**Exercise 6: Employee Management System - Implementing Pagination and Sorting**

**Business Scenario:**

Add pagination and sorting capabilities to your employee search functionality.

**Instructions:**

1. **Pagination:**
   * Implement pagination for the employee list using **Page** and **Pageable**.
2. **Sorting:**
   * Add sorting functionality to your queries.
   * Combine pagination and sorting in your search endpoint.

**Solution**

**EmployeeRepository.java**

package com.employee.employeemanagement.repository;

import com.employee.employeemanagement.entity.Employee;

import org.springframework.data.domain.Page;

import org.springframework.data.domain.Pageable;

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

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

import org.springframework.data.repository.query.Param;

public interface EmployeeRepository extends JpaRepository<Employee, Long> {

Page<Employee> findByName(String name, Pageable pageable);

@Query("SELECT e FROM Employee e WHERE e.department.name = :departmentName")

Page<Employee> findByDepartment(@Param("departmentName") String departmentName, Pageable pageable);

}

**EmployeeService.java**

import com.employee.employeemanagement.entity.Employee;

import com.employee.employeemanagement.repository.EmployeeRepository;

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

import org.springframework.data.domain.Page;

import org.springframework.data.domain.PageRequest;

import org.springframework.data.domain.Pageable;

import org.springframework.data.domain.Sort;

import org.springframework.stereotype.Service;

@Service

public class EmployeeService {

@Autowired

private EmployeeRepository employeeRepository;

public Page<Employee> getEmployeesByName(String name, int page, int size, String sortBy) {

Pageable pageable = PageRequest.of(page, size, Sort.by(sortBy));

return employeeRepository.findByName(name, pageable);

}

public Page<Employee> getEmployeesByDepartment(String departmentName, int page, int size, String sortBy) {

Pageable pageable = PageRequest.of(page, size, Sort.by(sortBy));

return employeeRepository.findByDepartment(departmentName, pageable);

}

}

**Main.java**

package com.employee.employeemanagement;

import com.employee.employeemanagement.entity.Employee;

import com.employee.employeemanagement.service.EmployeeService;

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

import org.springframework.boot.CommandLineRunner;

import org.springframework.data.domain.Page;

import org.springframework.stereotype.Component;

@Component

public class Main implements CommandLineRunner {

@Autowired

private EmployeeService employeeService;

@Override

public void run(String... args) throws Exception {

Page<Employee> employeesPageByName = employeeService.getEmployeesByName("John Doe", 0, 5, "name");

employeesPageByName.forEach(System.out::println);

Page<Employee> employeesPageByDepartment = employeeService.getEmployeesByDepartment("IT", 0, 5, "email");

employeesPageByDepartment.forEach(System.out::println);

}

}