# SPRING REST USING SPRING BOOT

## Problem Statement - Display Employee List and Edit Employee form using RESTful Web Service

employee.xml

<beans [xmlns="http://www.springframework.org/schema/beans"](http://www.springframework.org/schema/beans) [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](http://www.w3.org/2001/XMLSchema-instance)

xsi:schemaLoca[tion="http://www.springframework.org/schema/beans](http://www.springframework.org/schema/beans)

[http://www.springframework.org/schema/beans/spring-beans.xsd">](http://www.springframework.org/schema/beans/spring-beans.xsd)

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

<constructor-arg>

<list>

<bean class="com.cognizant.springlearn.model.Employee">

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

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

<property name="salary" value="45000"/>

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

</bean>

<bean class="com.cognizant.springlearn.model.Employee">

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

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

<property name="salary" value="55000"/>

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

</bean>

</list>

</constructor-arg>

</bean>

</beans>

# Employee.java

package com.cognizant.springlearn.model;

public class Employee { private int id;

private String name; private double salary;

private boolean permanent;

// Getters and Setters

}

EmployeeController.java

package com.cognizant.springlearn.controller;

import com.cognizant.springlearn.model.Employee; import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext; import org.springframework.web.bind.annotation.\*;

import java.util.List;

@RestController

public class EmployeeController {

@GetMapping("/employees")

public List<Employee> getAllEmployees() {

ApplicationContext context = new ClassPathXmlApplicationContext("employee.xml"); return (List<Employee>) context.getBean("employeeList");

}

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

public Employee getEmployeeById(@PathVariable int id) {

ApplicationContext context = new ClassPathXmlApplicationContext("employee.xml"); List<Employee> employees = (List<Employee>) context.getBean("employeeList");

return employees.stream()

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

.findFirst()

.orElse(null);

}

}

# employee.service.ts

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

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

export interface Employee { id: number;

name: string;

salary: number;

permanent: boolean;

}

@Injectable({

providedIn: 'root'

})

export class EmployeeService {

private apiUrl = ['http://localhost:8083/employees';](http://localhost:8083/employees%27%3B)

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}`);

}

}

# employee-list.component.ts

import { Component, OnInit } from '@angular/core';

import { Employee, EmployeeService } from '../employee.service'; import { Router } from '@angular/router';

@Component({

selector: 'app-employee-list',

templateUrl: './employee-list.component.html'

})

export class EmployeeListComponent implements OnInit { employees: Employee[] = [];

constructor(private employeeService: EmployeeService, private router: Router) {}

ngOnInit() {

this.employeeService.getEmployees().subscribe(data => this.employees = data);

}

onEdit(id: number) {

this.router.navigate(['/edit-employee', id]);

}

}

# employee-list.component.html

<table>

<tr \*ngFor="let emp of employees">

<td>{{ emp.id }}</td>

<td>{{ emp.name }}</td>

<td>{{ emp.salary }}</td>

<td>{{ emp.permanent }}</td>

<td><button (click)="onEdit(emp.id)">Edit</button></td>

</tr>

</table>

edit-employee.component.ts

import { Component, OnInit } from '@angular/core'; import { ActivatedRoute } from '@angular/router';

import { Employee, EmployeeService } from '../employee.service';

@Component({

selector: 'app-edit-employee',

templateUrl: './edit-employee.component.html'

})

export class EditEmployeeComponent implements OnInit { employee!: Employee;

constructor(private route: ActivatedRoute, private employeeService: EmployeeService) {}

ngOnInit() {

const id = +this.route.snapshot.paramMap.get('id')!;

this.employeeService.getEmployee(id).subscribe(data => this.employee = data);

}

}

# OUTPUT

[

{

"id": 1,

"name": "John Doe", "salary": 45000,

"permanent": true

},

{

"id": 2,

"name": "Jane Smith", "salary": 55000,

"permanent": false

}

]

