|  |  |
| --- | --- |
| Nama | : La Ode Muhammad gazali |
| NIM | : 222212696 |
| Kelas | : 2KS2 |

**MODUL 10 PRAKTIKUM PEMROGRAMAN BERORIENTASO OBJEK**

**(Input-Output)**

1. **Membaca File**

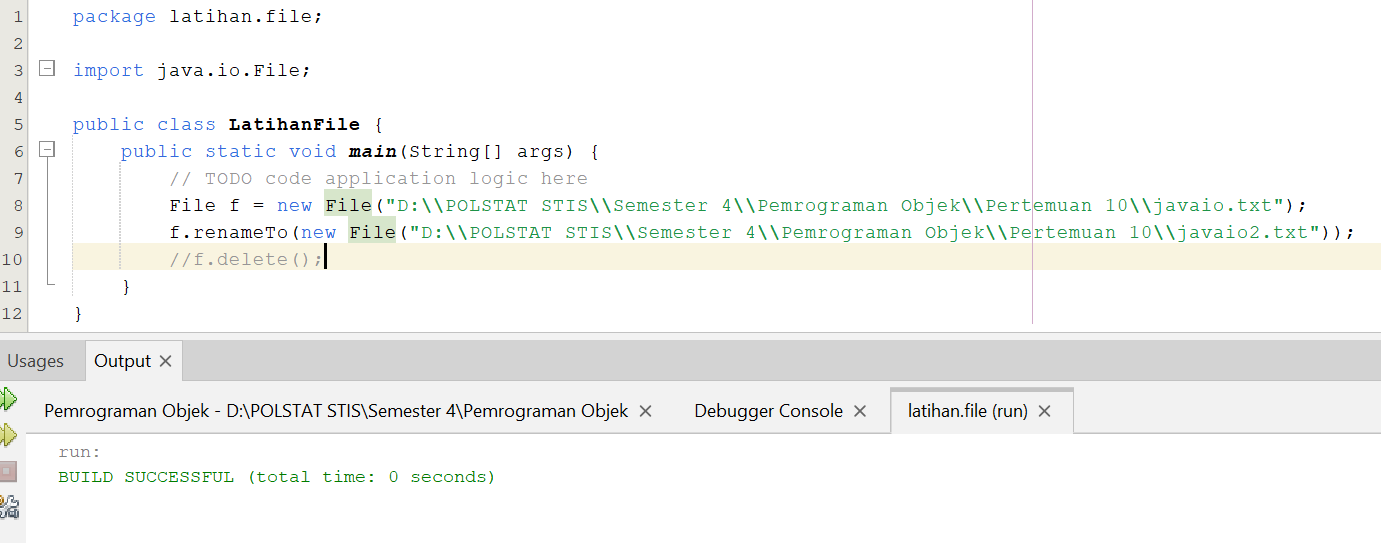
Membuat 2 file .txt

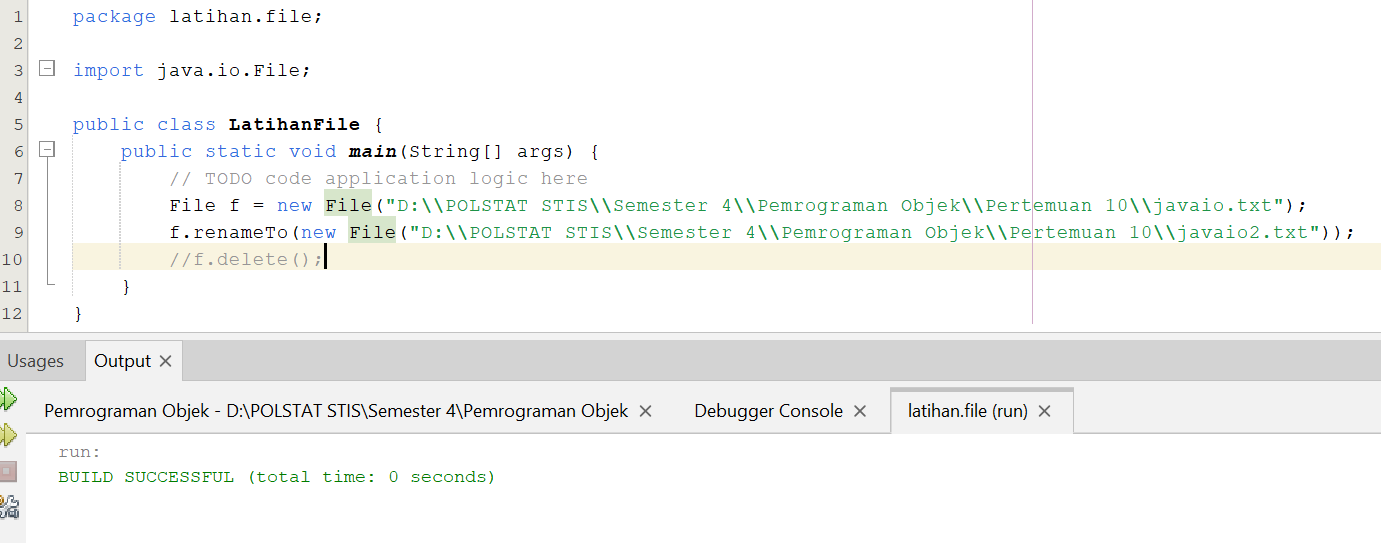


