JDBC WORKSHOP

# Create EntityManager and EntityManagerImpl files

In src/main/java/ro/teamnet/zth/api/em create an interface which will manage the most important operation on a DB. First create an interface EntityManager.java in which you will define the following methods:

* <T> T findById(Class<T> entityClass, Long id);
* <T> List<T> findAll(Class<T> entityClass);
* <T> Object insert(T entity);
* <T> T update(T entity);
* void delete(Object entity);

Next you need to implement EntityManager.java interface in class EntityManagerImpl.java.

Steps:

1. **Method <T> T findById(Class<T> entityClass, Long id)**

* create a connection to DB;
* get table name;
* get table columns;
* set column value with the value from entity;
* create a Condition object where you need to set id value;
* create a QueryBuilder object where you set the name of table, query type, columns and conditions;
* call createQuery() method from QueryBuilder.java;
* create a ResultSet object;
* create a Statement object and execute the query;
* create an instance of class and populate the object;
* return the instance;

1. **Method <T> List<T> findAll(Class<T> entityClass)**

* create a connection to DB;
* get table name;
* get table columns;
* set column value with the value from entity;
* create a Condition object where you need to set id value;
* create a QueryBuilder object where you set the name of table, query type and columns;
* call createQuery() method from QueryBuilder.java;
* create a ResultSet object;
* create a Statement object and execute the query;
* go through ResulSet object and populate the list;
* return the list;

1. **Method <T> Object insert(T entity)**

* create a connection to DB;
* get table name;
* get table columns;
* set column value with the value from entity;
* create a QueryBuilder object where you set the name of table, query type and columns;
* call createQuery() method from QueryBuilder.java;
* create a Statement object and execute the query;
* return the object;

1. **Method <T> T update(T entity)**

* create a connection to DB;
* get table name;
* get table columns;
* set column value with the value from entity;
* create a Condition object where you need to set id value which will be updated;
* create a QueryBuilder object where you set the name of table, query type, columns and conditions;
* call createQuery() method from QueryBuilder.java;
* create a Statement object and execute the query;
* return the object;

1. **Method void delete(Object entity)**

* create a connection to DB;
* get table name;
* get table columns;
* set column value with the value from entity;
* create a Condition object where you need to set id value which will be deleted;
* create a QueryBuilder object where you set the name of table, query type and conditions;
* call createQuery() method from QueryBuilder.java;
* create a Statement object and execute the query;

# Test methods EntityManagerImpl.java

Create in *src/test/java/ro/teamnet/zth/api/em*  a public class *EntityManagerImplTest.java* which will test all the methods implemented in *EntityManagerImpl.java*.

# Create DepartmentDao and LocationDao files

Create in *src/main/java/ro/teamnet/zth/appl/dao*, 2 public classes: *DepartmentDao.java, LocationDao.java*, these will be used to call created methods from *EntityManagerImpl.java.*

1. **DepartmentDao.java**

* create an EntityManager object;
* create 5 methods which will call *EntityManagerImpl.java* methods;

1. **LocationDao.java**

* do the same thing as DepartmentDao.java;

# Create tests for DepartmentDao and LocationDao files

Create in *src/test/java/ro/teamnet/zth/appl* 2 public classes *DepartmentDao.java* and *LocationDao.java* which will test all the methods implemented in *DepartmentDao.java* and *LocationDao.java.*