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

package com.cognizant.spring\_learn.model;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class Employee {

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

private int id;

private String name;

private double salary;

private boolean permanent;

private String department;

public Employee() {

*LOGGER*.debug("Inside Employee Constructor");

}

public int getId() {

*LOGGER*.debug("getId() called");

return id;

}

public void setId(int id) {

*LOGGER*.debug("setId() called");

this.id = id;

}

public String getName() {

*LOGGER*.debug("getName() called");

return name;

}

public void setName(String name) {

*LOGGER*.debug("setName() called");

this.name = name;

}

public double getSalary() {

*LOGGER*.debug("getSalary() called");

return salary;

}

public void setSalary(double salary) {

*LOGGER*.debug("setSalary() called");

this.salary = salary;

}

public boolean isPermanent() {

*LOGGER*.debug("isPermanent() called");

return permanent;

}

public void setPermanent(boolean permanent) {

*LOGGER*.debug("setPermanent() called");

this.permanent = permanent;

}

public String getDepartment() {

*LOGGER*.debug("getDepartment() called");

return department;

}

public void setDepartment(String department) {

*LOGGER*.debug("setDepartment() called");

this.department = department;

}

*@Override*

public String toString() {

return "Employee [id=" + id + ", name=" + name + ", salary=" + salary +

", permanent=" + permanent + ", department=" + department + "]";

}

}

**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">

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

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

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

<property name="salary" value="40000"/>

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

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

</bean>

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

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

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

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

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

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

</bean>

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

<constructor-arg>

<list>

<ref bean="emp1"/>

<ref bean="emp2"/>

</list>

</constructor-arg>

</bean>

</beans>

**EmployeeController.java**

package com.cognizant.spring\_learn.controller;

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

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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 {

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

*@GetMapping*("/employees")

public List<Employee> getAllEmployees() {

*LOGGER*.info("START - getAllEmployees()");

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

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

*LOGGER*.debug("Employees: {}", employees);

*LOGGER*.info("END - getAllEmployees()");

return employees;

}

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

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

*LOGGER*.info("START - getEmployeeById()");

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

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

return employees.stream()

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

.findFirst()

.orElse(null);

}

}

**SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import java.util.List;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

@SpringBootApplication(scanBasePackages = "com.cognizant.spring\_learn")

public class SpringLearnApplication {

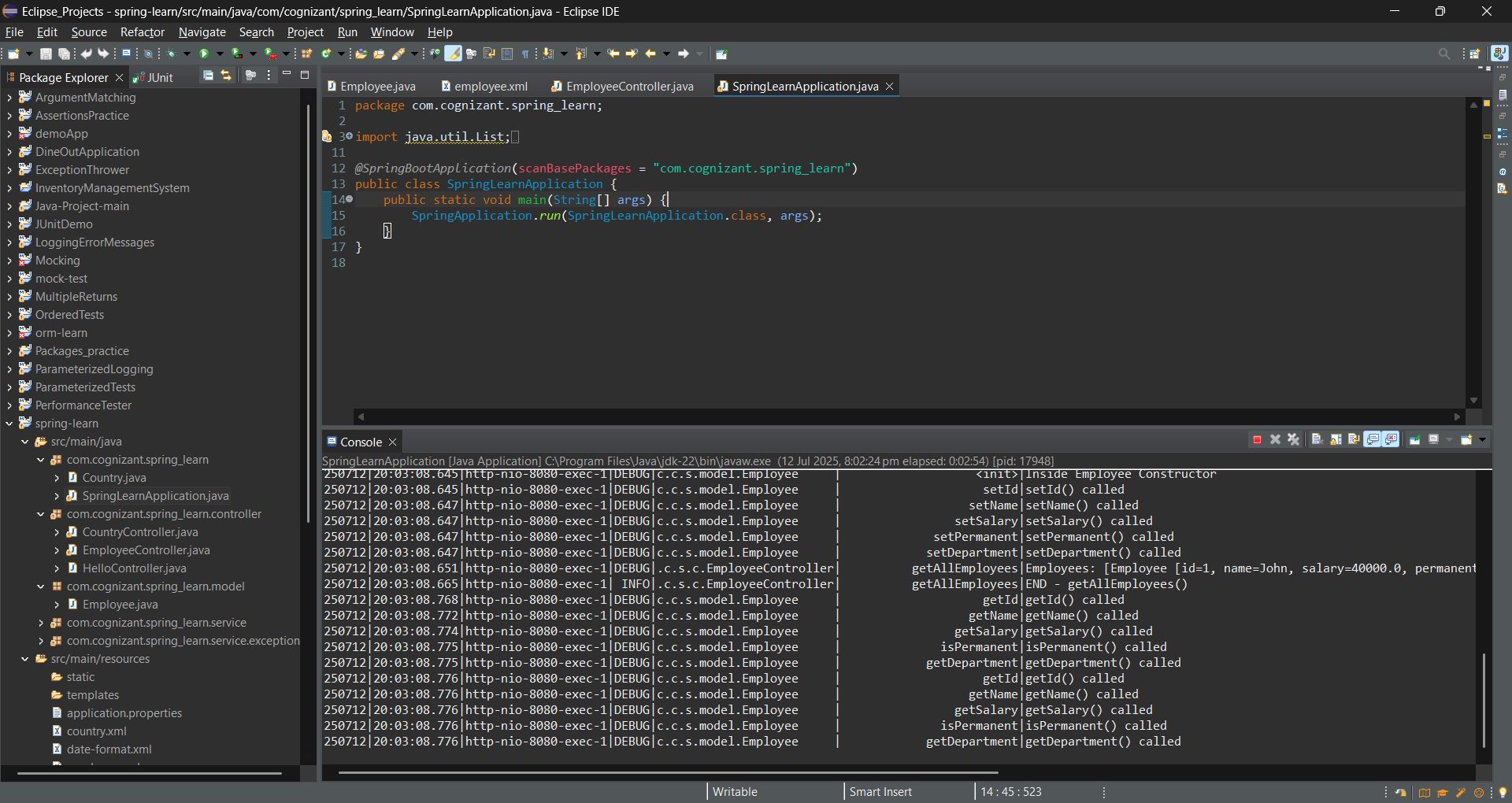
public static void main(String[] args) {

