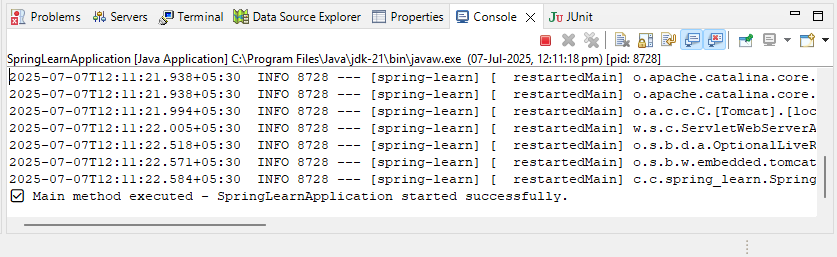
}

}

Expected Output:

✅ Main method executed - SpringLearnApplication started successfully.

Output:



**Exercise-4: Spring Core – Load Country from Spring Configuration XML**

Project Name:spring-learn

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**import** org.springframework.context.ApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**public** **class** SpringLearnApplication {

**public** **static** **void** main(String[] args) {

System.***out***.println("START");

*displayCountry*();

System.***out***.println("END");

}

**public** **static** **void** displayCountry() {

// Load the XML configuration file

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

// Get the country bean

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

// Print country details

System.***out***.println("Country : " + country.toString());

}

}

Country.java Code:

**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;

// Constructor

**public** Country() {

***LOGGER***.debug("Inside Country Constructor.");

}

// Getters and Setters

**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;

}

// toString method

@Override

**public** String toString() {

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

}

}

country.xml Code:

<?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*

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

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

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

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

</bean>

</beans>

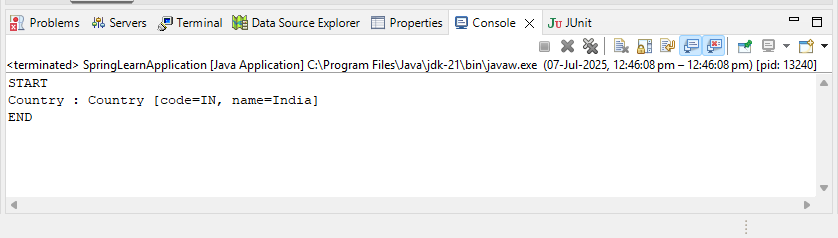
Excepted Output:

START

Country : Country [code=IN, name=India]

END

Output:



**Exercise-2:Hello World RESTful Web Service**

Project Name:spring-learn

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**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 {

**public** **static** **void** main(String[] args) {

// Start Spring Boot Application (for REST API)

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

// Display country details from XML config (Hands-on 4)

System.***out***.println("START");

*displayCountry*();

System.***out***.println("END");

}

**public** **static** **void** displayCountry() {

// Load Spring XML configuration from src/main/resources/country.xml

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

// Get the 'country' bean from XML

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

// Print country details

System.***out***.println("Country : " + country.toString());

}

}

Country.java Code:

**package** com.cognizant.spring\_learn;

**public** **class** Country {

**private** String code;

**private** String name;

**public** Country() {

System.***out***.println("Inside Country Constructor");

}

**public** String getCode() {

System.***out***.println("Inside getCode()");

**return** code;

}

**public** **void** setCode(String code) {

System.***out***.println("Inside setCode()");

**this**.code = code;

}

**public** String getName() {

System.***out***.println("Inside getName()");

**return** name;

}

**public** **void** setName(String name) {

System.***out***.println("Inside setName()");

**this**.name = name;

}

@Override

**public** String toString() {

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

}

}

HelloController.java Code:

**package** com.cognizant.spring\_learn.controller;

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

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

@RestController

**public** **class** HelloController {

@GetMapping("/hello")

**public** String sayHello() {

System.***out***.println("START");

String response = "Hello World!!";

System.***out***.println("END");

**return** response;

}

}

application.properties Code:

spring.application.name=spring-learn

server.port=8083

country.xml Code:

