

In this lab i implement an GUI app that read list of students from database and display it. For ORM pattern i used Hibernate. For GUI i used swing and awt java libraries.

In Student.hbm.xml file i store which fields from model implies in which column when model maps in table.

In hibernate.cfg.xml i store connection configuration.

When app start it calls listStudent() function. This function return String. After 2 parsing of string we write data that come in JTextField.

In the bottom there is textfields for adding new student.

Update, Add, and Delete

When you want to add new student you have to fulfil in the empty Text Edits name, lastname, faculty and year.

After this we put data in the ms.add() function. After this we are adding new elements(JTextBooks and JButtons) revalidate and resize our JFrame.

If you want to update info about existing students you have to click edit button next to student that you want to edit, after this textfields becomes editable and you can change info there, on edit button text becomes "UPDATE" . Also using ms(instance ManageStudent class) we put new data in ms.update() function. But before we read it from JTextFields.

Delete operation implemented in the same way as Add and Update.

Conclusion

The big advantage of Hibernate is that you interact with SQL database(in my case) without any string of SQL transaction, we just put in xml files which field from class implies on which column in table. And to update, delete, or add new Student we have to just create, or reset some fields through the ManageStudent class.