**SPRINGREST3**

**PROJECT STRUCTURE**

springrest3/

├── pom.xml

├── src/

│ └── main/

│ ├── java/

│ │ └── com/

│ │ └── example/

│ │ └── springrest3/

│ │ ├── Springrest3Application.java <-- main class

│ │ ├── controller/

│ │ │ ├── EmployeeController.java <-- GET /employees

│ │ │ └── DepartmentController.java <-- GET /departments

│ │ ├── service/

│ │ │ ├── EmployeeService.java

│ │ │ └── DepartmentService.java

│ │ ├── dao/

│ │ │ ├── EmployeeDao.java

│ │ │ └── DepartmentDao.java

│ │ └── model/

│ │ ├── Employee.java

│ │ ├── Department.java

│ │ └── Skill.java

│ └── resources/

│ ├── application.properties <-- optional settings (like server.port)

│ └── employee.xml <-- static config for employee & department list

**DepartmentController.java**

package com.example.springrest3.controller;

import com.example.springrest3.model.Department;

import com.example.springrest3.service.DepartmentService;

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

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

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

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

import java.util.List;

@RestController

@RequestMapping("/departments")

public class DepartmentController {

    @Autowired

    private DepartmentService departmentService;

    @GetMapping

    public List<Department> getAllDepartments() {

        return departmentService.getAllDepartments();

    }

}

**EmployeeController.java**

package com.example.springrest3.controller;

import com.example.springrest3.model.Employee;

import com.example.springrest3.service.EmployeeService;

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

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

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

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

import java.util.List;

@RestController

@RequestMapping("/employees")

public class EmployeeController {

    @Autowired

    private EmployeeService employeeService;

    @GetMapping

    public List<Employee> getAllEmployees() {

        return employeeService.getAllEmployees();

    }

}

**DepartmentDao.java**

package com.example.springrest3.dao;

import com.example.springrest3.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 {

    private static List<Department> DEPARTMENT\_LIST;

    @Autowired

    public DepartmentDao(@Qualifier("departmentList") List<Department> departmentList) {

        DEPARTMENT\_LIST = departmentList;

    }

    public List<Department> getAllDepartments() {

        return DEPARTMENT\_LIST;

    }

}

**EmployeeDao.java**

package com.example.springrest3.dao;

import com.example.springrest3.model.Employee;

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 EmployeeDao {

    private static List<Employee> EMPLOYEE\_LIST;

    @Autowired

    public EmployeeDao(@Qualifier("employeeList") List<Employee> employeeList) {

        EMPLOYEE\_LIST = employeeList;

    }

    public List<Employee> getAllEmployees() {

        return EMPLOYEE\_LIST;

    }

}

**Department.java**

package com.example.springrest3.model;

public class Department {

private int id;

private String name;

// Default constructor

public Department() {

}

// Parameterized constructor

public Department(int id, String name) {

this.id = id;

this.name = name;

}

// Getter and Setter for id

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

// Getter and Setter for name

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

// Optional: toString() for logging/debugging

@Override

public String toString() {

return "Department{id=" + id + ", name='" + name + "'}";

}

}

**Employee.java**

package com.example.springrest3.model;

import java.util.List;

public class Employee {

    private int id;

    private String name;

    private Department department;

    private List<Skill> skills;

    // 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 Department getDepartment() { return department; }

    public void setDepartment(Department department) { this.department = department; }

    public List<Skill> getSkills() { return skills; }

    public void setSkills(List<Skill> skills) { this.skills = skills; }

}

**Skill.java**

package com.example.springrest3.model;

public class Skill {

    private int id;

    private String 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; }

}

**DepartmentService.java**

package com.example.springrest3.service;

import com.example.springrest3.dao.DepartmentDao;

import com.example.springrest3.model.Department;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import java.util.List;

@Service

public class DepartmentService {

    @Autowired

    private DepartmentDao departmentDao;

    @Transactional(readOnly = true)

    public List<Department> getAllDepartments() {

        return departmentDao.getAllDepartments();

    }

}

**EmployeeService.java**

package com.example.springrest3.service;

import com.example.springrest3.dao.EmployeeDao;

import com.example.springrest3.model.Employee;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import java.util.List;

@Service

public class EmployeeService {

    @Autowired

    private EmployeeDao employeeDao;

    @Transactional(readOnly = true)

    public List<Employee> getAllEmployees() {

        return employeeDao.getAllEmployees();

    }

}

**Springrest3Application.java**

package com.example.springrest3;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.annotation.ImportResource;

@SpringBootApplication

@ImportResource("classpath:employee.xml")

public class Springrest3Application {

    public static void main(String[] args) {

        SpringApplication.run(Springrest3Application.class, args);

    }

}

**Employee.xml**

<?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="skill1" class="com.example.springrest3.model.Skill">

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

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

    </bean>

    <bean id="dept1" class="com.example.springrest3.model.Department">

        <property name="id" value="101"/>

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

    </bean>

    <bean id="dept2" class="com.example.springrest3.model.Department">

        <property name="id" value="102"/>

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

    </bean>

    <bean id="emp1" class="com.example.springrest3.model.Employee">

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

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

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

        <property name="skills">

            <list><ref bean="skill1"/></list>

        </property>

    </bean>

    <bean id="emp2" class="com.example.springrest3.model.Employee">

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

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

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

        <property name="skills">

            <list><ref bean="skill1"/></list>

        </property>

    </bean>

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

        <constructor-arg>

            <list>

                <ref bean="emp1"/>

                <ref bean="emp2"/>

            </list>

        </constructor-arg>

    </bean>

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

        <constructor-arg>

            <list>

                <ref bean="dept1"/>

                <ref bean="dept2"/>

            </list>

        </constructor-arg>

    </bean>

</beans>

**Pom.xml**

<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

         http://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>3.2.4</version>

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

    </parent>

    <groupId>com.example</groupId>

    <artifactId>springrest3</artifactId>

    <version>0.0.1-SNAPSHOT</version>

    <name>springrest3</name>

    <description>Spring Boot REST Project using XML Configuration</description>

    <properties>

        <java.version>17</java.version>

    </properties>

    <dependencies>

        <!-- Web Starter (includes Tomcat and JSON support) -->

        <dependency>

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

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

        </dependency>

        <!-- Spring Context for XML-based configuration -->

        <dependency>

            <groupId>org.springframework</groupId>

            <artifactId>spring-context</artifactId>

            <version>6.1.8</version>

        </dependency>

        <!-- Optional: Enable transactional support -->

        <dependency>

            <groupId>org.springframework</groupId>

            <artifactId>spring-tx</artifactId>

        </dependency>

    </dependencies>

    <build>

        <plugins>

            <!-- Spring Boot plugin for packaging and running -->

            <plugin>

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

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

            </plugin>

        </plugins>

    </build>

</project>

**Output:**





