

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

MODUL 10 PRAKTIKUM PEMROGRAMAN BERORIENTASO OBJEK

(Input-Output)

A. Membaca File

Membuat 2 file .txt

textfile.txt
javaio.txt

Misal ingin membaca file **javaio.txt**

```
1 package latihan.file;
2
3 import java.io.File;
4
5 public class LatihanFile {
6     public static void main(String[] args) {
7         // TODO code application logic here
8         File f = new File("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\javaio.txt");
9         System.out.println(f.getName());
10        System.out.println(f.getParent());
11        System.out.println(f.canWrite());
12    }
13 }
```

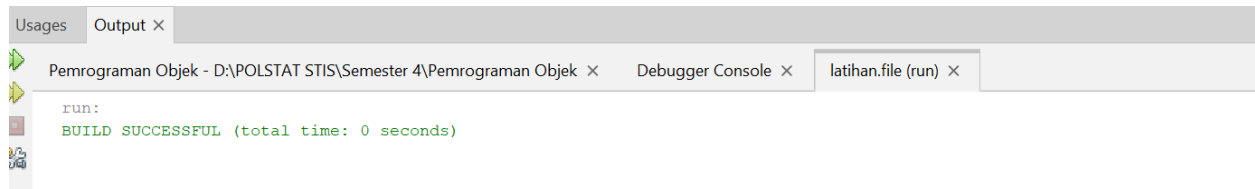
Usages Output X

Pemrograman Objek - D:\POLSTAT STIS\Semester 4\Pemrograman Objek x Debugger Console x latihan.file (run) x

run:
javaio.txt
D:\POLSTAT STIS\Semester 4\Pemrograman Objek\Pertemuan 10
true
BUILD SUCCESSFUL (total time: 0 seconds)

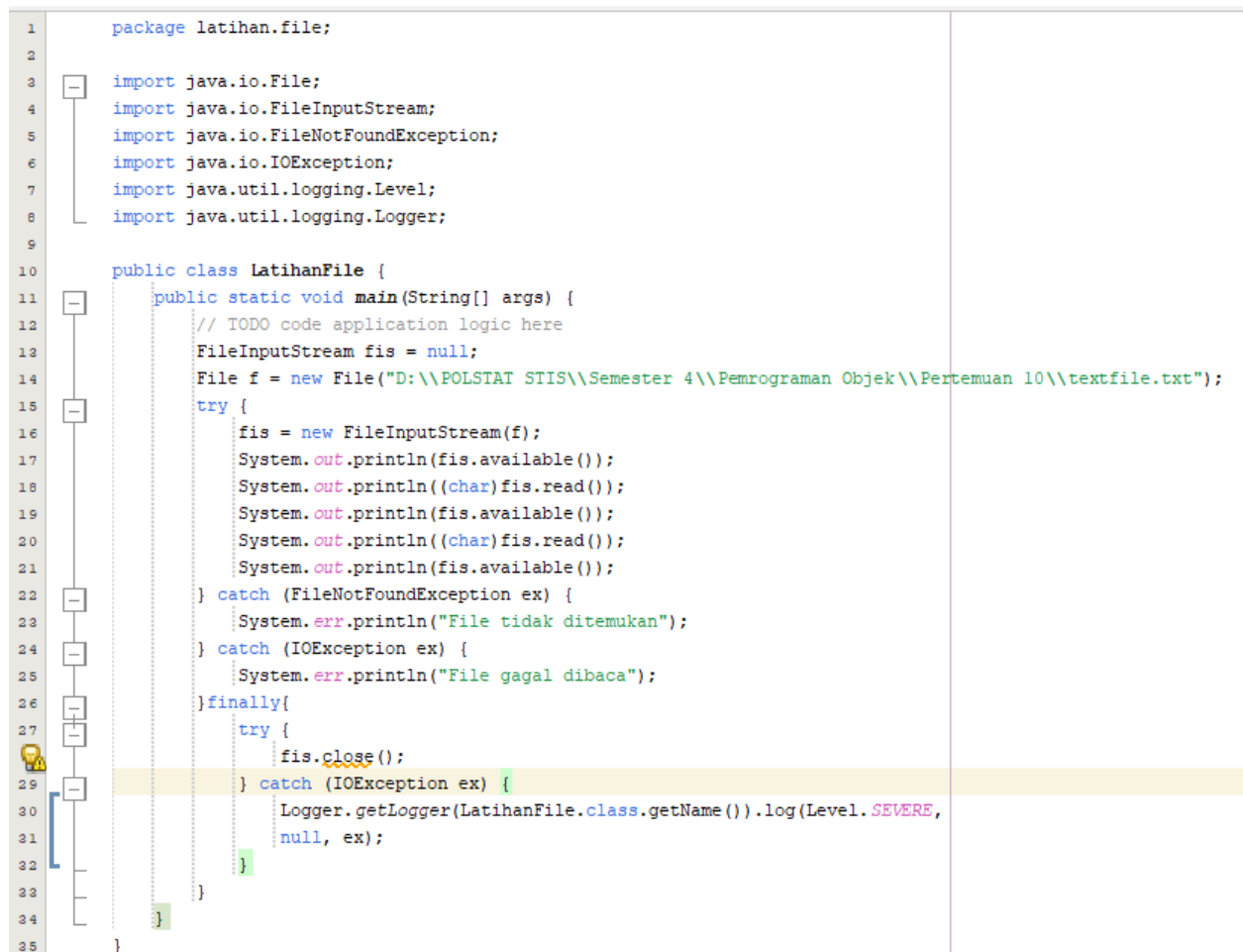
Kita juga bisa melakukan hal-hal yang biasa kita lakukan dengan explorer seperti rename file atau bahkan mendelete file. Misal rename file javaio.txt menjadi javaio2.txt

```
1 package latihan.file;
2
3 import java.io.File;
4
5 public class LatihanFile {
6     public static void main(String[] args) {
7         // TODO code application logic here
8         File f = new File("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\javaio.txt");
9         f.renameTo(new File("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\javaio2.txt"));
10        //f.delete();
11    }
12 }
```