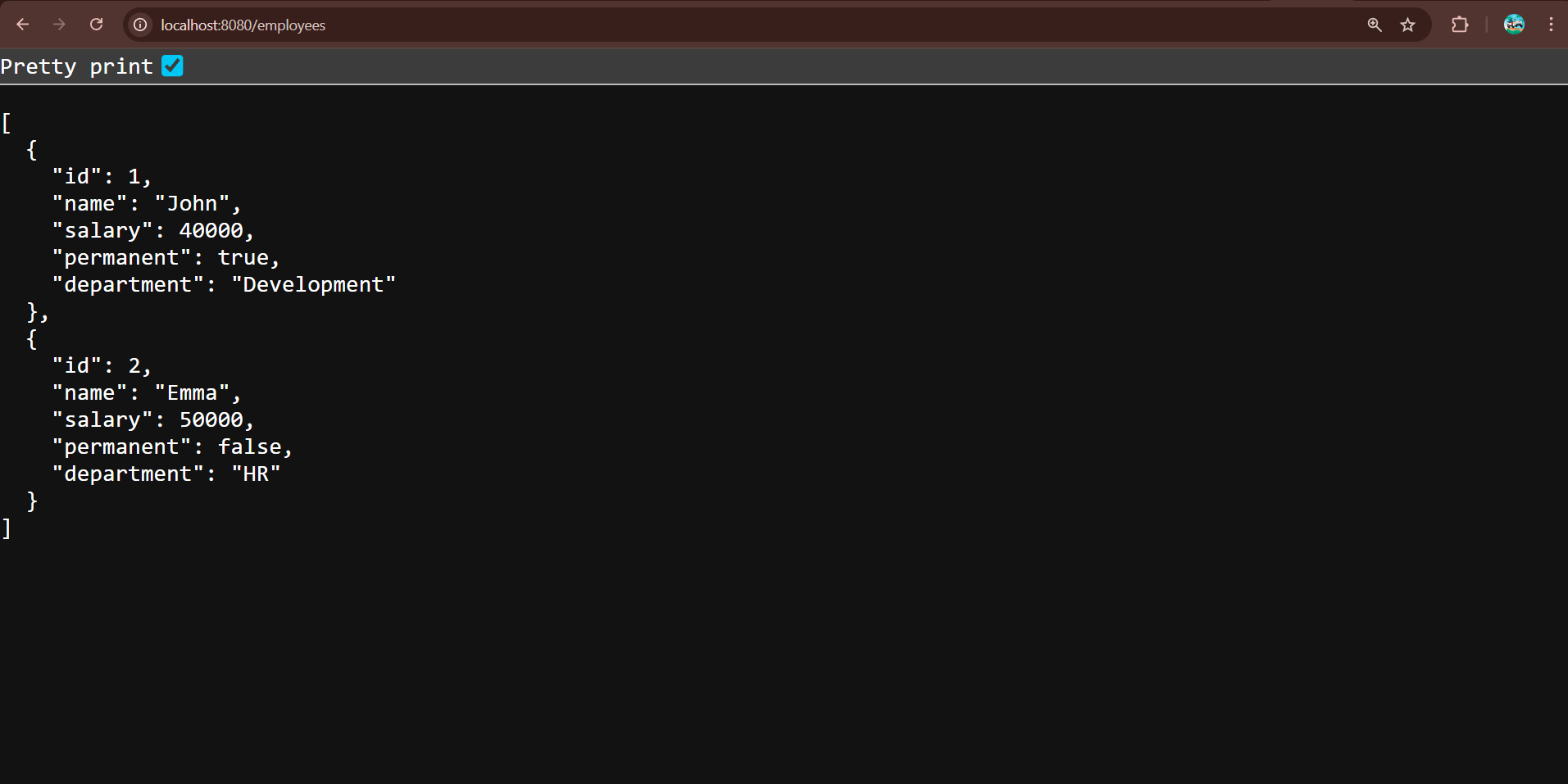
SpringApplication.run(SpringLearnApplication.class, args);

