1. **Display Employee List and Edit Employee form using RESTful Web Service**

Employee.java

package com.res.emploee;

import lombok.Data;

*@Data*

public class Employee {

private int id;

private String name;

private double salary;

private boolean permanent;

private String department;

}

EmployeeController.java

package com.res.emploee;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

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

import java.util.List;

*@RestController*

*@RequestMapping*("/employees")

*@CrossOrigin* // allow Angular to call this

public class EmployeeController {

private final ApplicationContext context;

// Inject it via constructor

public EmployeeController() {

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

}

*@GetMapping*

public List<Employee> getAllEmployees() {

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

}

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

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

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

return employees.stream()

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

.findFirst()

.orElseThrow(() -> new RuntimeException("Employee not found"));

}

}

DemoApplication.java

package com.res.emploee;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@*SpringBootApplication

public class DemoApplication {

public static void main(String[] args) {

SpringApplication.run(DemoApplication.class, args);

}

}

employee.xml

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

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:util="http://www.springframework.org/schema/util"

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

http://www.springframework.org/schema/util

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

<util:list id="employeeList">

<bean class="com.res.emploee.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" value="HR"/>

</bean>

<bean class="com.res.emploee.Employee">

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

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

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

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

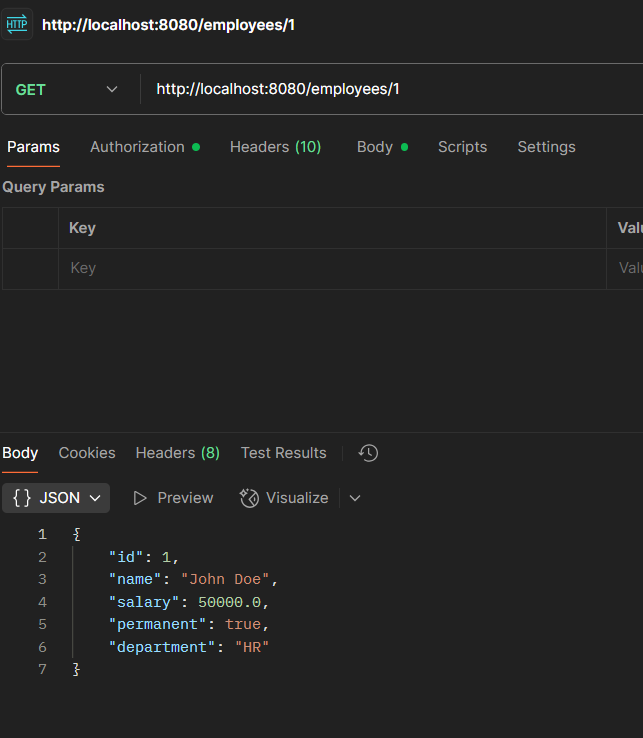
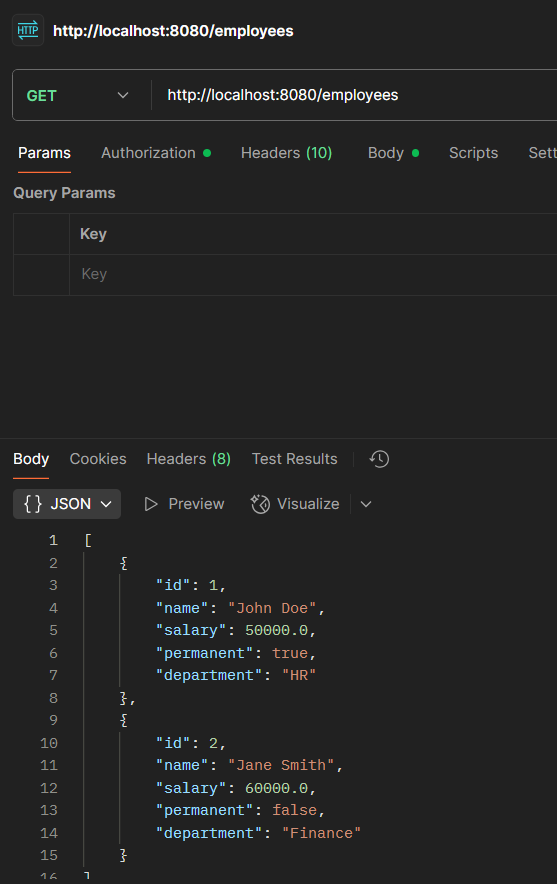
<property name="department" value="Finance"/>

