Lab#2

List of students using DAO

In this laboratory work I have implemented Data Access Object pattern to create webapplication where user can review list of students and also manage it: add new students, delete, update, and find them by id or name.

I store table STUDENTS1 in the MySQL database. For connecting and changing database I have Queries class. There are static functions for manipulating data:

- static void insertUser(String firsName, String lastName) adding new fields in the table where STUDENT_ID is auto increment, last and first name are input;
- static ArrayList<User> findAll() returns ArrayList of all students that are in the
 database;
- static void updateUser(String id, String firstName, String lastName) according id function set new values for first and last name of student;
- static void deleteteUser(String id) using input id function finds needed student and deletes him from the table;
- static User findByID(String id) using id function finds data about certain student, and returns appropriate user;
- static Vector<User> findByName(String name, String column) function finds student by the column: last or first name, and returns appropriate student(user);

Here I have following statements to connect with database:

```
Class.forName("com.mysql.jdbc.Driver");
    conn = DriverManager.getConnection(Main.DB_URL, Main.USER, Main.PASS);
and to manage data I have SQL statements as
    String sql = "insert into students1 (first_name, last_name) values
('"+firsName+"', '"+lastName+"'); ";
```

As a *model I* have User class. Its objects have id, last name and first name.

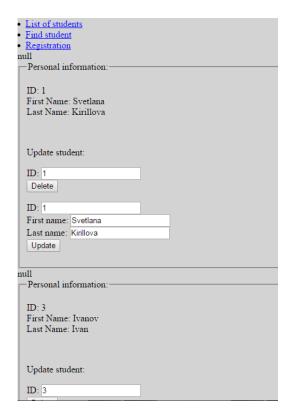
For good representation I have used Transform view and Two Step view patterns. So *Transform class* have next functions:

```
public static String viewStudents(User user) - returns html code for one student data;
static String viewForUpdating(String id, String firstName, String lastName) -
returns view for updating page;
static String viewForDeleting(String id) - returns view for deleting page;
static String listOfStudents() - returns view for list of all students of
database;
```

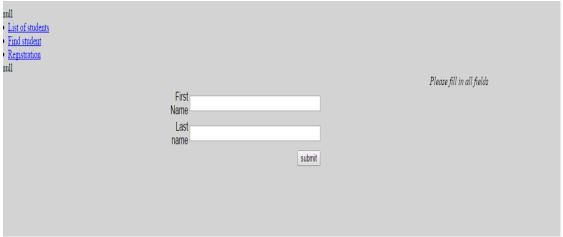
In the Main class runs HttpServer.

Thus Data Access Object pattern than works independently, and java code is not attached to SQL code.

User interface:







Data transaction:

