**EXERCISE 5**

**1. Defining Query Methods**

**Using Keywords in Method Names**

Spring Data JPA allows you to create custom query methods by naming the methods with specific keywords that describe the query. Spring Data JPA will automatically generate the appropriate query based on the method name.

* **findByDepartmentId(Long departmentId)**: This method name is parsed to create a query that retrieves all employees with the given department ID.
* **findByEmail(String email)**: This method name generates a query to find an employee based on their email.
* **findByNameContaining(String keyword)**: This method name creates a query to find employees whose names contain the specified substring.
* **findByJoinDateAfter(Date date)**: This method name generates a query to find employees who joined after the given date.

**Implementing Custom Query Methods Using the @Query Annotation**

For more complex queries or when you need to customize the query logic, you use the @Query annotation. This allows you to write JPQL (Java Persistence Query Language) or native SQL queries.

* **findDepartmentByName(String name)**: This method uses a JPQL query to find a department by its name. ?1 refers to the first method parameter.
* **findDepartmentsWithMoreThanEmployees(int employeeCount)**: This method uses a JPQL query to find departments with more than the specified number of employees. SIZE(d.employees) is a JPQL function that returns the number of employees in the department.

**2. Named Queries**

Named queries are predefined queries that can be defined in the entity class using the @NamedQuery or @NamedQueries annotation. Named queries are useful for queries that are reused across different parts of the application.

* **@NamedQuery(name = "Department.findByName", query = "SELECT d FROM Department d WHERE d.name = :name")**: This named query finds a department by its name using a parameter placeholder :name.
* **@NamedQuery(name = "Department.findAllDepartments", query = "SELECT d FROM Department d")**: This named query retrieves all departments.

To execute these named queries from the repository, you use the @Query annotation with the name of the named query.