EmployeeRepository.java

List<Employee> findByEmail(String email);

DepartmentRepository.java

List<Department> findByNameContaining(String keyword);

EmployeeRepository.java

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

List<Employee> findByNameAndDepartmentName(@Param("name") String name, @Param("departmentName") String departmentName);

DepartmentRepository.java

@Query("SELECT d FROM Department d WHERE size(d.employees) >= :minEmployees")

List<Department> findByMinEmployees(@Param("minEmployees") int minEmployees);

Employee.java

@Entity

@Table(name = "employees")

@Data

@NamedQueries({

@NamedQuery(

name = "Employee.findByDepartmentName",

query = "SELECT e FROM Employee e WHERE e.department.name = :departmentName"

),

@NamedQuery(

name = "Employee.findByEmailContaining",

query = "SELECT e FROM Employee e WHERE e.email LIKE %:email%"

)

})

public class Employee {

// fields, constructors, getters, and setters

}

EmployeeRepository.java

@Query(name = "Employee.findByDepartmentName")

List<Employee> findEmployeesByDepartmentName(@Param("departmentName") String departmentName);

@Query(name = "Employee.findByEmailContaining")

List<Employee> findEmployeesByEmailContaining(@Param("email") String email);