Misal ingin membaca file **javaio.txt**

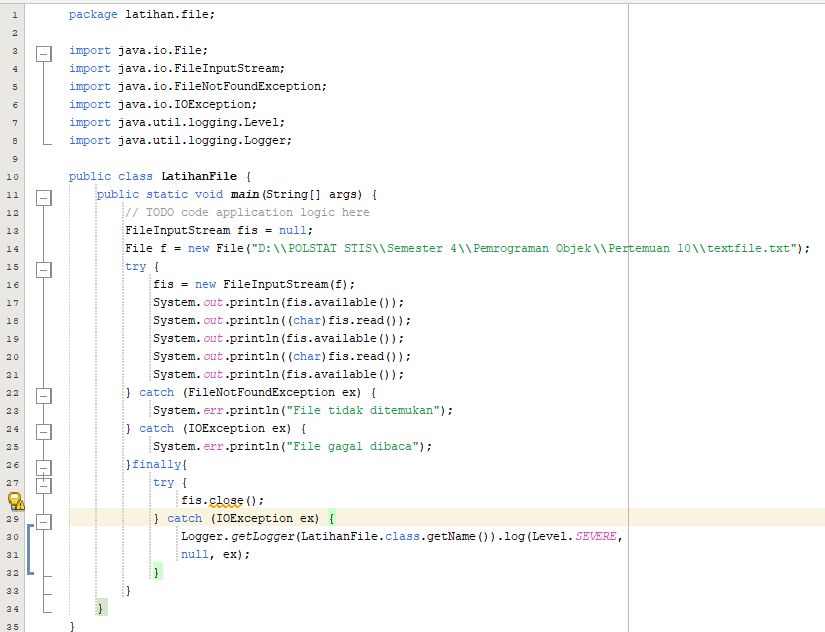


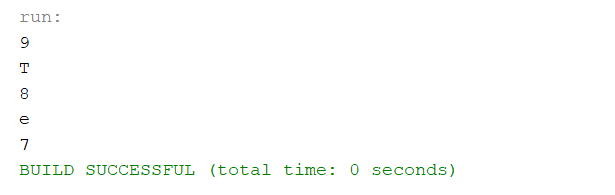
Kita juga bisa melakukan hal-hal yang biasa kita lakukan dengan explorer seperti rename file atau bahkan mendelete file. Misal rename file javaio.txt menjadi javaio2.txt