</bean>

</util:list>

</beans>

**Output**

1. **Create static employee list data using spring xml configuration**

employee.xml

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

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:util="http://www.springframework.org/schema/util"

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

http://www.springframework.org/schema/util

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

<!-- Departments -->

<bean id="hrDept" class="com.res.emploee.Department">

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

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

</bean>

<bean id="financeDept" class="com.res.emploee.Department">

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

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

</bean>

<bean id="itDept" class="com.res.emploee.Department">

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

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

</bean>

<!-- Skills -->

<bean id="skillJava" class="com.res.emploee.Skill">

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

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

</bean>

<bean id="skillAngular" class="com.res.emploee.Skill">

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

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

</bean>

<!-- Employee List -->

<util:list id="employeeList">

<bean class="com.res.emploee.Employee">

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

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

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

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

<property name="dateOfBirth" value="1985-06-20"/>

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

<property name="skills">

<list>

<ref bean="skillJava"/>

</list>

</property>

</bean>

<bean class="com.res.emploee.Employee">

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

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

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

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

<property name="dateOfBirth" value="1990-08-15"/>

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

<property name="skills">

<list>

<ref bean="skillAngular"/>

</list>

</property>

</bean>

<bean class="com.res.emploee.Employee">

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

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

<property name="salary" value="70000"/>

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

<property name="dateOfBirth" value="1988-11-10"/>

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

<property name="skills">

<list>

<ref bean="skillJava"/>

<ref bean="skillAngular"/>

</list>

</property>

</bean>

<bean class="com.res.emploee.Employee">

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

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

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

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

<property name="dateOfBirth" value="1992-03-05"/>

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

<property name="skills">

<list>

<ref bean="skillAngular"/>

</list>

</property>

</bean>

</util:list>

</beans>

Employee.java

package com.res.emploee;

import java.util.List;

import lombok.Data;

*@Data*

public class Employee {

private int id;

private String name;

private double salary;

private boolean permanent;

private String dateOfBirth;

private Department department;

private List<Skill> skills;

}

Department.java

package com.res.emploee;

import lombok.Data;

*@Data*

public class Department {

private int id;

private String name;

}

Skill.java

package com.res.emploee;

import lombok.Data;

*@Data*

public class Skill {

private int id;

private String name;

}

EmployeeDao.java

package com.res.emploee;

import org.springframework.context.ApplicationContext;

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 ArrayList<Employee> *EMPLOYEE\_LIST*;

public EmployeeDao() {

try {

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

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

*EMPLOYEE\_LIST* = new ArrayList<>(employees);

} catch (Exception e) {

e.printStackTrace();

throw e; // rethrow to see the cause in logs

}

}

public ArrayList<Employee> getAllEmployees() {

return *EMPLOYEE\_LIST*;

}

public Employee getEmployeeById(int id) {

return *EMPLOYEE\_LIST*.stream()

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

.findFirst()

.orElseThrow(() -> new RuntimeException("Employee not found"));

}

}

EmployeeController.java

package com.res.emploee;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

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

import java.util.List;

*@RestController*

*@RequestMapping*("/employees")

*@CrossOrigin* // allow Angular to call this

public class EmployeeController {

private final EmployeeDao employeeDao;

// Inject it via constructor

public EmployeeController() {

this.employeeDao = new EmployeeDao();

}

*@GetMapping*

public List<Employee> getAllEmployees() {

return employeeDao.getAllEmployees();