## Create static employee list data using spring xml conflguration

employee.xml

<!-- src/main/resources/employee.xml -->

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

<beans [xmlns="http://www.springframework.org/schema/beans"](http://www.springframework.org/schema/beans) [xmlns:util="http://www.springframework.org/schema/util](http://www.springframework.org/schema/util)"

[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](http://www.w3.org/2001/XMLSchema-instance)

xsi:schemaLoca[tion="http://www.springframework.org/schema/beans](http://www.springframework.org/schema/beans)

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

[http://www.springframework.org/schema/util/spring-util.xsd">](http://www.springframework.org/schema/util/spring-util.xsd)

<!-- Departments -->

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

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

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

</bean>

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

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

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

</bean>

<!-- Skills -->

<bean id="skillJava" class="com.cognizant.spring\_learn.model.Skill">

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

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

</bean>

<bean id="skillSQL" class="com.cognizant.spring\_learn.model.Skill">

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

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

</bean>

<!-- Employees -->

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

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

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

<property name="salary" value="45000"/>

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

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

<property name="skills">

<list>

<ref bean="skillJava"/>

<ref bean="skillSQL"/>

</list>

</property>

</bean>

<bean id="emp2" class="com.cognizant.spring\_learn.model.Employee">

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

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

<property name="salary" value="55000"/>

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

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

<property name="skills">

<list>

<ref bean="skillSQL"/>

</list>

</property>

</bean>

<bean id="emp3" class="com.cognizant.spring\_learn.model.Employee">

<property name="id" value="3"/>

<property name="name" value="Alex Brown"/>

<property name="salary" value="62000"/>

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

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

<property name="skills">

<list>

<ref bean="skillJava"/>

</list>

</property>

</bean>

<bean id="emp4" class="com.cognizant.spring\_learn.model.Employee">

<property name="id" value="4"/>

<property name="name" value="Emma Watson"/>

<property name="salary" value="72000"/>

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

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

<property name="skills">

<list>

<ref bean="skillJava"/>

<ref bean="skillSQL"/>

</list>

</property>

</bean>

<!-- Employee List -->

<util:list id="employeeList" value-type="com.cognizant.spring\_learn.model.Employee">

<ref bean="emp1"/>

<ref bean="emp2"/>

<ref bean="emp3"/>

<ref bean="emp4"/>

</util:list>

</beans>

# EmployeeDao.java

package com.cognizant.spring\_learn.dao;

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

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

import org.springframework.context.support.ClassPathXmlApplicationContext; import org.springframework.stereotype.Repository;

import java.util.List;

@Repository

public class EmployeeDao {

private static List<Employee> EMPLOYEE\_LIST;

public EmployeeDao() {

ApplicationContext context = new ClassPathXmlApplicationContext("employee.xml"); EMPLOYEE\_LIST = (List<Employee>) context.getBean("employeeList");

}

public List<Employee> getAllEmployees() { return EMPLOYEE\_LIST;

}

}

# Employee.java

package com.cognizant.spring\_learn.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;

// Getters and Setters

// toString()

}

OUTPUT

Employee [id=1, name=John Doe, salary=45000.0, permanent=true, department=Human Resources, skills=[Java, SQL]]

Employee [id=2, name=Jane Smith, salary=55000.0, permanent=false, department=Finance, skills=[SQL]]

Employee [id=3, name=Alex Brown, salary=62000.0, permanent=true, department=Human Resources, skills=[Java]]

Employee [id=4, name=Emma Watson, salary=72000.0, permanent=true, department=Finance, skills=[Java, SQL]]

## Create REST service to gets all employees

EmployeeService.java

package com.cognizant.springlearn.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.springlearn.dao.EmployeeDao; import com.cognizant.springlearn.model.Employee;

@Service

public class EmployeeService {

@Autowired

private EmployeeDao employeeDao;

@Transactional

public List<Employee> getAllEmployees() { return employeeDao.getAllEmployees();

