**Exercise 5: Employee Management System - Defining Query Methods**

**EmployeeRepository**

package com.employee.management.system.repositories;

import com.employee.management.system.entities.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> {

// Custom query method using keywords

List<Employee> findByEmailContaining(String email);

// Custom query method using @Query annotation

@Query("SELECT e FROM Employee e WHERE e.name LIKE %:name%")

List<Employee> findEmployeesByName(@Param("name") String name);

// Named query using @NamedQuery

@NamedQuery(name = "Employee.findByDepartment",

query = "SELECT e FROM Employee e WHERE e.department = :department")

List<Employee> findEmployeesByDepartment(@Param("department") Department department);

}

**DepartmentRepository**

package com.employee.management.system.repositories;

import com.employee.management.system.entities.Department;

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

import org.springframework.stereotype.Repository;

import java.util.List;

@Repository

public interface DepartmentRepository extends JpaRepository<Department, Long> {

// Custom query method using keywords

Department findByName(String name);

}

In the above code, i have enhanced the EmployeeRepository and DepartmentRepository to support custom queries.

i have defined custom query methods using keywords in method names, such as findByEmailContaining and findByName.

i have also implemented custom query methods using the @Query annotation, such as findEmployeesByName.

Additionally, i have defined a named query using the @NamedQuery annotation, Employee.findByDepartment, which can be executed using the findEmployeesByDepartment method.

Note that i have used @Param to specify the parameter names in the custom query methods.