}

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

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

return employeeDao.getEmployeeById(id);

}

}

DemoApplication.java

package com.res.emploee;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class DemoApplication {

public static void main(String[] args) {

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

}

}

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

<modelVersion>4.0.0</modelVersion>

<groupId>com.res</groupId>

<artifactId>emploee</artifactId>

<version>0.0.1-SNAPSHOT</version>

<parent>

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

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

<version>3.5.3</version>

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

</parent>

<properties>

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

</properties>

<dependencies>

<!-- Spring Boot Web Starter (REST APIs and embedded server) -->

<dependency>

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

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

</dependency>

<!-- Lombok for easy getters/setters (optional) -->

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<scope>provided</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

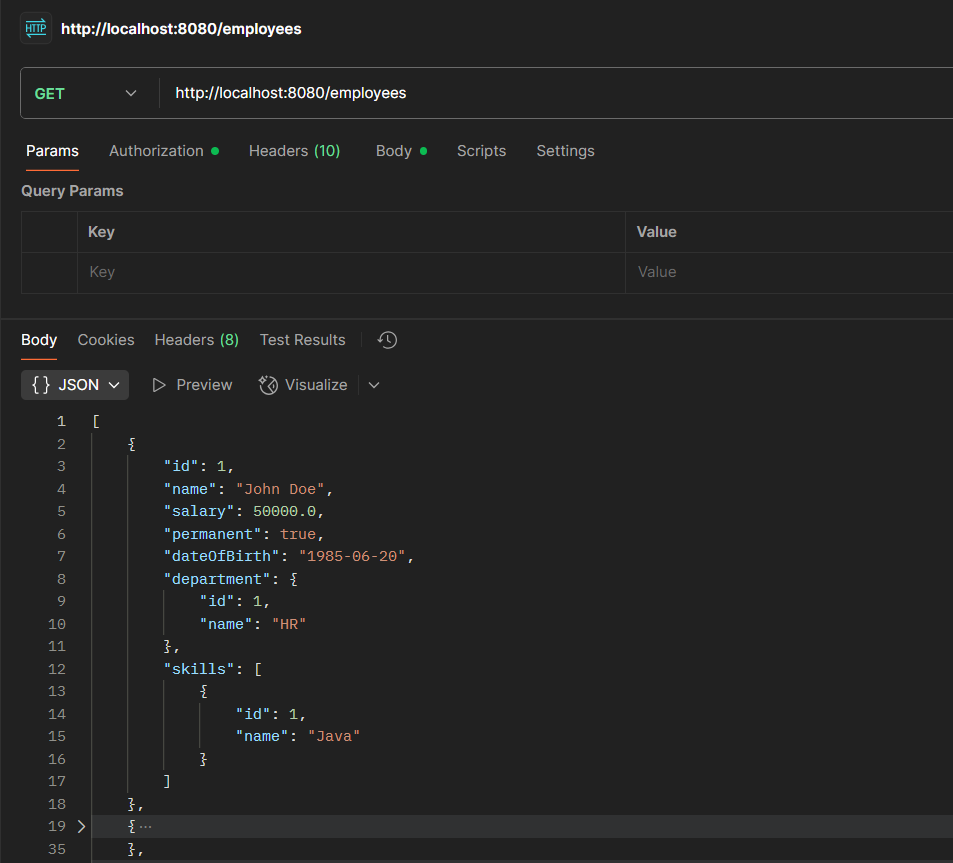
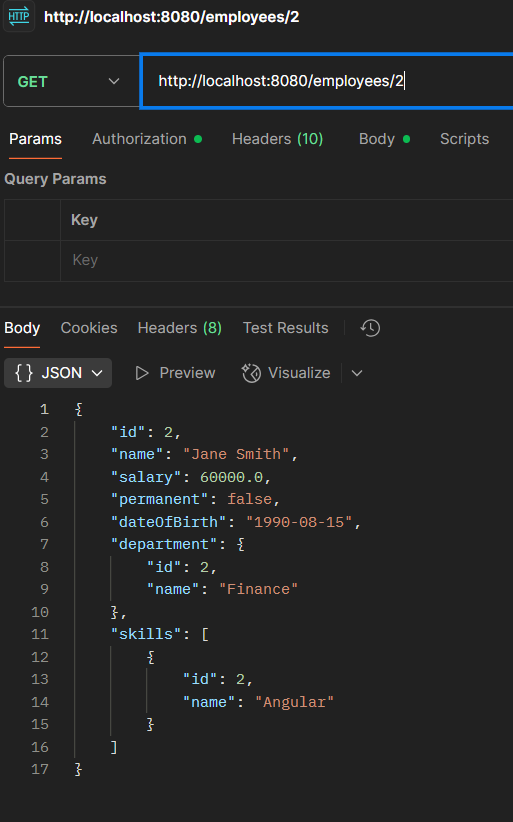
</plugin>

