Sprig Rest

1. **Static Employee Data Configuration**

---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

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

<!-- Departments -->

<bean id="hr" class="com.cognizant.model.Department">

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

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

</bean>

<bean id="finance" class="com.cognizant.model.Department">

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

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

</bean>

<!-- Skills -->

<bean id="java" class="com.cognizant.model.Skill">

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

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

</bean>

<bean id="angular" class="com.cognizant.model.Skill">

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

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

</bean>

<!-- Employees -->

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

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

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

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

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

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

<property name="skills">

<list>

<ref bean="java"/>

<ref bean="angular"/>

</list>

</property>

</bean>

<!-- Employee List -->

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

<constructor-arg>

<list>

<ref bean="emp1"/>

<!-- Add more employee references here -->

</list>

</constructor-arg>

</bean>

<!-- Department List -->

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

<constructor-arg>

<list>

<ref bean="hr"/>

<ref bean="finance"/>

</list>

</constructor-arg>

</bean>

</beans>

1. **Employee DAO Implementation**

---EmployeeDao.java

package com.cognizant.dao;

import java.util.ArrayList;

import java.util.List;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Repository;

import com.cognizant.model.Employee;

@Repository

public class EmployeeDao {

private static final List<Employee> EMPLOYEE\_LIST = new ArrayList<Employee>();

public EmployeeDao() {

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

EMPLOYEE\_LIST.addAll((List<Employee>) context.getBean("employeeList"));

}

public List<Employee> getAllEmployees() {

return EMPLOYEE\_LIST;

}

}

1. **Employee Service Implementation**

---EmployeeService.java

package com.cognizant.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import com.cognizant.dao.EmployeeDao;

import com.cognizant.model.Employee;

@Service

public class EmployeeService {

@Autowired

private EmployeeDao employeeDao;

@Transactional

public List<Employee> getAllEmployees() {

return employeeDao.getAllEmployees();

}

}

1. **Employee Controller Implementation**

---EmployeeController.java

package com.cognizant.controller;

import java.util.List;

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

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

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

import com.cognizant.model.Employee;

import com.cognizant.service.EmployeeService;

@RestController

public class EmployeeController {

@Autowired

private EmployeeService employeeService;

@GetMapping("/employees")

public List<Employee> getAllEmployees() {

return employeeService.getAllEmployees();

}

}

1. **Department Implementation**

---DepartementDao.java

package com.cognizant.dao;

import java.util.ArrayList;

import java.util.List;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Repository;

import com.cognizant.model.Department;

@Repository

public class DepartmentDao {

private static final List<Department> DEPARTMENT\_LIST = new ArrayList<Department>();

public DepartmentDao() {

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

DEPARTMENT\_LIST.addAll((List<Department>) context.getBean("departmentList"));

}

public List<Department> getAllDepartments() {

return DEPARTMENT\_LIST;

}

}

---Depatementservice.java

package com.cognizant.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import com.cognizant.dao.DepartmentDao;

import com.cognizant.model.Department;

@Service

public class DepartmentService {

@Autowired

private DepartmentDao departmentDao;

@Transactional

public List<Department> getAllDepartments() {

return departmentDao.getAllDepartments();

}

}

---DepartementController.java

package com.cognizant.controller;

import java.util.List;

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

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

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

import com.cognizant.model.Department;

import com.cognizant.service.DepartmentService;

@RestController

public class DepartmentController {

@Autowired

private DepartmentService departmentService;

@GetMapping("/departments")

public List<Department> getAllDepartments() {

return departmentService.getAllDepartments();

}

}

1. **Model Classes**

---Employee.java

package com.cognizant.model;

import java.util.List;

public class Employee {

private int id;

private String name;

private double salary;

private boolean permanent;

private Department department;

private List<Skill> skills;

// Constructors, getters and setters

}

---Departement.java

package com.cognizant.model;

public class Department {

private int id;

private String name;

// Constructors, getters and setters

}

---Skill.java

package com.cognizant.model;

public class Skill {

private int id;

private String name;

// Constructors, getters and setters

}

**Angular Service Integration**

---employee.services.ts

import { Injectable } from '@angular/core';

import { HttpClient } from '@angular/common/http';

import { Observable } from 'rxjs';

import { Employee } from './employee.model';

@Injectable({

providedIn: 'root'

})

export class EmployeeService {

private apiUrl = 'http://localhost:8080/employees';

constructor(private http: HttpClient) { }

getEmployees(): Observable<Employee[]> {

return this.http.get<Employee[]>(this.apiUrl);

}

getEmployee(id: number): Observable<Employee> {

return this.http.get<Employee>(`${this.apiUrl}/${id}`);

}

}

---departement.services.ts

import { Injectable } from '@angular/core';

import { HttpClient } from '@angular/common/http';

import { Observable } from 'rxjs';

import { Department } from './department.model';

@Injectable({

providedIn: 'root'

})

export class DepartmentService {

private apiUrl = 'http://localhost:8080/departments';

constructor(private http: HttpClient) { }

getDepartments(): Observable<Department[]> {

return this.http.get<Department[]>(this.apiUrl);

}

}