<?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*

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

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

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

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

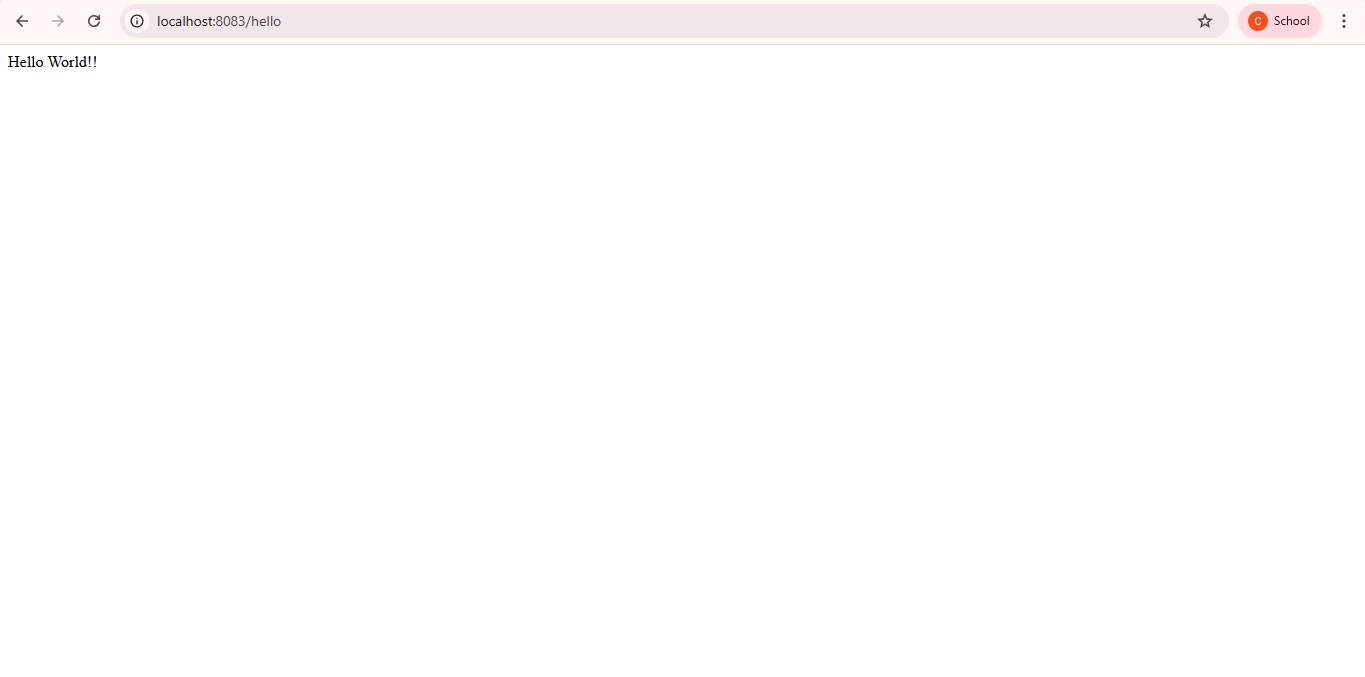
</bean>

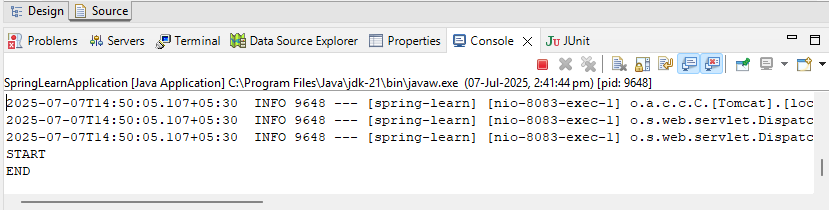
</beans>

Expected Output:

Hello World!!

Output:





**Exercise-3: REST - Country Web Service**

Project Name:spring-learn

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**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 {

**public** **static** **void** main(String[] args) {

// Start Spring Boot Application (for REST API)

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

// Display country details from XML config (Hands-on 4)

System.***out***.println("START");

*displayCountry*();

System.***out***.println("END");

}

**public** **static** **void** displayCountry() {

// Load Spring XML configuration from src/main/resources/country.xml

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

// Get the 'country' bean from XML

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

// Print country details

System.***out***.println("Country : " + country.toString());

}

}

Country.java Code:

**package** com.cognizant.spring\_learn;

**public** **class** Country {

**private** String code;

**private** String name;

**public** Country() {

System.***out***.println("Inside Country Constructor");

}

**public** String getCode() {

System.***out***.println("Inside getCode()");

**return** code;

}

**public** **void** setCode(String code) {

System.***out***.println("Inside setCode()");

**this**.code = code;

}

**public** String getName() {

System.***out***.println("Inside getName()");