</plugins>

</build>

</project>

**Output**

1. **Create REST service to gets all employees**

EmployeeController.java

package com.res.emploee;

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

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

import java.util.List;

*@RestController*

*@RequestMapping*("/employees")

*@CrossOrigin*

public class EmployeeController {

*@Autowired*

private final EmployeeService employeeService;

*@GetMapping*

public List<Employee> getAllEmployees() {

return employeeService.getAllEmployees();

}

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

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

return employeeService.getEmployeeById(id);

}

}

EmployeeService.java

package com.res.emploee;

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

import org.springframework.stereotype.Service;

import java.util.List;

*@Service*

public class EmployeeService {

*@Autowired*

private final EmployeeDao employeeDao;

public List<Employee> getAllEmployees() {

return employeeDao.getAllEmployees();

}

public Employee getEmployeeById(int id) {

return employeeDao.getEmployeeById(id);

}

}

EmployeeDao.java

package com.res.emploee;

import org.springframework.context.ApplicationContext;

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 ArrayList<Employee> *EMPLOYEE\_LIST*;

public EmployeeDao() {

try {

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

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

*EMPLOYEE\_LIST* = new ArrayList<>(employees);

} catch (Exception e) {

e.printStackTrace();

throw e; // rethrow to see the cause in logs

}

}

public ArrayList<Employee> getAllEmployees() {

return *EMPLOYEE\_LIST*;

}

public Employee getEmployeeById(int id) {

return *EMPLOYEE\_LIST*.stream()

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

.findFirst()

.orElseThrow(() -> new RuntimeException("Employee not found"));

}

}

DemoApplication.java

package com.res.emploee;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class DemoApplication {

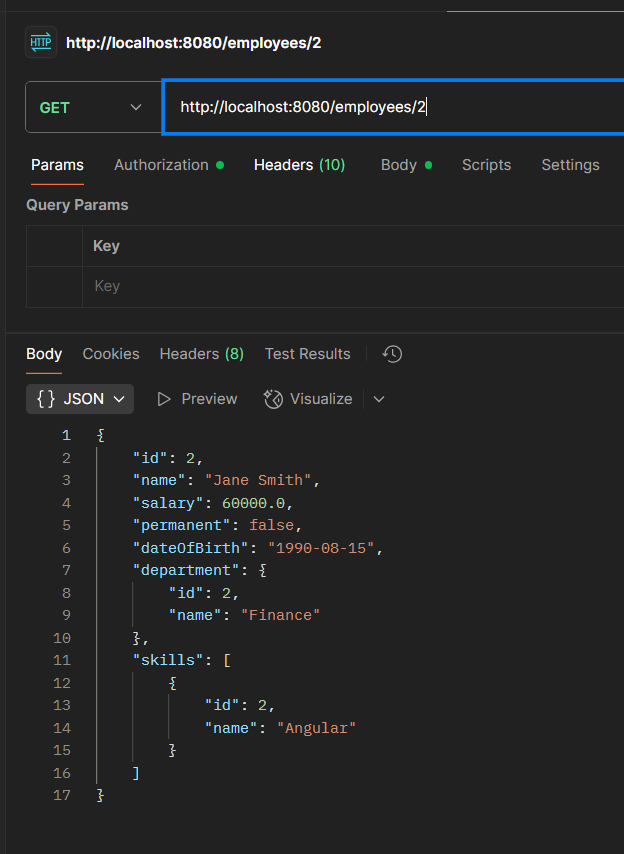
public static void main(String[] args) {

