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

In the previous angular module, we developed a screen that lists employees and it was populated with hard coded values. Now this angular application has be changed to get the data from RESTful Web Service developed in Spring. The following are the high level activities that needs to be done to accomplish this:

- **Create static employee list data using spring xml configuration**

```
<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="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 Doe" />
             <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 Smith" />
            <property name="salary" value="60000" />
            <property name="permanent" value="false" />
        </bean>
    </list>
</constructor-arg>
</bean>

</beans>
```

- **Create a REST Service that reads data from xml configuration and returns it**

  Create REST Service to Return Employee List

  Employee.java

```
package com.cognizant.spring_learn.model;

public class Employee {
    private int id;
    private String name;
    private double salary;
    private boolean permanent;
```

```
    // Getters and Setters
}
```

EmployeeDao.java

```
package com.cognizant.spring_learn.dao;

import com.cognizant.spring_learn.model.Employee;
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;

    static {
        ClassPathXmlApplicationContext context = new
ClassPathXmlApplicationContext("employee.xml");
        EMPLOYEE_LIST = context.getBean("employeeList", List.class);
        context.close();
```

```
    }

    public List<Employee> getAllEmployees() {
        return EMPLOYEE_LIST;
    }

    public Employee getEmployeeById(int id) {
        return EMPLOYEE_LIST.stream().filter(e -> e.getId() ==
id).findFirst().orElse(null);
    }
}
```

EmployeeService.java

```
package com.cognizant.spring_learn.service;

import com.cognizant.spring_learn.dao.EmployeeDao;
import com.cognizant.spring_learn.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
    public List<Employee> getAllEmployees() {
        return employeeDao.getAllEmployees();
    }

    @Transactional
    public Employee getEmployee(int id) {
        return employeeDao.getEmployeeById(id);
    }
}
```

EmployeeController.java

```
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 getEmployee(@PathVariable int id) {
        return employeeService.getEmployee(id);
    }
}
```

- **Make changes in angular component to consume the created REST Service**

this.http.get<Employee[]>('http://localhost:8083/employees')

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

Once above activities are completed, clicking on the Edit button against each employee should display Edit Employee form with values retrieved from RESTful Web Service. This will also involve activities similar to the one specified above.

NOTE: There is no specific activity as part of this hands on, refer the next hands ons that covers above three activities in detail.

OUTPUT:


```
"id": 1,
"name": "John Doe",
"salary": 50000,
"permanent": true
```

Create static employee list data using spring xml configuration

Follow steps below to accomplish this activity:

- **Incorporate the following in employee.xml:**
  - **Create one or two more departments**
  - **Create four more instances of Employee. (use employee sample data from angular)**
  - **Reuse existing skills instead of creating new ones**
  - **Include all four employee instances in an ArrayList.**

employee.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">

    <!-- 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="Technology"/>
</bean>

<!-- Skills -->
<bean id="skill1" class="com.cognizant.spring_learn.model.Skill">
    <property name="id" value="1"/>
    <property name="name" value="Java"/>
</bean>

<bean id="skill2" class="com.cognizant.spring_learn.model.Skill">
    <property name="id" value="2"/>
    <property name="name" value="Angular"/>
</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 Doe"/>
    <property name="salary" value="60000"/>
    <property name="permanent" value="true"/>
    <property name="department" ref="dept1"/>
    <property name="skills">
        <list>
            <ref bean="skill1"/>
            <ref bean="skill2"/>
        </list>
    </property>
</bean>
<bean 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="skill1"/>
        </list>
    </property>
```

```
</bean>
<bean class="com.cognizant.spring_learn.model.Employee">
    <property name="id" value="3"/>
    <property name="name" value="Alice Brown"/>
    <property name="salary" value="52000"/>
    <property name="permanent" value="true"/>
    <property name="department" ref="dept1"/>
    <property name="skills">
        <list>
            <ref bean="skill2"/>
        </list>
    </property>
</bean>
<bean class="com.cognizant.spring_learn.model.Employee">
    <property name="id" value="4"/>
    <property name="name" value="Bob Williams"/>
    <property name="salary" value="70000"/>
    <property name="permanent" value="false"/>
    <property name="department" ref="dept2"/>
    <property name="skills">
        <list>
            <ref bean="skill1"/>
```

```
                    <ref bean="skill2"/>
                </list>
            </property>
        </bean>
    </list>
</constructor-arg>
</bean>
</beans>
```

- **In EmployeeDao, incorporate the following:**
  - **Create static variable with name EMPLOYEE_LIST of type ArrayList<Employee>**
  - **Include constructor that reads employee list from xml config and set the EMPLOYEE_LIST**
  - **Create method getAllEmployees() that returns the EMPLOYEE_LIST**
  -

EmployeeDao.java

```
package com.cognizant.spring_learn.dao;

import com.cognizant.spring_learn.model.Employee;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import org.springframework.stereotype.Repository;

import java.util.ArrayList;
```

```
import java.util.List;