**return** name;

}

**public** **void** setName(String name) {

System.***out***.println("Inside setName()");

**this**.name = name;

}

@Override

**public** String toString() {

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

}

}

CountryController.java Code:

**package** com.cognizant.spring\_learn.controller;

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

**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 {

@RequestMapping("/country")

**public** Country getCountryIndia() {

System.***out***.println("START");

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

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

System.***out***.println("END");

**return** country;

}

}

Country.xml Code:

<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=*"country"* class=*"com.cognizant.spring\_learn.Country"*>

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

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

</bean>

</beans>

application.properties Code:

server.port=9091

spring.application.name=spring-learn

Expected Output:

{

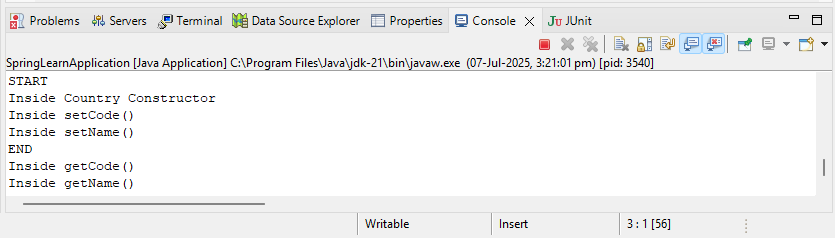
"code": "IN",

"name": "India"

}

Output:





**Exercise-3: REST - Get country based on country code**   
Project Name:spring-learn

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**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 {

**public** **static** **void** main(String[] args) {

// Start Spring Boot Application (for REST API)

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

// Display country details from XML config (Hands-on 4)

System.***out***.println("START");

*displayCountry*();

System.***out***.println("END");

}

**public** **static** **void** displayCountry() {

// Load Spring XML configuration from src/main/resources/country.xml

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

// Get the 'country' bean from XML

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

// Print country details

System.***out***.println("Country : " + country.toString());

}

}

country.xml Code:

<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=*"country"* class=*"com.cognizant.spring\_learn.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.Country"*>

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

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

</bean>

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

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

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

</bean>

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

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

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

</bean>

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

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

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

</bean>

</list>

</constructor-arg>

</bean>

</beans>

CountryService.java Code:

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

**import** java.util.List;

**import** org.springframework.context.ApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**import** org.springframework.stereotype.Service;

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

@Service

**public** **class** CountryService {

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

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

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

**for** (Country c : countries) {

**if** (c.getCode().equalsIgnoreCase(code)) {

**return** c;

}

}

**return** **null**;

}

**public** List<Country> getAllCountries() {

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

**return** context.getBean("countryList", List.**class**);

}

}

CountryController.java Code:

**package** com.cognizant.spring\_learn.controller;

**import** java.util.List;

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

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

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

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

@RestController

**public** **class** CountryController {

@Autowired

**private** CountryService countryService;

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

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

**return** countryService.getCountry(code);

}

@GetMapping("/countries")

**public** List<Country> getAllCountries() {

**return** countryService.getAllCountries();

}

}

Country.java Code:

**package** com.cognizant.spring\_learn;

**public** **class** Country {

**private** String code;

**private** String name;

**public** Country() {

System.***out***.println("Inside Country Constructor");

}

**public** String getCode() {

System.***out***.println("Inside getCode()");

**return** code;

}

**public** **void** setCode(String code) {

System.***out***.println("Inside setCode()");

**this**.code = code;

}

**public** String getName() {

System.***out***.println("Inside getName()");

**return** name;

}

**public** **void** setName(String name) {

System.***out***.println("Inside setName()");

**this**.name = name;

}

@Override

**public** String toString() {

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

}

}

application.properties Code:

server.port=9092

spring.application.name=spring-learn

Expected Output:

{

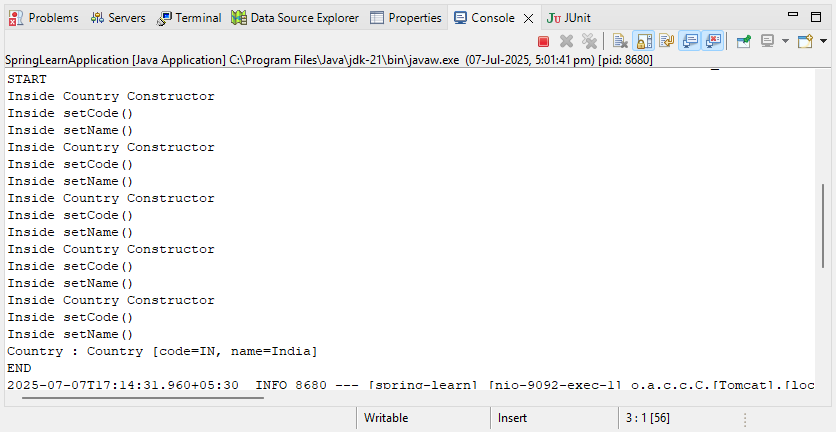
"code": "IN",

"name": "India"

}

Output:





**Exercise-3: Create authentication service that returns JWT**

Project Name:spring-learn

pom.xml Code:

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>3.5.3</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<groupId>com.cognizant</groupId>

<artifactId>spring-learn</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>spring-learn</name>

<description>Demo project for Spring Boot</description>

<properties>

<java.version>21</java.version>

</properties>

<dependencies>

<!-- Spring Web -->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<!-- Spring Security -->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-security</artifactId>

</dependency>

<!-- JWT - Java JSON Web Token library -->

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt</artifactId>

<version>0.9.1</version>

</dependency>

<!-- DevTools (for hot reload) -->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<!-- Spring Boot Test -->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

SecurityConfig.java Code:

**package** com.cognizant.spring\_learn.config;

**import** org.springframework.context.annotation.Configuration;

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

**import** org.springframework.security.web.SecurityFilterChain;

**import** org.springframework.context.annotation.Bean;

@Configuration

**public** **class** SecurityConfig {

@Bean

**public** SecurityFilterChain filterChain(HttpSecurity http) **throws** Exception {

http.~~csrf~~().disable()

.~~authorizeHttpRequests~~()

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

.anyRequest().authenticated()

.~~and~~()

.~~httpBasic~~();

**return** http.build();

}

}

JwtUtil.java Code:

**package** com.cognizant.spring\_learn.util;

**import** io.jsonwebtoken.Jwts;

**import** io.jsonwebtoken.SignatureAlgorithm;

**import** java.util.Date;

**public** **class** JwtUtil {

**private** **static** **final** String ***SECRET\_KEY*** = "secret"; // Use a secure key in real apps

**public** **static** String generateToken(String username) {

**return** Jwts.*builder*()

.setSubject(username)

.setIssuedAt(**new** Date(System.*currentTimeMillis*()))

.setExpiration(**new** Date(System.*currentTimeMillis*() + 1000 \* 60 \* 60)) // 1 hour

.signWith(SignatureAlgorithm.***HS256***, ***SECRET\_KEY***)

.compact();

}

}

AuthenticationController.java Code:

**package** com.cognizant.spring\_learn.controller;

**import** org.springframework.http.ResponseEntity;

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

**import** com.cognizant.spring\_learn.util.JwtUtil;

**import** java.util.Base64;

@RestController

**public** **class** AuthenticationController {

@RequestMapping(value = "/authenticate", method = RequestMethod.***GET***)

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

// Basic base64 encoded string: "user:pwd"

**if** (authHeader != **null** && authHeader.startsWith("Basic ")) {

String base64Credentials = authHeader.substring("Basic ".length());

**byte**[] credDecoded = Base64.*getDecoder*().decode(base64Credentials);

String credentials = **new** String(credDecoded);

String[] values = credentials.split(":", 2);

String username = values[0];

String password = values[1];

// Simple check

**if** (username.equals("user") && password.equals("pwd")) {

String token = JwtUtil.*generateToken*(username);

**return** ResponseEntity.*ok*().body("{\"token\":\"" + token + "\"}");

}

}

**return** ResponseEntity.*status*(401).body("Unauthorized");

}

}

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**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 {

**public** **static** **void** main(String[] args) {

// Start Spring Boot Application (for REST API)

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

// Display country details from XML config (Hands-on 4)

System.***out***.println("START");

*displayCountry*();

System.***out***.println("END");

}

**public** **static** **void** displayCountry() {

// Load Spring XML configuration from src/main/resources/country.xml

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

// Get the 'country' bean from XML

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

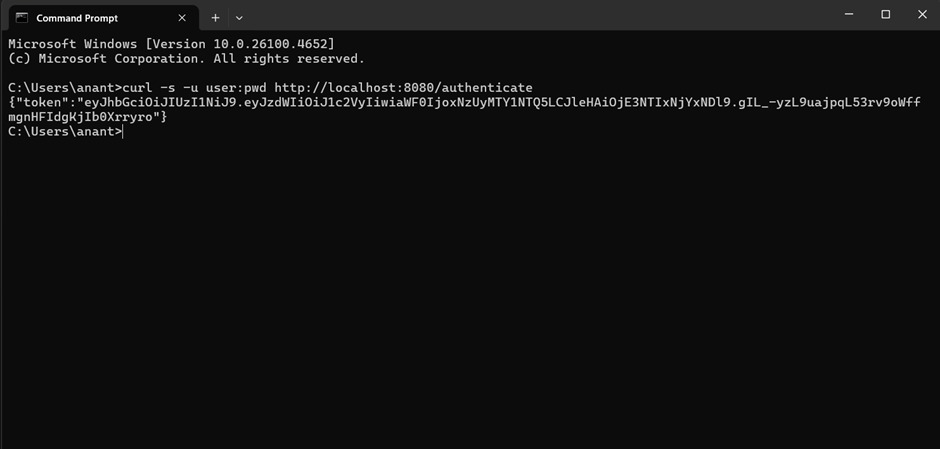
// Print country details

System.***out***.println("Country : " + country.toString());

}

}

Output:



Additional Hands-On:

**Exercise-1: Display Employee List and Edit Employee form using RESTful Web Service**

Project Name:spring-learn

Employee.xml Code:

<?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=*"employeeList"* class=*"java.util.ArrayList"*>

<constructor-arg>

<list>

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

<property name=*"id"* value=*"1"*/>

<property name=*"name"* value=*"John"*/>

<property name=*"salary"* value=*"50000"*/>

<property name=*"permanent"* value=*"true"*/>

</bean>

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

<property name=*"id"* value=*"2"*/>

<property name=*"name"* value=*"Jane"*/>

<property name=*"salary"* value=*"60000"*/>

<property name=*"permanent"* value=*"false"*/>

</bean>

</list>

</constructor-arg>

</bean>

</beans>

Employee.java Code:

**package** com.cognizant.spring\_learn.model;

**public** **class** Employee {

**private** **int** id;

**private** String name;

**private** **double** salary;

**private** **boolean** permanent;

// Getters and Setters

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **double** getSalary() {

**return** salary;

}

**public** **void** setSalary(**double** salary) {

**this**.salary = salary;

}

**public** **boolean** isPermanent() {

**return** permanent;

}

**public** **void** setPermanent(**boolean** permanent) {

**this**.permanent = permanent;

}

}

EmployeeService.java Code:

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

**import** com.cognizant.spring\_learn.model.Employee;

**import** org.springframework.context.ApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**import** org.springframework.stereotype.Service;

**import** java.util.List;

@Service

**public** **class** EmployeeService {

**public** List<Employee> getAllEmployees() {

ApplicationContext context = **new** ClassPathXmlApplicationContext("employee.xml");

**return** (List<Employee>) context.getBean("employeeList");

}

**public** Employee getEmployeeById(**int** id) {

**return** getAllEmployees().stream()

.filter(emp -> emp.getId() == id)

.findFirst()

.orElse(**null**);

}

}

EmployeeController.java Code:

**package** com.cognizant.spring\_learn.controller;

**import** com.cognizant.spring\_learn.model.Employee;

**import** com.cognizant.spring\_learn.service.EmployeeService;

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

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

**import** java.util.List;

@RestController

**public** **class** EmployeeController {

@Autowired

**private** EmployeeService employeeService;

@GetMapping("/employees")

**public** List<Employee> getAllEmployees() {

**return** employeeService.getAllEmployees();

}

@GetMapping("/employees/{id}")

**public** Employee getEmployeeById(@PathVariable **int** id) {

**return** employeeService.getEmployeeById(id);

}

}

application.properties Code:

server.port=8092

spring.application.name=spring-learn

SpringLearnApplication.java Code:

**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);

System.***out***.println("✅ SpringLearnApplication started successfully!");

}

}

Expected Output:

[

{

"id": 1,

"name": "John",

"salary": 50000,

"permanent": true

},

{

"id": 2,

"name": "Jane",

"salary": 60000,

"permanent": false

}

]

Output:



**Exercise-2: Create static employee list data using spring xml configuration**

Project Name:spring-learn

Employee.java Code:

**package** com.cognizant.spring\_learn.model;

**public** **class** Employee {

**private** **int** id;

**private** String name;

**private** **double** salary;

**private** **boolean** permanent;

**private** Department department;

// Getters and Setters

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **double** getSalary() {

**return** salary;

}

**public** **void** setSalary(**double** salary) {

**this**.salary = salary;

}

**public** **boolean** isPermanent() {

**return** permanent;

}

**public** **void** setPermanent(**boolean** permanent) {

**this**.permanent = permanent;

}

**public** Department getDepartment() {

**return** department;

}

**public** **void** setDepartment(Department department) {

**this**.department = department;

}

}

employee.xml Code:

<?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*

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

<!-- Departments -->

<bean id=*"dept1"* class=*"com.cognizant.spring\_learn.model.Department"*>

<property name=*"id"* value=*"1"* />

<property name=*"name"* value=*"HR"* />

</bean>

<bean id=*"dept2"* class=*"com.cognizant.spring\_learn.model.Department"*>

<property name=*"id"* value=*"2"* />

<property name=*"name"* value=*"Tech"* />

</bean>

<!-- Employees -->

<bean id=*"employeeList"* class=*"java.util.ArrayList"*>

<constructor-arg>

<list>

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

<property name=*"id"* value=*"1"* />

<property name=*"name"* value=*"John"* />

<property name=*"salary"* value=*"50000"* />

<property name=*"permanent"* value=*"true"* />

<property name=*"department"* ref=*"dept1"* />

</bean>

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

<property name=*"id"* value=*"2"* />

<property name=*"name"* value=*"Jane"* />

<property name=*"salary"* value=*"60000"* />

<property name=*"permanent"* value=*"false"* />

<property name=*"department"* ref=*"dept2"* />

</bean>

</list>

</constructor-arg>

</bean>

</beans>

EmployeeService.java Code:

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

**import** java.util.List;

**import** org.springframework.context.ApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**import** com.cognizant.spring\_learn.model.Employee;

**public** **class** EmployeeService {

**private** **static** List<Employee> *employeeList*;

**static** {

ApplicationContext context = **new** ClassPathXmlApplicationContext("employee.xml");

*employeeList* = (List<Employee>) context.getBean("employeeList");

}

**public** List<Employee> getAllEmployees() {

**return** *employeeList*;

}

}

EmployeeController.java Code:

**package** com.cognizant.spring\_learn.controller;

**import** java.util.List;

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

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

**import** com.cognizant.spring\_learn.model.Employee;

**import** com.cognizant.spring\_learn.service.EmployeeService;

@RestController

**public** **class** EmployeeController {

**private** EmployeeService employeeService = **new** EmployeeService();

@GetMapping("/employees")

**public** List<Employee> getAllEmployees() {

**return** employeeService.getAllEmployees();

}

}

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**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 {

**public** **static** **void** main(String[] args) {

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

// Load employee.xml for initializing the employee list

ApplicationContext context = **new** ClassPathXmlApplicationContext("employee.xml");

}

}

application.properties Code:

server.port=8080

spring.application.name=spring-learn

Excepted Output:

[

{

"id": 1,

"name": "John",

"salary": 50000,

"permanent": true,

"department": {

"id": 1,

"name": "HR"

}

},

{

"id": 2,

"name": "Jane",

"salary": 60000,

"permanent": false,

"department": {

"id": 2,

"name": "Tech"

}

}

]

Output:



**Exercise-3: Create REST service to gets all employees**   
Project Name:spring-learn

Employee.java Code:

**package** com.cognizant.spring\_learn.model;

**import** java.util.Date;

**public** **class** Employee {

**private** **int** id;

**private** String name;

**private** **double** salary;

**private** **boolean** permanent;

**private** Date dateOfBirth;

**private** Department department;

// ✅ All-args constructor matching employee.xml

**public** Employee(**int** id, String name, **double** salary, **boolean** permanent, Date dateOfBirth, Department department) {

**this**.id = id;

**this**.name = name;

**this**.salary = salary;

**this**.permanent = permanent;

**this**.dateOfBirth = dateOfBirth;

**this**.department = department;

}

// ✅ No-args constructor (required by Spring for other purposes)

