

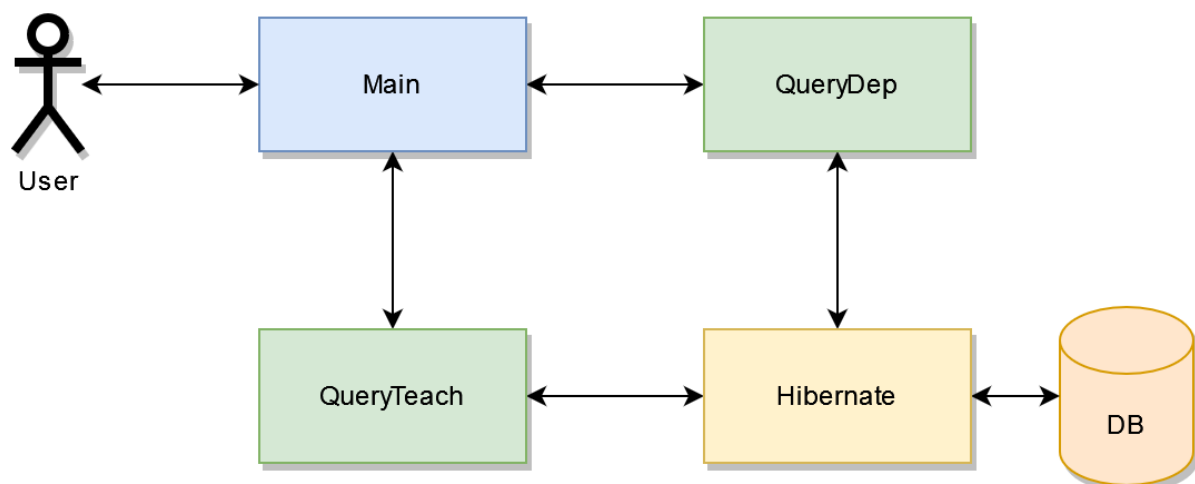
ACTIVITY 4.3

1. CONNECT TO DATABASE

Configure a new Project and use Hibernate to connect to the database created in activity 4.1 (database named act4_1).

We will have a three classes:

- **Main**: interacts with the user. To access the database uses the methods of QueryDep and QueryTeach.
- **QueryDep**: access the database and provides the Main class with the data related to Departments.
- **QueryTeach**: access the database and provides the Main class with the data related to Teachers.



2. QUERYDEPT CLASS

Create a class called QueryDep. It will contain the following static methods. Each method should open and close a Hibernate Session if it is needed. Nothing more than static methods or attributes should be included in this class

- **public static void showDepartment(Departments dep)**
Prints in the console the department number, the department name, the office and the number of teachers belonging to the department.
- **public static Departments[] getAllDepartments()**
Uses a query in HQL to retrieve all Departments.

- **public static Departments getDepartmentByName(String patternName)**
Uses a query in HQL to retrieve the department whose name is LIKE the pattern introduced as argument. **Use parameters with the :name syntax. Use .uniqueResult(). Be careful if several results are returned, catch the exception and return only the first result.**
- **public static double getAverageSalaryofDepartment(String depName)**
Uses a query in HQL to obtain the average salary of the department whose name is THE SAME AS the name introduced as argument. **Use parameters with the :name syntax. Use .uniqueResult().**
- **public static HashMap<String, Double> getAverageSalaryPerDept()**
Uses a query in HQL to obtain the name of each department and its average salary. The result is returned in a HashMap that relates the name of the department with the average salary.

3. QUERYTEACH CLASS

Create a class called QueryTeach. It will contain the following static methods. Each method should open and close a Hibernate Session if it is needed. Nothing more than static methods or attributes should be included in this class

- **public static void showTeacher(Teachers dep)**
Prints in the console the name, surname, email, startDate, salary and the name of the department to which the teacher belongs.
- **public static Teachers[] getAllTeachers()**
Uses a query in HQL to retrieve all Teachers.
- **public static Teachers getMostVeteranTeacher()**
Uses a query in HQL to retrieve the most veteran teacher (minimum start_date). **Use .uniqueResult(). Be careful if several results are returned, catch the exception and return only the first result.**
- **public static int setSalary(int newSalary)**
Uses a query in HQL to set the salary of all Teachers to the one introduced as argument. **Use parameters with the :name syntax.**
- **public static int riseSalaryOfSeniors(int numOfYearToBeSenior, int prctRise)**
Uses a query in HQL to rise the salary in a percentage indicated by *prctRise* of those Teachers that have a seniority of at least *numOfYearToBeSenior* to the one introduced as argument. The method returns the number of updated rows. **Use parameters with the :name syntax.**

Check [Hibernate manual](#) to know how to work with dates. (11.4.5)

- **public static int deleteTeachersOfDepartment(String depName)**
Uses a query in HQL to delete all teachers belonging to the department whose name is THE SAME AS the name introduced as argument.. **Use parameters with the :name syntax.**

4. MAIN CLASS

The main class should show a menu to test each method.

1. Show all departments
2. Show department whose name matches a pattern
3. Get average salary of a department (by name)
4. Show average salary of each department
5. Show all teachers
6. Show most veteran teacher
7. Set salary
8. Rise salary of senior teachers
9. Delete teachers of a department
10. Quit

After executing an option, the menu will be shown again.

Each option will ask for all the information needed to make the queries, obtain the result by calling the corresponding method of QueryDep or QueryTeach classes and print the result in console, either using the showing methods of QueryDep or QueryTeach or using plain println for simple data.