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

}

}

**Output**



1. **Create REST service for department**

employee.xml

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

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:util="http://www.springframework.org/schema/util"

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

http://www.springframework.org/schema/util

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

<!-- Departments -->

<bean id="hrDept" class="com.res.emploee.Department">

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

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

</bean>

<bean id="financeDept" class="com.res.emploee.Department">

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

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

</bean>

<bean id="itDept" class="com.res.emploee.Department">

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

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

</bean>

<util:list id="departmentList">

<ref bean="hrDept"/>

<ref bean="financeDept"/>

<ref bean="itDept"/>

</util:list>

<!-- Skills -->

<bean id="skillJava" class="com.res.emploee.Skill">

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

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

</bean>

<bean id="skillAngular" class="com.res.emploee.Skill">

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

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

</bean>

<util:list id="skillList">

<ref bean="skillJava"/>

<ref bean="skillAngular"/>

</util:list>

<!-- Employee List -->

<util:list id="employeeList">

<bean class="com.res.emploee.Employee">

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

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

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

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

<property name="dateOfBirth" value="1985-06-20"/>

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

<property name="skills">

<list>

<ref bean="skillJava"/>

</list>

</property>

</bean>

<bean class="com.res.emploee.Employee">

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

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

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

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

<property name="dateOfBirth" value="1990-08-15"/>

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

<property name="skills">

<list>

<ref bean="skillAngular"/>

</list>

</property>

</bean>

<bean class="com.res.emploee.Employee">

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

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

<property name="salary" value="70000"/>

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

<property name="dateOfBirth" value="1988-11-10"/>

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

<property name="skills">

<list>

<ref bean="skillJava"/>

<ref bean="skillAngular"/>

</list>

</property>

</bean>

<bean class="com.res.emploee.Employee">

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

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

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

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

<property name="dateOfBirth" value="1992-03-05"/>

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

<property name="skills">

<list>

<ref bean="skillAngular"/>

</list>

</property>

</bean>

</util:list>

</beans>

DepartmentController.java

package com.res.emploee;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import java.util.List;

*@RestController*

*@RequestMapping*("/departments")

*@CrossOrigin*

public class DepartmentController {

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

private final DepartmentService departmentService;

public DepartmentController(DepartmentService departmentService) {

this.departmentService = departmentService;

}

*@GetMapping*

public List<Department> getAllDepartments() {

***LOGGER***.debug("Department REST service called: getAllDepartments()");

return departmentService.getAllDepartments();

}

}

DepartmentService.java

package com.res.emploee;

import org.springframework.stereotype.Service;

import java.util.List;

*@Service*

public class DepartmentService {

private final DepartmentDao departmentDao;

public DepartmentService(DepartmentDao departmentDao) {

this.departmentDao = departmentDao;

}

public List<Department> getAllDepartments() {

return departmentDao.getAllDepartments();

}

}

DepartmentDao.java

package com.res.emploee;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Repository;

import java.util.ArrayList;

import java.util.List;

*@Repository*

public class DepartmentDao {

public static ArrayList<Department> *DEPARTMENT\_LIST*;

public DepartmentDao() {

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

List<Department> departments = (List<Department>) context.getBean("departmentList");

*DEPARTMENT\_LIST* = new ArrayList<>(departments);

}

public ArrayList<Department> getAllDepartments() {

return *DEPARTMENT\_LIST*;

}

}

**Output**