}

}

# EmployeeController.java

package com.cognizant.springlearn.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.springlearn.model.Employee;

import com.cognizant.springlearn.service.EmployeeService;

@RestController

public class EmployeeController {

@Autowired

private EmployeeService employeeService;

@GetMapping("/employees")

public List<Employee> getAllEmployees() {

return employeeService.getAllEmployees();

}

}

# output

[

{

"id": 1,

"name": "John Doe", "salary": 45000,

"permanent": true, "department": { "id": 1,

"name": "Human Resources"

},

"skills": [

{ "id": 1, "name": "Java" },

{ "id": 2, "name": "SQL" }

]

},

{

"id": 2,

"name": "Jane Smith", "salary": 55000,

"permanent": false,

"department": { "id": 2,

"name": "Finance"

},

"skills": [

{ "id": 3, "name": "Accounting" },

{ "id": 4, "name": "Excel" }

]

},

{

"id": 3,

"name": "Alice Johnson", "salary": 60000,

"permanent": true, "department": { "id": 3,

"name": "IT"

},

"skills": [

{ "id": 1, "name": "Java" },

{ "id": 5, "name": "Angular" }

]

},

{

"id": 4,

"name": "Bob Williams", "salary": 47000,

"permanent": true,

"department": { "id": 1,

"name": "Human Resources"

},

"skills": [

{ "id": 2, "name": "SQL" },

{ "id": 6, "name": "Communication" }

]

},

{

"id": 5,

"name": "Carol Brown", "salary": 52000,

"permanent": false, "department": { "id": 2,

"name": "Finance"

},

"skills": [

{ "id": 3, "name": "Accounting" },

{ "id": 7, "name": "Leadership" }

]

},

{

"id": 6,

"name": "David Lee", "salary": 63000,

"permanent": true,

"department": { "id": 3,

"name": "IT"

},

"skills": [

{ "id": 1, "name": "Java" },

{ "id": 5, "name": "Angular" }

]

}

]

## Create REST service for department

department.xml

<beans [xmlns="http://www.springframework.org/schema/beans"](http://www.springframework.org/schema/beans)

...>

<bean id="department1" class="com.cognizant.spring\_learn.model.Department">

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

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

</bean>

<bean id="department2" class="com.cognizant.spring\_learn.model.Department">

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

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

</bean>

<bean id="department3" class="com.cognizant.spring\_learn.model.Department">

<property name="id" value="3"/>

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

</bean>

<util:list id="departmentList" value-type="com.cognizant.spring\_learn.model.Department">

<ref bean="department1"/>

<ref bean="department2"/>

<ref bean="department3"/>

</util:list>

</beans>

# DepartmentDao.java

package com.cognizant.spring\_learn.dao;

import java.util.ArrayList; import java.util.List;

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

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

@Component

public class DepartmentDao {

private static List<Department> DEPARTMENT\_LIST;

@Autowired

public DepartmentDao(List<Department> departmentList) { DEPARTMENT\_LIST = new ArrayList<>(departmentList);

}

public List<Department> getAllDepartments() { return DEPARTMENT\_LIST;

}

}

# DepartmentService.java

package com.cognizant.spring\_learn.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.spring\_learn.dao.DepartmentDao; import com.cognizant.spring\_learn.model.Department;

@Service

public class DepartmentService {

@Autowired

private DepartmentDao departmentDao;

@Transactional

public List<Department> getAllDepartments() { return departmentDao.getAllDepartments();

}

}

# DepartmentController.java

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

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

@RestController

public class DepartmentController {

@Autowired

private DepartmentService departmentService;

@GetMapping("/departments")

public List<Department> getAllDepartments() {

System.out.println("DepartmentController: getAllDepartments called"); return departmentService.getAllDepartments();

}

}

# output

[

{

"id": 1,

"name": "Human Resources"

},

{

"id": 2,

"name": "Finance"

},

{

"id": 3,

"name": "IT"

}

]