**COMPUTING SUBJECT:** Arrays of user defined classes

**TYPE:** GROUP WORK ASSIGNMENT

**IDENTIFICATION:** CUSTOMERDBDAO/MC

**COPYRIGHT:** *Michael Claudius*

**DEGREE OF DIFFICULTY:** Easy

**TIME CONSUMPTION:** 2 hours

**EXTENT:** 50 lines

**OBJECTIVE:** Database and handling collections of objects

**COMMANDS:**

**IDENTIFICATION:** FANTASY HOTEL NO K: CUSTOMERDBDAO/MC

*This exercise is no. K in the story of Fantasy Hotel with a Database. The previous exercises were:*

*Fantasy Hotel No. J: CustomerDB*

*Later on we will investigate persistent objects on a database.*

The mission

You are to develop a program, which can manage, control and handle writing and reading of customer objects using a relational database.

The general objective

We want a program, which permanently can register customer data in a database using the DAO-classes and a GUI-interface.

The objective of this exercise

You are to develop program, which can combine a database and a customer register and insert/select/update/delete customer data utilizing the previously made CustomerRegister class and a CustomerDAO class

Note: A preliminary project version (HotelDatabase2013 vs. 1.9) -with the Model, View, Controller DAO and Utility packages - can be downloaded from your teacher’s home page. Alternatively you can use your own project and just copy the DAO-package.

Assignment 1: TestCustomerDAO, Select and create customers

In the new class TestCustomerDAO you must try to test the class CustomerDAO by declaring an object of CustomerRegister and using methods in the CustomerDAO.

In *main* show how to:

* declare and construct the object, customerReg ,of the class CustomerRegister
* declare an object customerDAO of the class CustomerDAO
* use the *getAllCustomersAsResultSet* method to select all customers, then
* traverse the resultset
* create customer object for each hit
* add this object to a register of customers
* finally print out the register using a toString mehod

*Tip; Very similar to CustomerQuery*

Assignment 2: TestCustomerDAO, Insert values into tables

*In main* show how to:

read in customer data using (JOptionPane or Keyboard)

create a customer object

use the method *addCustomer* save the customer object in the database

*Tip; Very similar to parts of CustomerInsert*

Assignment 3: CustomerDAO, Insert customer

To insert customer data directly extend CustomerDAO with the method:

*insertCustomerDB(String customerNo, String name, String customerType, int year, int stays)*

Create an insert string with five “?”

Build up the sql-sentence

Assing values to the question-marks (like: prepStmt.setString(1, customerNo))

Execute the prepared statement

*Tip; Very similar to parts of CustomerInsert*

*Try also to use this method in main.*

Assignment 3: CustomerDAO Class: Search a customer

Extend the class CustomerDao with a method to search for a specific customer with a specific customerNo:

*public Customer getCustomerDB(String aCustomerNo)*

Creates a statement stmt using DBTool

Select the customer with aCustomerNo from the database as a Resultset

Create customer object for the hit, if any

Return this object

Compile!

Assignment 4: TestCustomerDAO: Search a specific customer

In the *main* you must try to test the method getCustomerDB by adding the constructions:

* call the method *getCustomerDB* and print out the customer object

Compile and Execute.

Assignment 5: CustomerDAO: Select customers return as a register

Extend the class CustomerDAO with a method to select and all customers and return the resultset as a CustomerRegister:

*public CustomerRegister getAllCustomersAsRegister ()*

Creates a statement stmt using DBTool

Select all customers from the database as a Resultset

Create customer object for the hit, if any

Add customer to a customerRegister

Return the customerRegister

Compile!

Assignment 6: TestCustomerDAO: Select customers

In TestCustomerDAO you must try to test the method *getAllCustomersAsRegister.*  In *main* show how to:

* use the *getAllCustomersAsResultSet* method to select all customers, then
* finally print out the register using a toString mehod

*Tip; Super easy, two lines of code…*

ITS TIME TO STOP TESTING AND PROCEED TO A GUI-PROGRAM WITH DAO-CLASSES