**public** Employee() {}

// ✅ Getters and Setters

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **double** getSalary() {

**return** salary;

}

**public** **void** setSalary(**double** salary) {

**this**.salary = salary;

}

**public** **boolean** isPermanent() {

**return** permanent;

}

**public** **void** setPermanent(**boolean** permanent) {

**this**.permanent = permanent;

}

**public** Date getDateOfBirth() {

**return** dateOfBirth;

}

**public** **void** setDateOfBirth(Date dateOfBirth) {

**this**.dateOfBirth = dateOfBirth;

}

**public** Department getDepartment() {

**return** department;

}

**public** **void** setDepartment(Department department) {

**this**.department = department;

}

@Override

**public** String toString() {

**return** "Employee [id=" + id + ", name=" + name + ", salary=" + salary +

", permanent=" + permanent + ", dateOfBirth=" + dateOfBirth +

", department=" + department + "]";

}

}

Department.java Code:

**package** com.cognizant.spring\_learn.model;

**public** **class** Department {

**private** **int** id;

**private** String name;

**public** Department() {} // default constructor

**public** Department(**int** id, String name) {

**this**.id = id;

**this**.name = name;

}

// getters and setters

}

employee.xml Code:

<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"*>

<!-- Department Bean -->

<bean id=*"dept1"* class=*"com.cognizant.spring\_learn.model.Department"*>

<constructor-arg index=*"0"* value=*"1"*/>

<constructor-arg index=*"1"* value=*"HR"*/>

</bean>

<!-- Employee Bean -->

<bean id=*"employee1"* class=*"com.cognizant.spring\_learn.model.Employee"*>

<constructor-arg index=*"0"* value=*"101"*/>

<constructor-arg index=*"1"* value=*"John"*/>

<constructor-arg index=*"2"* value=*"50000"*/>

<constructor-arg index=*"3"* value=*"true"*/>

<constructor-arg index=*"4"*>

<bean class=*"java.util.Date"*/>

</constructor-arg>

<constructor-arg index=*"5"* ref=*"dept1"*/>

</bean>

</beans>

EmployeeService.java Code:

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

**import** org.springframework.stereotype.Service;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**import** com.cognizant.spring\_learn.model.Employee;

@Service

**public** **class** EmployeeService {

**public** Employee getEmployee() {

ClassPathXmlApplicationContext context = **new** ClassPathXmlApplicationContext("employee.xml");

Employee emp = (Employee) context.getBean("employee1");

context.close();

**return** emp;

}

}

EmployeeController.java Code:

**package** com.cognizant.spring\_learn.controller;

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

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

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

**import** com.cognizant.spring\_learn.model.Employee;

**import** com.cognizant.spring\_learn.service.EmployeeService;

@RestController

**public** **class** EmployeeController {

@Autowired

**private** EmployeeService employeeService;

@GetMapping("/employee")

**public** Employee getEmployee() {

**return** employeeService.getEmployee();

}

}

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication // This will scan all subpackages

**public** **class** SpringLearnApplication {

**public** **static** **void** main(String[] args) {

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

}

}

application.properties Code:

server.port=9090

spring.application.name=spring-learn

Expected Output:

[

{

"id": 1,

"name": "John",

"salary": 50000,

"permanent": true,

"dateOfBirth": "2025-07-12T14:41:16.592+00:00"