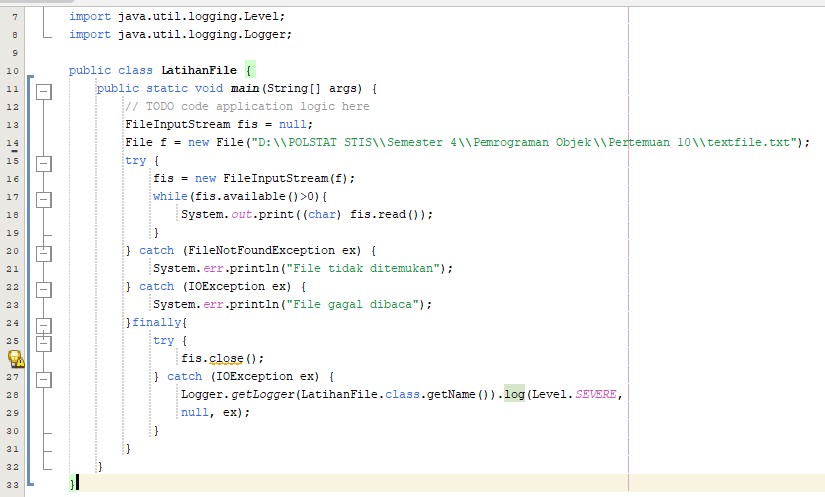
Menjadi :

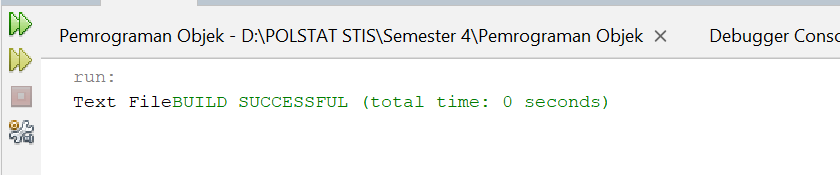
Kita juga bisa mendapatkan informasi berapa byte yang belum dibaca menggunakan metode available(). Berikut adalah listing programnya





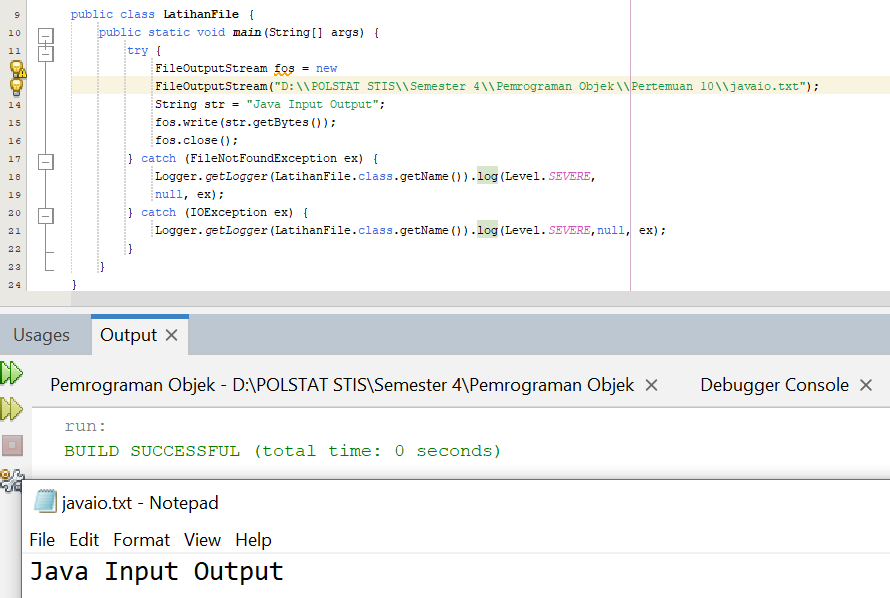
Jika ingin membaca hingga akhir teks perkarakter maka kita bisa menggunakan listing program di bawah ini:



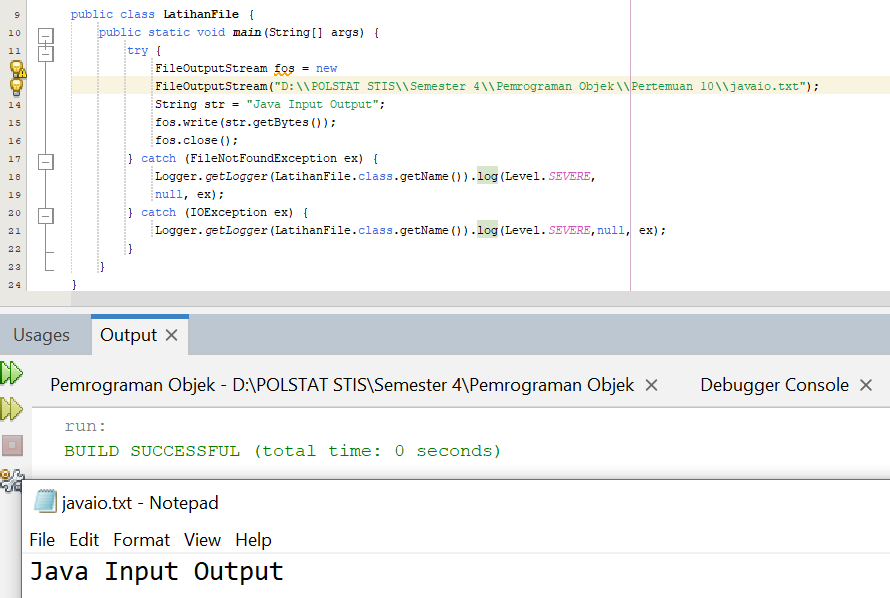


1. **Menulis File**

Untuk menulis file, perhatikan listing program di bawah ini:

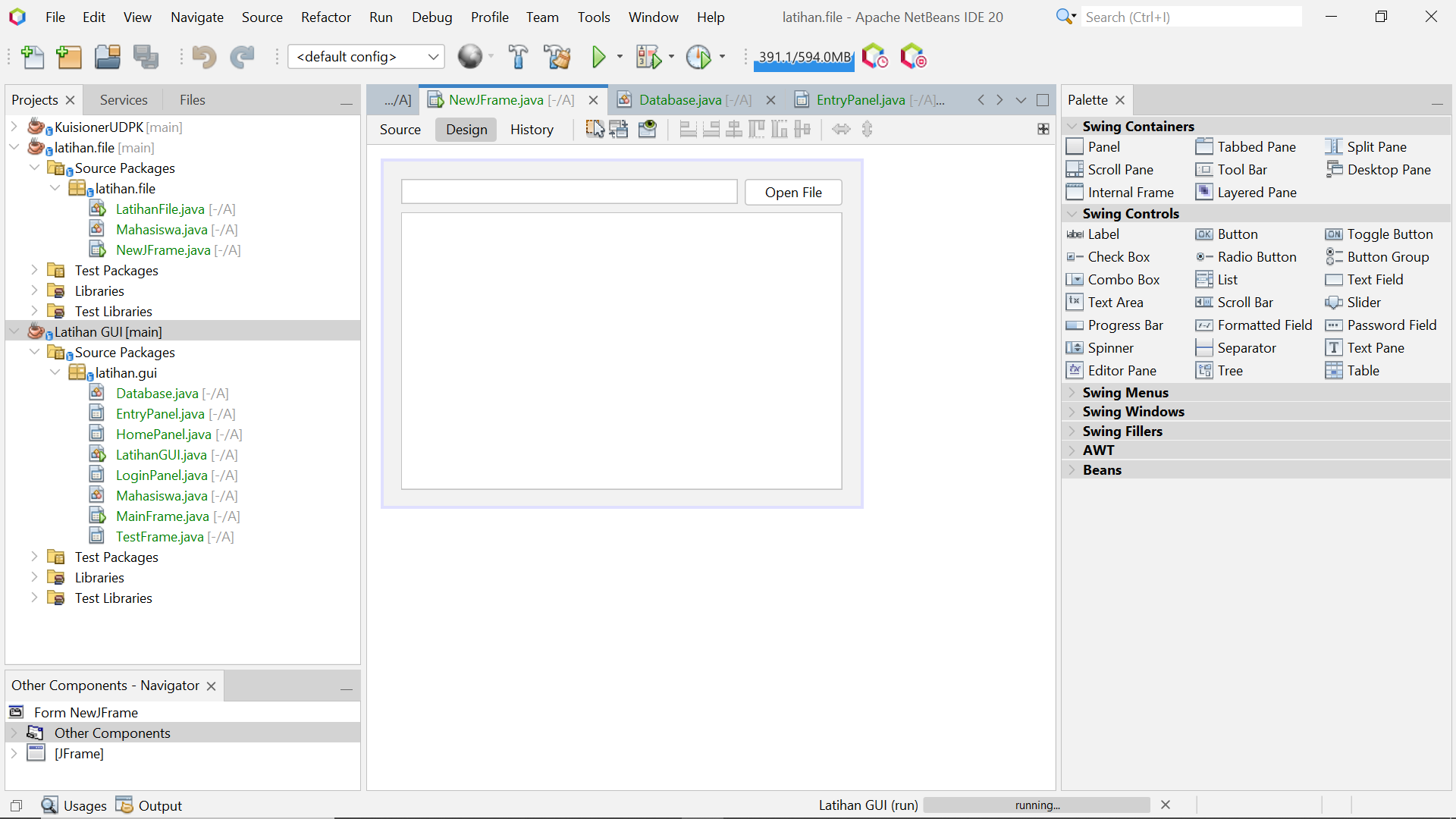


Ketika dijalankan, maka file javaio.txt akan berisi kalimat Java Input Ouput.



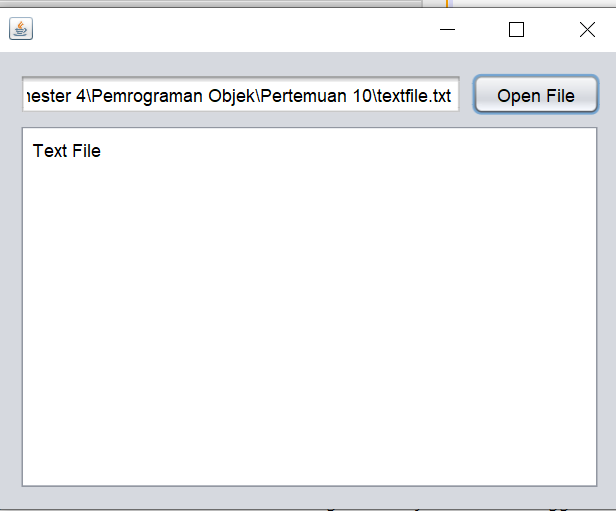
1. **GUI**

Buatlah rancangan GUI sebagai berikut:



Pada button Open File, tambahkan syntax sesuai yang ada pada modul

Maka, ketika dijlankan file tersebut. Berikut adalah tampilan layarnya



**MEMBACA DAN MENULIS OBJECT**

Untuk membaca object, kita menggunakan kelas ObjectOutputStream dan untukmembaca object kita menggunakan kelas ObjectInputStreamBuka kembali project sebelumnya, kopikan kelas mahasiwa ke project yangsekarang. Kemudian kelas mahasiswa ini harus mengimplementasikan interfaceserializable.

package latihan.file;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.ObjectOutputStream;

import java.util.logging.Level;

import java.util.logging.Logger;

public class LatihanFile {

    public static void main(String[] args) {

        try {

            FileOutputStream fos = new

            FileOutputStream("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\objek.dat");

            ObjectOutputStream oos = new ObjectOutputStream(fos);

            Mahasiswa mhs = new Mahasiswa();

            mhs.setNim("1234");

            mhs.setNama("Andi");

            oos.writeObject(mhs);

            oos.close();

            fos.close();

        } catch (FileNotFoundException ex) {

            Logger.getLogger(LatihanFile.class.getName()).log(Level.SEVERE, null, ex);

        } catch (IOException ex) {

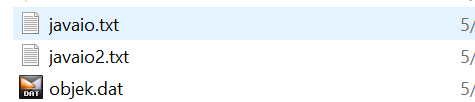
            Logger.getLogger(LatihanFile.class.getName()).log(Level.SEVERE,null, ex);

        }

    }

}

Ketika dijalankan maka di folder tersebut ada file baru bernama objek.dat yang menyimpan data 1 mahasiswa.



