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

SpringLearnApplication.java :-

package com.cognizant.spring\_learn;  
  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
  
@SpringBootApplication  
public class SpringLearnApplication {  
  
 public static void main(String[] args) {  
 SpringApplication.*run*(SpringLearnApplication.class, args);  
 }  
  
}

**EmployeeController.java :-**

package com.cognizant.spring\_learn.controller;  
  
import com.cognizant.spring\_learn.dao.EmployeeDao;  
import com.cognizant.spring\_learn.model.Employee;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.web.bind.annotation.\*;  
  
import java.util.List;  
  
@RestController  
@RequestMapping("/employees")  
public class EmployeeController {  
  
 @Autowired  
 private EmployeeDao employeeDao;  
  
 @GetMapping  
 public List<Employee> getAllEmployees() {  
 return employeeDao.getAllEmployees();  
 }  
  
 @GetMapping("/{id}")  
 public Employee getEmployeeById(@PathVariable int id) {  
 return employeeDao.getEmployeeById(id);  
 }  
}

**EmployeeDao.java :-**

package com.cognizant.spring\_learn.dao;  
  
import com.cognizant.spring\_learn.model.Employee;  
import org.springframework.beans.propertyeditors.CustomDateEditor;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.AbstractApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
import org.springframework.stereotype.Repository;  
  
import java.text.SimpleDateFormat;  
import java.util.List;  
  
@Repository  
public class EmployeeDao {  
  
 private static List<Employee> *EMPLOYEE\_LIST*;  
  
 public EmployeeDao() {  
 try {  
 ApplicationContext context = new ClassPathXmlApplicationContext("employee.xml");  
 *EMPLOYEE\_LIST* = (List<Employee>) context.getBean("employeeList");  
 } catch (Exception e) {  
 throw new RuntimeException("Failed to load employee list from XML", e);  
 }  
 }  
  
  
 public List<Employee> getAllEmployees() {  
 return *EMPLOYEE\_LIST*;  
 }  
  
 public Employee getEmployeeById(int id) {  
 for (Employee emp : *EMPLOYEE\_LIST*) {  
 if (emp.getId() == id) {  
 return emp;  
 }  
 }  
 return null;  
 }  
}

**Department.java :-**

package com.cognizant.spring\_learn.model;  
  
public class Department {  
 private long id;  
 private String name;  
  
 // Required no-arg constructor  
 public Department() {}  
  
 // Getters and Setters  
 public long getId() {  
 return id;  
 }  
  
 public void setId(long id) {  
 this.id = id;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
}

**Employee.java :-**

package com.cognizant.spring\_learn.model;  
  
import com.cognizant.spring\_learn.model.Department;  
import com.cognizant.spring\_learn.model.Skill;  
  
import java.util.Date;  
import java.util.List;  
  
public class Employee {  
 private long id;  
 private String name;  
 private double salary;  
 private boolean permanent;  
 private Department department;  
 private Date dateOfBirth;  
 private List<Skill> skillList;  
  
 public Employee() {}  
  
 public long getId() { return id; }  
 public void setId(long id) { this.id = id; }  
  
 public String getName() { return name; }  
 public void setName(String name) { this.name = name; }  
  
 public double getSalary() { return salary; }  
 public void setSalary(double salary) { this.salary = salary; }  
  
 public boolean isPermanent() { return permanent; }  
 public void setPermanent(boolean permanent) { this.permanent = permanent; }  
  
 public Department getDepartment() { return department; }  
 public void setDepartment(Department department) { this.department = department; }  
  
 public Date getDateOfBirth() { return dateOfBirth; }  
 public void setDateOfBirth(Date dateOfBirth) { this.dateOfBirth = dateOfBirth; }  
  
 public List<Skill> getSkillList() { return skillList; }  
 public void setSkillList(List<Skill> skillList) { this.skillList = skillList; }  
}

**Skill.java :-**

package com.cognizant.spring\_learn.model;  
  
public class Skill {  
 private int id;  
 private String name;  
  
 // Getter  
 public int getId() {  
 return id;  
 }  
  
 // Setter  
 public void setId(int id) {  
 this.id = id;  
 }  
  
 // Getter  
 public String getName() {  
 return name;  
 }  
  
 // Setter  
 public void setName(String name) {  
 this.name = name;  
 }  
}

**EmployeeService.java :-**

package com.cognizant.spring\_learn.service;  
  
import com.cognizant.spring\_learn.model.Employee;  
import org.springframework.beans.factory.annotation.Value;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
import org.springframework.stereotype.Service;  
  
import java.util.List;  
  
@Service  
public class EmployeeService {  
 ApplicationContext context = new ClassPathXmlApplicationContext("employee.xml");  
  
 public List<Employee> getAllEmployees() {  
 return (List<Employee>) context.getBean("employeeList");  
 }  
  
 public Employee getEmployeeById(int id) {  
 List<Employee> employees = getAllEmployees();  
 return employees.stream()  
 .filter(emp -> emp.getId() == id)  
 .findFirst()  
 .orElse(null);  
 }  
}

**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">  
  
 <!-- Departments -->  
 <bean id="dept1" class="com.cognizant.spring\_learn.model.Department">  
 <property name="id" value="1"/>  
 <property name="name" value="HR"/>  
 </bean>  
  
 <bean id="dept2" class="com.cognizant.spring\_learn.model.Department">  
 <property name="id" value="2"/>  
 <property name="name" value="IT"/>  
 </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="Spring Boot"/>  
 </bean>  
  
 <!-- Employee List -->  
 <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"/>  
 <property name="salary" value="50000"/>  
 <property name="permanent" value="true"/>  
 <property name="department" ref="dept1"/>  
 <property name="dateOfBirth" value="01/01/1990"/>  
 <property name="skillList">  
 <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"/>  
 <property name="salary" value="60000"/>  
 <property name="permanent" value="false"/>  
 <property name="department" ref="dept2"/>  
 <property name="dateOfBirth" value="15/03/1992"/>  
 <property name="skillList">  
 <list>  
 <ref bean="skill1"/>  
 </list>  
 </property>  
 </bean>  
 </list>  
 </constructor-arg>  
 </bean>  
  
</beans>

**Pom.xml :-**

<?xml version="1.0" encoding="UTF-8"?>  
<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>  
 <parent>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-parent</artifactId>  
 <version>3.5.3</version>  
 <relativePath/> <!-- lookup parent from repository -->  
 </parent>  
 <groupId>com.cognizant</groupId>  
 <artifactId>spring\_learn</artifactId>  
 <version>0.0.1-SNAPSHOT</version>  
 <name>spring\_learn</name>  
 <description>Demo project for Spring Boot</description>  
 <url/>  
 <licenses>  
 <license/>  
 </licenses>  
 <developers>  
 <developer/>  
 </developers>  
 <scm>  
 <connection/>  
 <developerConnection/>  
 <tag/>  
 <url/>  
 </scm>  
 <properties>  
 <java.version>21</java.version>  
 </properties>  
 <dependencies>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>test</scope>  
 </dependency>  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 </dependency>  
  
 <!-- Spring Context for XML Configuration -->  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-context</artifactId>  
 </dependency>  
  
 <!-- For Hot Reload (Optional) -->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-devtools</artifactId>  
 <scope>runtime</scope>  
 <optional>true</optional>  
 </dependency>  
  
 <!-- Test Support -->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
  
 <build>  
 <plugins>  
 <plugin>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-maven-plugin</artifactId>  
 </plugin>  
 </plugins>  
 </build>  
  
</project>

Output :-





