**EmployeeRepository.java:**

public interface EmployeeRepository extends JpaRepository<Employee, Long> {

// Using keywords in method names

List<Employee> findByFirstName(String firstName);

List<Employee> findByLastName(String lastName);

List<Employee> findByDepartment(Department department);

// Implementing custom query methods using @Query

@Query("SELECT e FROM Employee e WHERE e.salary > :salary")

List<Employee> findEmployeesWithSalaryGreaterThan(@Param("salary") double salary);

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

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

}

**DepartmentRepository.java:**

public interface DepartmentRepository extends JpaRepository<Department, Long> {

// Using keywords in method names

List<Department> findByName(String name);

List<Department> findByLocation(String location);

// Implementing custom query methods using @Query

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

List<Department> findDepartmentsWithMoreThanXEmployees(@Param("employeeCount") int employeeCount);

}

**Employee.java:**

@Entity

@NamedQueries({

@NamedQuery(name = "Employee.findBySalaryRange",

query = "SELECT e FROM Employee e WHERE e.salary BETWEEN :minSalary AND :maxSalary"),

@NamedQuery(name = "Employee.findByDepartmentName",

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

})

public class Employee {

// ...

}

**Department.java:**

@Entity

@NamedQueries({

@NamedQuery(name = "Department.findByLocation",

query = "SELECT d FROM Department d WHERE d.location = :location"),

@NamedQuery(name = "Department.findByEmployeeCount",

query = "SELECT d FROM Department d WHERE d.employees.size > :employeeCount")

})

public class Department {

// ...

}

**EmployeeService.java:**

@Service

public class EmployeeService {

@Autowired

private EntityManager entityManager;

public List<Employee> findEmployeesBySalaryRange(double minSalary, double maxSalary) {

Query query = entityManager.createNamedQuery("Employee.findBySalaryRange");

query.setParameter("minSalary", minSalary);

query.setParameter("maxSalary", maxSalary);

return query.getResultList();

}

public List<Employee> findEmployeesByDepartmentName(String departmentName) {

Query query = entityManager.createNamedQuery("Employee.findByDepartmentName");

query.setParameter("departmentName", departmentName);

return query.getResultList();

}

}

**DepartmentService.java:**

@Service

public class DepartmentService {

@Autowired

private EntityManager entityManager;

public List<Department> findDepartmentsByLocation(String location) {

Query query = entityManager.createNamedQuery("Department.findByLocation");

query.setParameter("location", location);

return query.getResultList();

}

public List<Department> findDepartmentsByEmployeeCount(int employeeCount) {

Query query = entityManager.createNamedQuery("Department.findByEmployeeCount");

query.setParameter("employeeCount", employeeCount);

return query.getResultList();

}

}