1. **Membaca Object**

Perhatikan listng program LatihanFile.java berikut

package latihan.file;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.ObjectOutputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

 \*

 \* @author U53R

 \*/

public class LatihanFile {

    public static void main(String[] args) {

        try {

            File f = new File("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\objek.dat");

            FileInputStream fis = new FileInputStream(f);

            ObjectInputStream ois = new ObjectInputStream(fis);

            Mahasiswa mhs = (Mahasiswa) ois.readObject();

            System.out.println(mhs.getNim());

            System.out.println(mhs.getNama());

            ois.close();

            fis.close();

            } catch (FileNotFoundException ex) {

                System.err.println("File tidak ditemukan");

            } catch (IOException ex) {

                System.err.println("File gagal dibaca");

            } catch (ClassNotFoundException ex) {

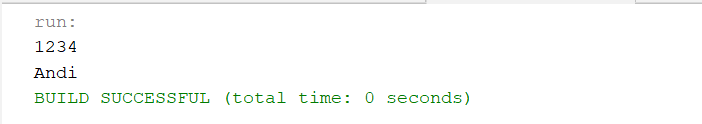
                System.err.println("Format File salah");

            }

    }

}

Jika dijalankan maka sebagai berikut:



Berikutnya buka kembali project LatihanGUI sebelumnya, kita akan menyimpan ke dalam file untuk data mahasiswa yang telah dientri. Buka kelas database.java dan implementasikan interface serializable. Jangan lupa kelas mahasiswa.java pada project tersebut juga harus implements Serializable Pada Database.java ubahlah menjadi seperti berikut:

**Database.java**

/\*

 \* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

 \* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

 \*/

package latihan.gui;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.io.ObjectOutputStream;

import java.io.Serializable;

import java.util.ArrayList;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

 \*

 \* @author U53R

 \*/

public class Database implements Serializable{

    public static Database instance;

    private ArrayList<Mahasiswa> data = new ArrayList<>();

    private Database(){

    }

    public static synchronized Database getInstance(){

        loadFile();

        if(instance == null){

            instance = new Database();

        }

        return instance;

    }

    public void insertMahasiswa(Mahasiswa mahasiswa){

        data.add(mahasiswa);

        updateFile();

    }

    public List<Mahasiswa> getListMahasiswa(){

        return data;

    }

    private static void loadFile(){

        try {

            File f = new File("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\database.dat");

            FileInputStream fis = new FileInputStream(f);

            ObjectInputStream ois = new ObjectInputStream(fis);

            instance = (Database) ois.readObject();

            ois.close();

            fis.close();

        } catch (FileNotFoundException ex) {

            System.err.println("File tidak ditemukan");

        } catch (IOException ex) {

            System.err.println("File gagal dibaca");

        } catch (ClassNotFoundException ex) {

            System.err.println("Format File salah");

        }

    }

    private void updateFile(){

        try {

            FileOutputStream fos = new

            FileOutputStream("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\database.dat");

            ObjectOutputStream oos = new ObjectOutputStream(fos);

            oos.writeObject(instance);

            oos.close();

            fos.close();

        } catch (FileNotFoundException ex) {

            System.err.println("File tidak ditemukan");

        } catch (IOException ex) {

            System.err.println("Gagal menulis file");

        }

    }

}

Pada EntryPanel.java juga perlu ditambahkan loadTableData saat contruksi

    public EntryPanel() {

        initComponents();

        loadTableData();

    }

Kemudian run program tersebut dan entri satu data. Kemudian tutup window tersebut.

Ketika program dijalankan ulang maka data yang sebelumnya kita entri langsung ter-load pada tabel. Tidak hilang seperti sebelumnya.