@Repository
public class EmployeeDao {


    private static List<Employee> EMPLOYEE_LIST;


      public EmployeeDao() {
        ClassPathXmlApplicationContext context = new
ClassPathXmlApplicationContext("employee.xml");
        EMPLOYEE_LIST = context.getBean("employeeList", ArrayList.class);
        context.close();
    }


    public List<Employee> getAllEmployees() {
        return EMPLOYEE_LIST;
    }
}
```

EmployeeDao.java

```
package com.cognizant.spring_learn.dao;


import com.cognizant.spring_learn.model.Employee;
```

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

import org.springframework.stereotype.Repository;

import java.util.ArrayList;

import java.util.List;


@Repository

public class EmployeeDao {


    private static List<Employee> EMPLOYEE_LIST;


    public EmployeeDao() {

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

        EMPLOYEE_LIST = context.getBean("employeeList", ArrayList.class);

        context.close();

    }


    public List<Employee> getAllEmployees() {

        return EMPLOYEE_LIST;

    }

}
```

Create REST service to gets all employees

Follow steps below to accomplish this activity:

- **In EmployeeService, incorporate the following:**
  - **Change the annotation for this class from @Component to @Service**
  - **Create method getAllEmployees() that invokes employeeDao.getAllEmployees() and return the employee list**
  - **Define @Transactional annotation for this method.**

```
 package com.cognizant.spring_learn.service;

import com.cognizant.spring_learn.dao.EmployeeDao;
import com.cognizant.spring_learn.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  // Changed from @Component to @Service
public class EmployeeService {

    @Autowired
    private EmployeeDao employeeDao;

    @Transactional
    public List<Employee> getAllEmployees() {
        return employeeDao.getAllEmployees();
    }
}
```

- **In EmployeeController, incorporate the following:**
  - **Include a new get method with name getAllEmployees() that returns the employee list**
  - **Mark this method as GetMapping annotation with the URL as '/employees'**
  - **Within this method invoke employeeService.getAllEmployees() and return the same**.

```
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();
    }
}
```

- **Test the service using postman.**
  - **URL:** http://localhost:8083/employees

  OUTPUT:


```
[
  {
    "id": 1,
    "name": "John Doe",
    "salary": 60000,
    "permanent": true,
    "department": {
      "id": 1,
      "name": "Human Resources"
    },
    "skills": [
      { "id": 1, "name": "Java" },
      { "id": 2, "name": "Angular" }
    ]
  },
```

Create REST service for department

Create a new service to get all the departments.

Follow steps below to achieve this:

- **Create a new REST Service, define below list of classes and respective methods:**
  - **DepartmentController**
    - **getAllDepartments() with URL "/departments", this method will return array of departments**
  - **DepartmentService**
    - **getAllDepartments()**
  - **DepartmentDao**
    - **getAllDepartments() - Create a static variable DEPARTMENT_LIST, this should be populated from spring xml configuration**

department.xml – Spring XML Config

```
<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="Technology"/>
                </bean>
                <bean class="com.cognizant.spring_learn.model.Department">
                    <property name="id" value="3"/>
                    <property name="name" value="Finance"/>
```

```
            </bean>
        </list>
    </constructor-arg>
</bean>

</beans>
```

Department.java

```
package com.cognizant.spring_learn.model;

public class Department {
    private int id;
    private String name;

    // Getters and Setters
}
```

DepartmentDao.java

```
package com.cognizant.spring_learn.dao;

import com.cognizant.spring_learn.model.Department;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import org.springframework.stereotype.Repository;

import java.util.List;

@Repository
public class DepartmentDao {

    private static List<Department> DEPARTMENT_LIST;

    static {
        ClassPathXmlApplicationContext context = new
ClassPathXmlApplicationContext("department.xml");
        DEPARTMENT_LIST = context.getBean("departmentList", List.class);
        context.close();
    }

    public List<Department> getAllDepartments() {
        return DEPARTMENT_LIST;
```

```
    }
}
```

DepartmentService.java

```
package com.cognizant.spring_learn.service;

import com.cognizant.spring_learn.dao.DepartmentDao;
import com.cognizant.spring_learn.model.Department;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.List;

@Service
public class DepartmentService {

    @Autowired
    private DepartmentDao departmentDao;

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

DepartmentController.java

```
package com.cognizant.spring_learn.controller;

import com.cognizant.spring_learn.model.Department;
import com.cognizant.spring_learn.service.DepartmentService;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

import java.util.List;

@RestController
public class DepartmentController {

    private static final Logger LOGGER =
LoggerFactory.getLogger(DepartmentController.class);
```

```
    @Autowired
    private DepartmentService departmentService;

    @GetMapping("/departments")
    public List<Department> getAllDepartments() {
        LOGGER.info("Start getAllDepartments");
        List<Department> list = departmentService.getAllDepartments();
        LOGGER.info("End getAllDepartments");
        return list;
    }
}
```

- **Test the service using postman.**

```
[
  { "id": 1, "name": "Human Resources" },
  { "id": 2, "name": "Technology" },
  { "id": 3, "name": "Finance" }
]
```

- **Also verify if department REST service is called by looking into the logs.**

```
INFO  com.cognizant.spring_learn.controller.DepartmentController - Start getAllDepartments
INFO  com.cognizant.spring_learn.controller.DepartmentController - End getAllDepartments
```

SUPERSET ID: 6407636

KANMANI MURUGHAIYAN