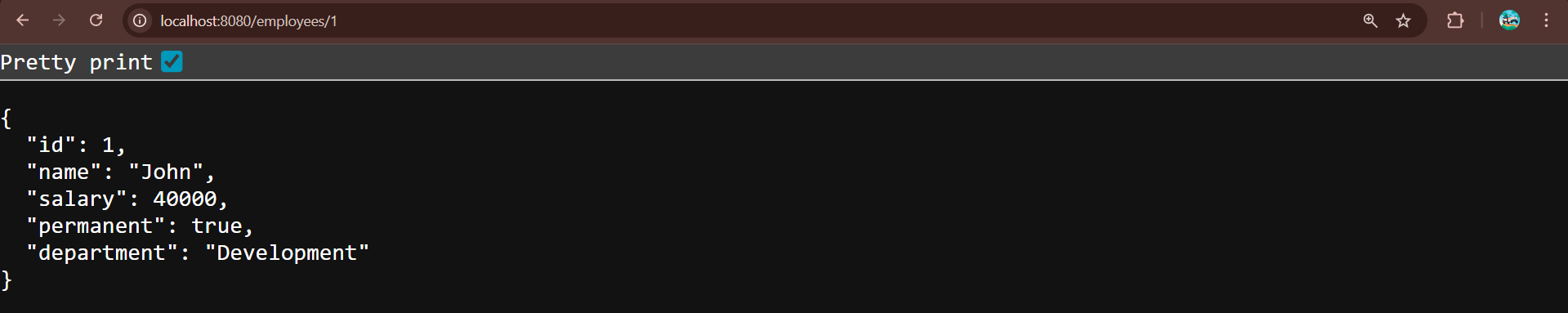
}

}

**Output:**







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

**Employee.java**

package com.cognizant.spring\_learn.model;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class Employee {

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

private int id;

private String name;

private double salary;

private boolean permanent;

private String department;

public Employee() {

LOGGER.debug("Inside Employee Constructor");

}

public int getId() {

LOGGER.debug("getId() called");

return id;

}

public void setId(int id) {

LOGGER.debug("setId() called");

this.id = id;

}

public String getName() {

LOGGER.debug("getName() called");

return name;

}

public void setName(String name) {

LOGGER.debug("setName() called");

this.name = name;

}

public double getSalary() {

LOGGER.debug("getSalary() called");

return salary;

}

public void setSalary(double salary) {

LOGGER.debug("setSalary() called");

this.salary = salary;

}

public boolean isPermanent() {

LOGGER.debug("isPermanent() called");

return permanent;

}

public void setPermanent(boolean permanent) {

LOGGER.debug("setPermanent() called");

this.permanent = permanent;

}

public String getDepartment() {

LOGGER.debug("getDepartment() called");

return department;

}

public void setDepartment(String department) {

LOGGER.debug("setDepartment() called");

this.department = department;

}

@Override

public String toString() {

return "Employee [id=" + id + ", name=" + name + ", salary=" + salary +

", permanent=" + permanent + ", department=" + department + "]";

}

}

**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="development" class="java.lang.String">

<constructor-arg value="Development"/>

</bean>

<bean id="hr" class="java.lang.String">

<constructor-arg value="HR"/>

</bean>

<!-- Skills -->

<bean id="java" class="java.lang.String">

<constructor-arg value="Java"/>

</bean>

<bean id="angular" class="java.lang.String">

<constructor-arg value="Angular"/>

</bean>

<!-- Employees -->

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

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

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

<property name="salary" value="40000"/>

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

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

</bean>

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

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

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

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

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

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

</bean>

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

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

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

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

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

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

</bean>

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

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

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

<property name="salary" value="48000"/>

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

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

</bean>

<bean id="EMPLOYEE\_LIST" class="java.util.ArrayList">

<constructor-arg>

<list>

<ref bean="emp1"/>

<ref bean="emp2"/>

<ref bean="emp3"/>

<ref bean="emp4"/>

</list>

</constructor-arg>

</bean>

</beans>

**EmployeeController.java**

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.dao.EmployeeDao;

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

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import java.util.List;

@RestController

public class EmployeeController {

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

private final EmployeeDao employeeDao = new EmployeeDao();

@GetMapping("/employees")

public List<Employee> getAllEmployees() {

LOGGER.info("START - getAllEmployees()");

List<Employee> employees = employeeDao.getAllEmployees();

LOGGER.debug("Employees: {}", employees);

LOGGER.info("END - getAllEmployees()");

return employees;

}

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

public Employee getEmployeeById(@PathVariable int id) {

LOGGER.info("START - getEmployeeById()");

return employeeDao.getAllEmployees()

.stream()

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

.findFirst()

.orElse(null);

}

}

**EmployeeDao.java**

package com.cognizant.spring\_learn.dao;

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

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import java.util.ArrayList;

import java.util.List;

public class EmployeeDao {

private static ArrayList<Employee> EMPLOYEE\_LIST;

public EmployeeDao() {

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

EMPLOYEE\_LIST = (ArrayList<Employee>) context.getBean("EMPLOYEE\_LIST");

}

public List<Employee> getAllEmployees() {

return EMPLOYEE\_LIST;

}

}

**SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import java.util.List;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

