**Step 1: Define Interface-Based Projections**

public interface EmployeeSummary {

Long getId();

String getFirstName();

String getLastName();

String getDepartmentName();

}

**Step 2: Define Class-Based Projections**

public class DepartmentInfo {

private Long id;

private String name;

private String location;

public DepartmentInfo(Long id, String name, String location) {

this.id = id;

this.name = name;

this.location = location;

}

// getters

} **Step 3: Use @Value and Constructor Expressions**

public interface EmployeeSummary {

@Value("#{target.id}")

Long getId();

@Value("#{target.firstName}")

String getFirstName();

@Value("#{target.lastName}")

String getLastName();

@Value("#{target.department.name}")

String getDepartmentName();

}

**DepartmentInfo.java:**

public class DepartmentInfo {

private Long id;

private String name;

private String location;

public DepartmentInfo(@Value("id") Long id,

@Value("name") String name,

@Value("location") String location) {

this.id = id;

this.name = name;

this.location = location;

}

// getters

}

**EmployeeRepository.java:**

public interface EmployeeRepository extends JpaRepository<Employee, Long> {

List<EmployeeSummary> findByDepartmentName(String departmentName);

}

**DepartmentRepository.java:**

public interface DepartmentRepository extends JpaRepository<Department, Long> {

List<DepartmentInfo> findAllByLocation(String location);

}