Iben Andersen
cph-ia62@cphbusiness.dk

ORM + JPQL

General part

1) Explain the rationale behind the topic Object Relational Mapping and the Pros and Cons in using ORM

Object Relational Mapping (ORM) is a functionality which is used for converting data between an object and a relational database which are two incompatible systems. The functionality is mapping an object state to a database column and it is able to insert, update and deleting objects.

Some of the pros of ORM are that you are able to eliminate manual mapping work and that you often reduce the amount of code that needs to be written in order to convert data.

A con of using ORM can be that the code can get a bit abstract which can make the code obscure and hard to understand.

2) Discuss how we usually have queried a relational database

Before we learned about Object Relational Mapping, we were mapping manually in Java using JDBC every time that we had to work with a database. We had to make a Database Connection class to get access to the database. We also had to make a mapper method with SQL-query in order to get access to data from the database and in order to insert and delete data from database.

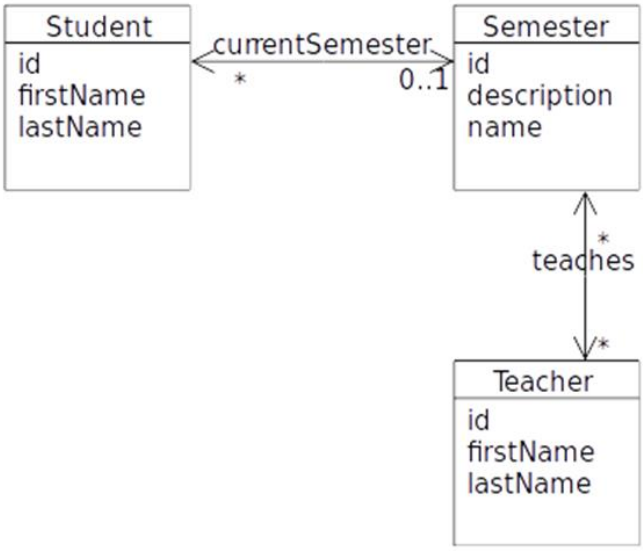
3) Discuss the methods we can use to query a JPA design and compare with what you explained above

With Java Persistence API (JPA) you are able to use methods and classes to persist, store, data into databases. An important annotation from JPA is @Entity. The @Entity annotation is used to specify that this class is representing an entity type. It is managed by the persistence. Other important annotations from JAP are the relation annotations @ManyToMany, @ManyToOne, @OneToMany and @OneToOne which are used to specify database relationships between entities.

A big difference between Database handling using JDBC and JPA is that when you use JDBC you make manual mappers in order to access and add data to a database. When using JPA you use annotations in mapping java objects which makes JPA easier to work with.

Iben Andersen
cph-ia62@cphbusiness.dk

Practical part



A) Use NetBeans to create a set of matching Entity Classes (see hints at the end). Make sure you understand what was created, and that you understand how classes and tables are related

script.sql is set up and database *ExampPreparationJPQL* was created from script.

NetBeans project ExamPrep_JPQL is created and Entity Classes Semester, Student and Teacher are created from script.

B) Investigate the generated Entity classes and observe the NamedQueries generated by the Wizard

NamedQueries are generated in each Entity class.

C) Create Dynamic Queries in a façade class (or if possible, use a named Query generated by the wizard) to solve the following problems:

Facade is implemented in the NetBeans project ExamPrep_JPQL in the Source Package facades Facade.java.