@SpringBootApplication(scanBasePackages = "com.cognizant.spring\_learn")

public class SpringLearnApplication {

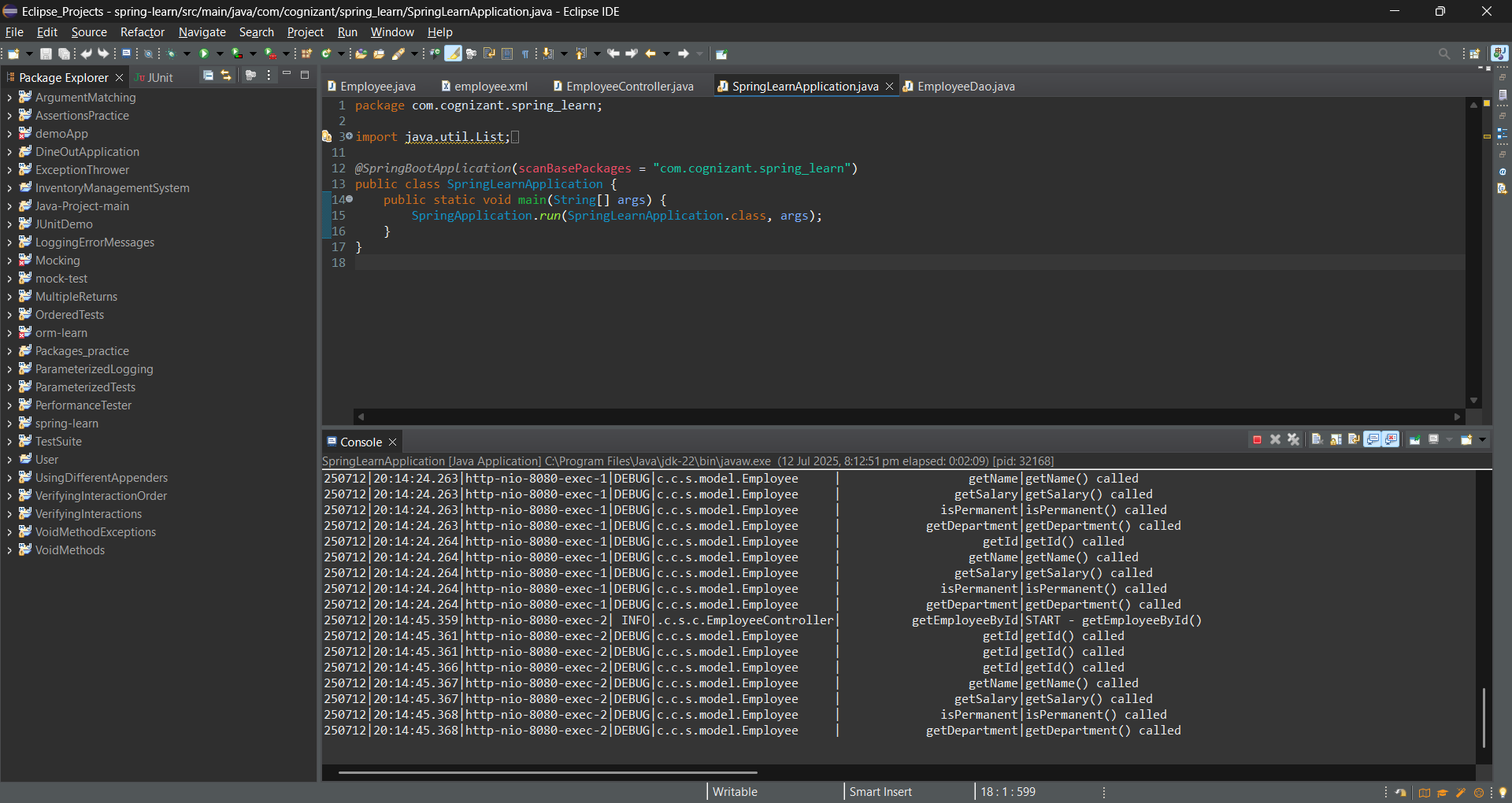
public static void main(String[] args) {

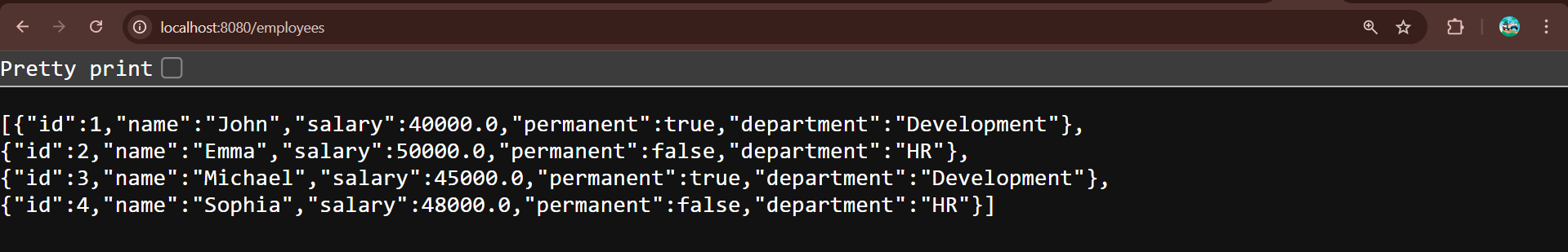
SpringApplication.run(SpringLearnApplication.class, args);

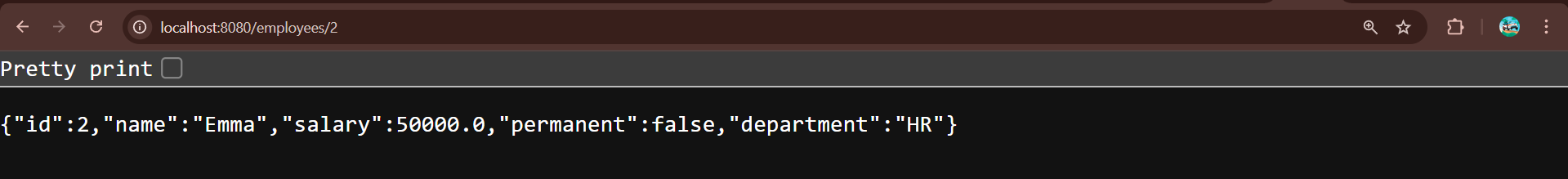
}

}

**Output:**







**Create REST service to gets all employees**

**Employee.java**

package com.cognizant.spring\_learn.model;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class Employee {

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

private int id;

private String name;

private double salary;

private boolean permanent;

private String department;

public Employee() {

LOGGER.debug("Inside Employee Constructor");

}

public int getId() {

LOGGER.debug("getId() called");

return id;

}

public void setId(int id) {

LOGGER.debug("setId() called");

this.id = id;

}

public String getName() {

LOGGER.debug("getName() called");

return name;

}

public void setName(String name) {

LOGGER.debug("setName() called");

this.name = name;

}

public double getSalary() {

LOGGER.debug("getSalary() called");

return salary;

}

public void setSalary(double salary) {

LOGGER.debug("setSalary() called");

this.salary = salary;

}

public boolean isPermanent() {

LOGGER.debug("isPermanent() called");

return permanent;

}

public void setPermanent(boolean permanent) {

LOGGER.debug("setPermanent() called");

this.permanent = permanent;

}

public String getDepartment() {

LOGGER.debug("getDepartment() called");

return department;

}

public void setDepartment(String department) {

LOGGER.debug("setDepartment() called");

this.department = department;

}

@Override

public String toString() {

return "Employee [id=" + id + ", name=" + name + ", salary=" + salary +

", permanent=" + permanent + ", department=" + department + "]";

}

}

**EmployeeService.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.EmployeeDao;

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

@Service

public class EmployeeService {

@Autowired

private EmployeeDao employeeDao;

@Transactional

public List<Employee> getAllEmployees() {

return employeeDao.getAllEmployees();

}

}

**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="development" class="java.lang.String">

<constructor-arg value="Development"/>

</bean>

<bean id="hr" class="java.lang.String">

<constructor-arg value="HR"/>

</bean>

<!-- Skills -->

<bean id="java" class="java.lang.String">

<constructor-arg value="Java"/>

</bean>

<bean id="angular" class="java.lang.String">

<constructor-arg value="Angular"/>

</bean>

<!-- Employees -->

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

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

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

<property name="salary" value="40000"/>

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

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

</bean>

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

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

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

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

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

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

</bean>

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

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

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

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

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

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

</bean>

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

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

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

<property name="salary" value="48000"/>

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

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

</bean>

<bean id="EMPLOYEE\_LIST" class="java.util.ArrayList">

<constructor-arg>

<list>

<ref bean="emp1"/>

<ref bean="emp2"/>

<ref bean="emp3"/>

<ref bean="emp4"/>

</list>

</constructor-arg>

</bean>

</beans>

**EmployeeController.java**

package com.cognizant.spring\_learn.controller;

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

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

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 EmployeeController {

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

@Autowired

private EmployeeService employeeService;

// ✅ GET /employees

@GetMapping("/employees")

public List<Employee> getAllEmployees() {

LOGGER.info("START getAllEmployees");

List<Employee> employees = employeeService.getAllEmployees();

LOGGER.debug("Employee List: {}", employees);

LOGGER.info("END getAllEmployees");

return employees;

}

}

**EmployeeDao.java**

package com.cognizant.spring\_learn.dao;

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

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() {

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

EMPLOYEE\_LIST = (ArrayList<Employee>) context.getBean("EMPLOYEE\_LIST");

}

public List<Employee> getAllEmployees() {

return EMPLOYEE\_LIST;

}

}

**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>

<properties>

<java.version>21</java.version>

</properties>

<dependencies>

<!-- Spring Boot Starter for Web -->

<dependency>

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

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

</dependency>

<!-- Spring Boot DevTools for live reload -->

<dependency>

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

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<!-- Spring Boot Starter Test (JUnit + MockMvc) -->

<dependency>

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

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

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-tx</artifactId>

</dependency>

</dependencies>

<build>

<plugins>

<!-- Spring Boot Maven Plugin -->

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>

**SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

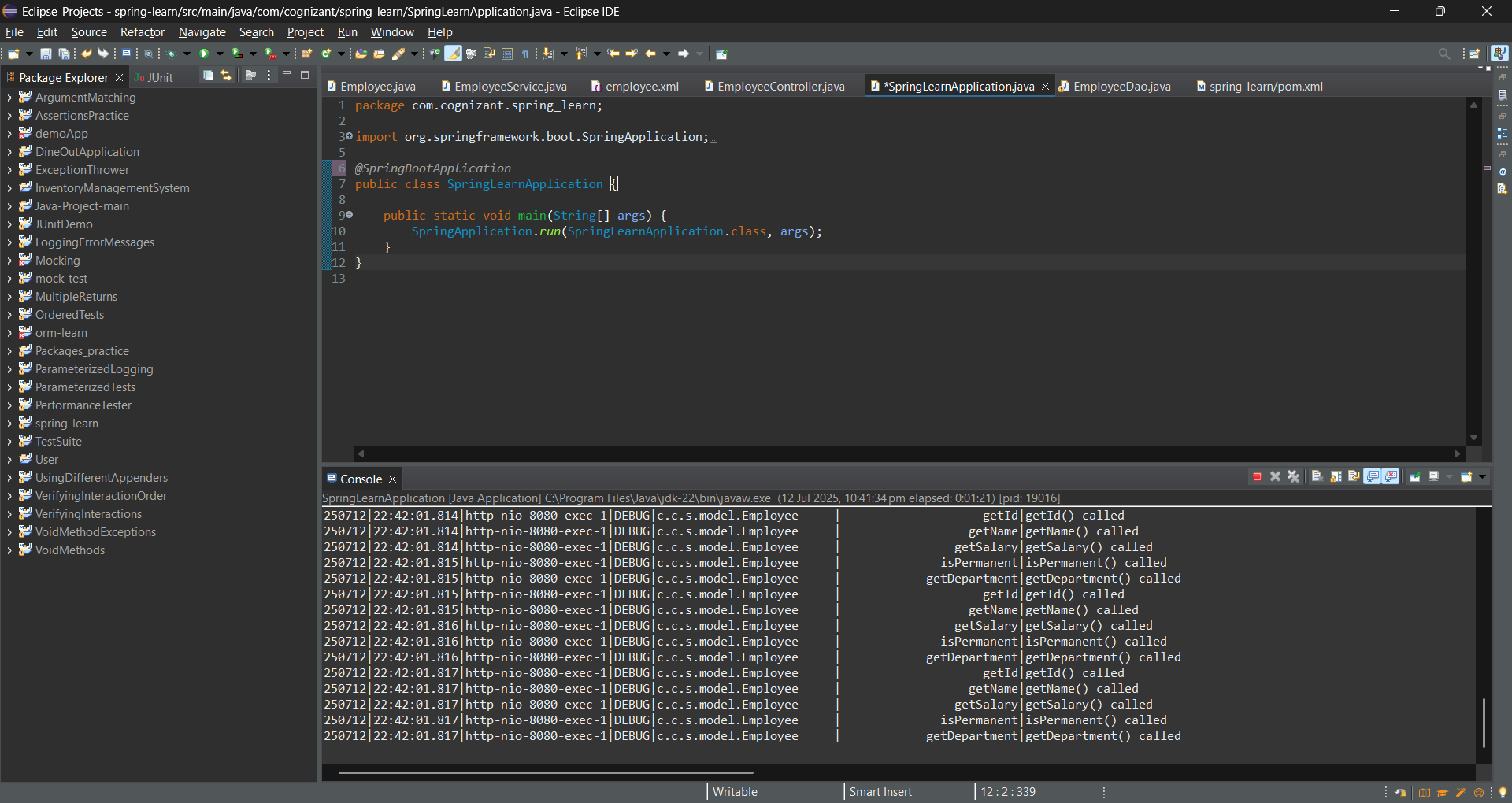
public class SpringLearnApplication {

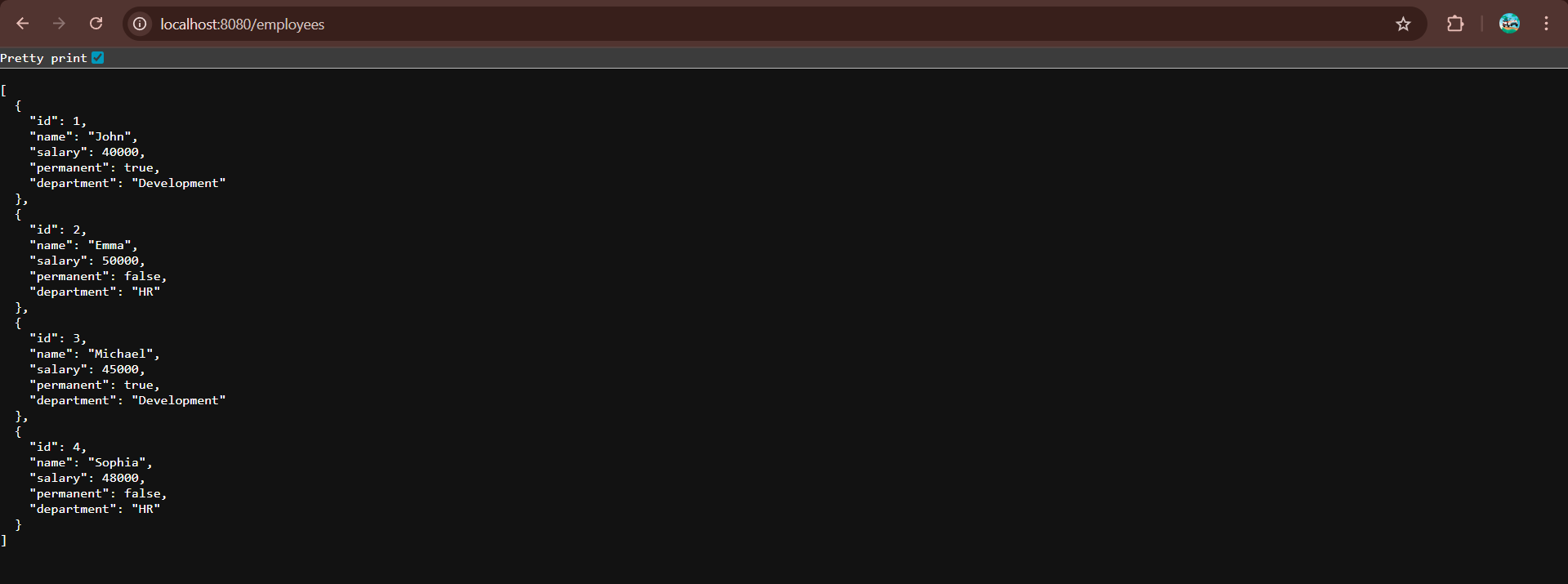
public static void main(String[] args) {

SpringApplication.run(SpringLearnApplication.class, args);

}

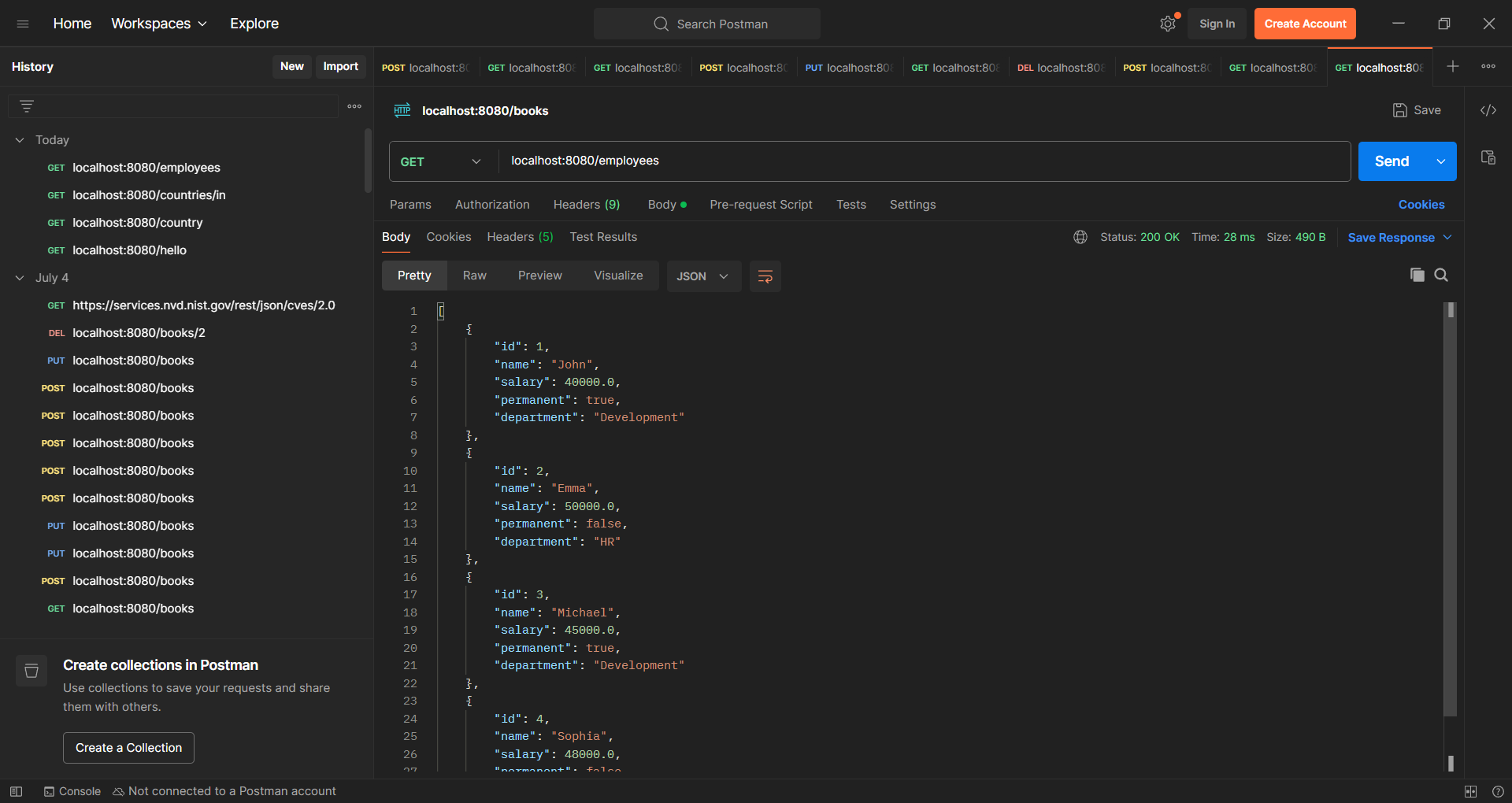
}  
  
**Output:**





​​​​​​

**Test ​the service using postman.**



**Create REST service for department**

**DepartmentController.java**

package com.cognizant.spring\_learn.controller;

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

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

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

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

import java.util.List;

*@RestController*

public class DepartmentController {

*@Autowired*

private DepartmentService departmentService;

*@GetMapping*("/departments")

public List<Department> getAllDepartments() {

return departmentService.getAllDepartments();

}

}

**DepartmentDao.java**

package com.cognizant.spring\_learn.dao;

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

import org.springframework.context.ApplicationContext;

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*;

public DepartmentDao() {

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

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

}

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 org.springframework.transaction.annotation.Transactional;

import java.util.List;

*@Service*

public class DepartmentService {

*@Autowired*

private DepartmentDao departmentDao;

*@Transactional*

public List<Department> getAllDepartments() {

return departmentDao.getAllDepartments();

}

}

**Department.java**

package com.cognizant.spring\_learn.model;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class Department {

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

private int id;

private String name;

public Department() {

***LOGGER***.debug("Inside Department Constructor");

}

public int getId() {

***LOGGER***.debug("getId() called");

return id;

}

public void setId(int id) {

***LOGGER***.debug("setId() called");

this.id = id;

}

public String getName() {

***LOGGER***.debug("getName() called");

return name;

}

public void setName(String name) {

***LOGGER***.debug("setName() called");

this.name = name;

}

*@Override*

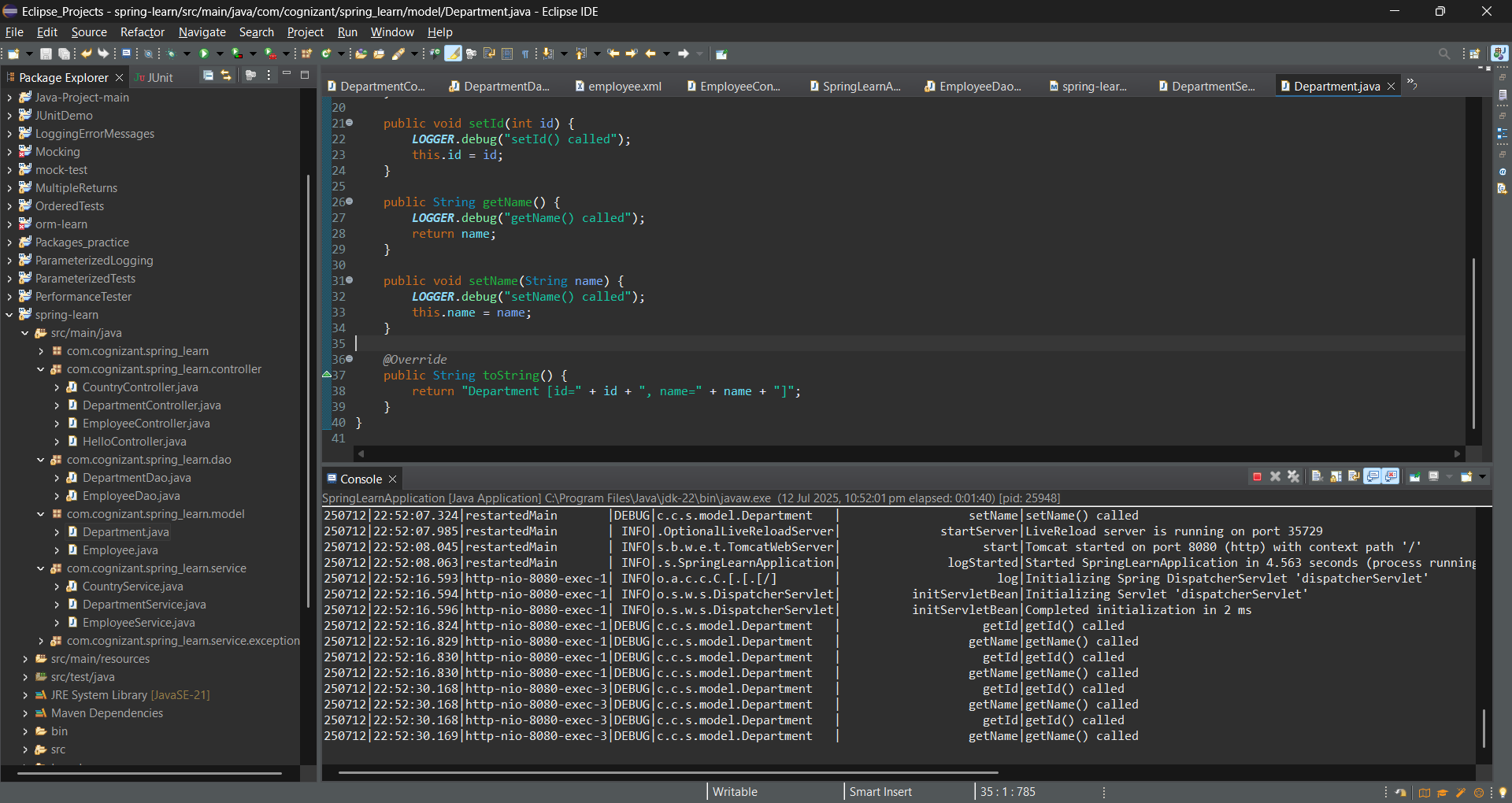
public String toString() {

return "Department [id=" + id + ", name=" + name + "]";

}

}

**Output:**



**Test ​the service using postman.**