},

{

"id": 2,

"name": "Alice",

"salary": 60000,

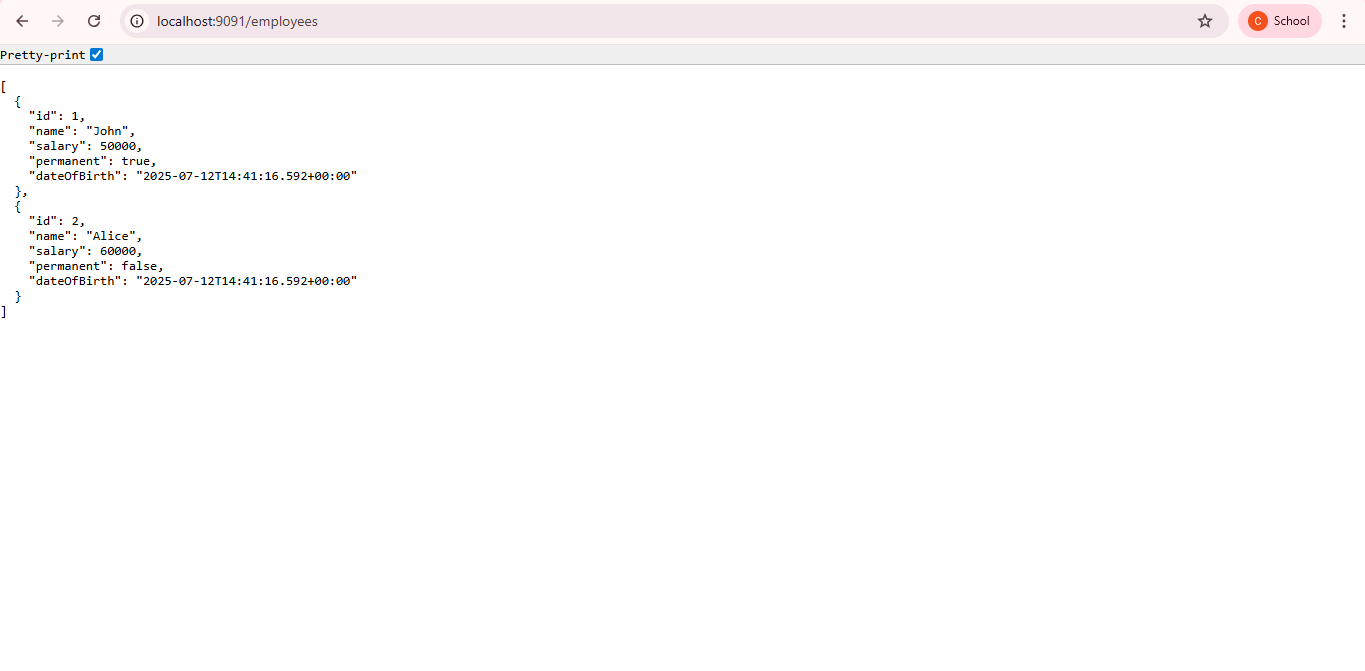
"permanent": false,

"dateOfBirth": "2025-07-12T14:41:16.592+00:00"

}

]

Output:



**Exercise-4: Create REST service for department**

Project Name:spring-learn

Department.java Code:

**package** com.cognizant.spring\_learn.model;

**public** **class** Department {

**private** **int** id;

**private** String name;

// Constructors

**public** Department() {}

**public** Department(**int** id, String name) {

**this**.id = id;

**this**.name = name;

}

// Getters and setters

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

// toString

@Override

**public** String toString() {

**return** "Department [id=" + id + ", name=" + name + "]";

}

}

DepartmentDao.java Code:

**package** com.cognizant.spring\_learn.dao;

**import** com.cognizant.spring\_learn.model.Department;

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

**import** org.springframework.beans.factory.annotation.Qualifier;

**import** org.springframework.stereotype.Repository;

**import** java.util.List;

@Repository

**public** **class** DepartmentDao {

@Autowired

@Qualifier("departmentList")

**private** List<Department> departmentList;

**public** List<Department> getAllDepartments() {

**return** departmentList;

}

}

AppConfig.java Code:

**package** com.cognizant.spring\_learn.config;

**import** org.springframework.context.annotation.Configuration;

**import** org.springframework.context.annotation.ImportResource;

@Configuration

@ImportResource("classpath:spring-learn.xml")

**public** **class** AppConfig {

}

spring-learn.xml Code:

<?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

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

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

<constructor-arg>

<list>

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

<property name="id" value="1"/>

<property name="name" value="Human Resources"/>

</bean>

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

<property name="id" value="2"/>

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

</bean>

</list>

</constructor-arg>

</bean>

</beans>

SpringLearnApplication.java Code:

**package** com.cognizant.spring\_learn;

**import** com.cognizant.spring\_learn.dao.DepartmentDao;

**import** com.cognizant.spring\_learn.model.Department;

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

**import** org.springframework.boot.CommandLineRunner;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

**import** java.util.List;

@SpringBootApplication

**public** **class** SpringLearnApplication **implements** CommandLineRunner {

@Autowired

**private** DepartmentDao departmentDao;

**public** **static** **void** main(String[] args) {

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

}

@Override

**public** **void** run(String... args) **throws** Exception {

List<Department> departments = departmentDao.getAllDepartments();

System.***out***.println("Department List:");

**for** (Department d : departments) {

System.***out***.println(d);

}

}

}

Expected Output:

Department List:

Department [id=1, name=Human Resources]

Department [id=2, name=Finance]

Output:

