Step 1: Defining Query Methods

We'll enhance our EmployeeRepository and DepartmentRepository to support custom queries using keywords in method names and the @Query annotation.

EmployeeRepository.java:

package com.example.employeemanagementsystem.repository;

import com.example.employeemanagementsystem.entity.Employee;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.data.jpa.repository.Query;

import org.springframework.stereotype.Repository;

import java.util.List;

@Repository

public interface EmployeeRepository extends JpaRepository<Employee, Long> {

List<Employee> findByName(String name);

List<Employee> findByDepartment(Department department);

// Custom query method using keyword

List<Employee> findBySalaryGreaterThan(double salary);

// Custom query method using @Query annotation

@Query("SELECT e FROM Employee e WHERE e.department.name = ?1")

List<Employee> findByDepartmentName(String departmentName);

}

DepartmentRepository.java:

package com.example.employeemanagementsystem.repository;

import com.example.employeemanagementsystem.entity.Department;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.data.jpa.repository.Query;

import org.springframework.stereotype.Repository;

import java.util.List;

@Repository

public interface DepartmentRepository extends JpaRepository<Department, Long> {

Department findByName(String name);

// Custom query method using keyword

List<Department> findByLocation(String location);

// Custom query method using @Query annotation

@Query("SELECT d FROM Department d WHERE d.employees.size > ?1")

List<Department> findByEmployeeCountGreaterThan(int employeeCount);

}

Step 2: Named Queries

We'll define and execute named queries using the @NamedQuery and @NamedQueries annotations.

Employee.java:

package com.example.employeemanagementsystem.entity;

import javax.persistence.Entity;

import javax.persistence.NamedQueries;

import javax.persistence.NamedQuery;

import javax.persistence.Table;

@Entity

@Table(name = "employees")

@NamedQueries({

@NamedQuery(name = "Employee.findBySalaryRange",

query = "SELECT e FROM Employee e WHERE e.salary BETWEEN ?1 AND ?2"),

@NamedQuery(name = "Employee.findByDepartmentAndSalary",

query = "SELECT e FROM Employee e WHERE e.department.name = ?1 AND e.salary > ?2")

})

public class Employee {

// ...

}

Department.java:

package com.example.employeemanagementsystem.entity;

import javax.persistence.Entity;

import javax.persistence.NamedQueries;

import javax.persistence.NamedQuery;

import javax.persistence.Table;

@Entity

@Table(name = "departments")

@NamedQueries({

@NamedQuery(name = "Department.findByLocationAndEmployeeCount",

query = "SELECT d FROM Department d WHERE d.location = ?1 AND d.employees.size > ?2"),

@NamedQuery(name = "Department.findByManager",

query = "SELECT d FROM Department d WHERE d.manager.name = ?1")

})

public class Department { // ...}