Menjadi :  javaio2.txt

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



```
run:
9
T
8
e
7
BUILD SUCCESSFUL (total time: 0 seconds)
```

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

```
7      import java.util.logging.Level;
8      import java.util.logging.Logger;
9
10     public class LatihanFile {
11     public static void main(String[] args) {
12         // TODO code application logic here
13         FileInputStream fis = null;
14         File f = new File("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\textfile.txt");
15         try {
16             fis = new FileInputStream(f);
17             while(fis.available()>0){
18                 System.out.print((char) fis.read());
19             }
20         } catch (FileNotFoundException ex) {
21             System.err.println("File tidak ditemukan");
22         } catch (IOException ex) {
23             System.err.println("File gagal dibaca");
24         } finally{
25             try {
26                 fis.close();
27             } catch (IOException ex) {
28                 Logger.getLogger(LatihanFile.class.getName()).log(Level.SEVERE,
29                     null, ex);
30             }
31         }
32     }
33 }
```

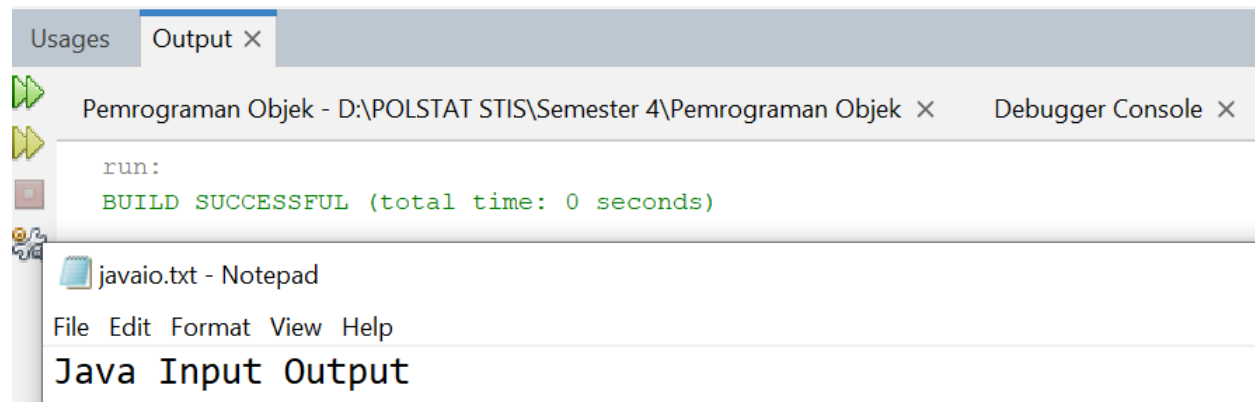
```
Pemrograman Objek - D:\POLSTAT STIS\Semester 4\Pemrograman Objek x Debugger Console
run:
Text FileBUILD SUCCESSFUL (total time: 0 seconds)
```

B. Menulis File

Untuk menulis file, perhatikan listing program di bawah ini:

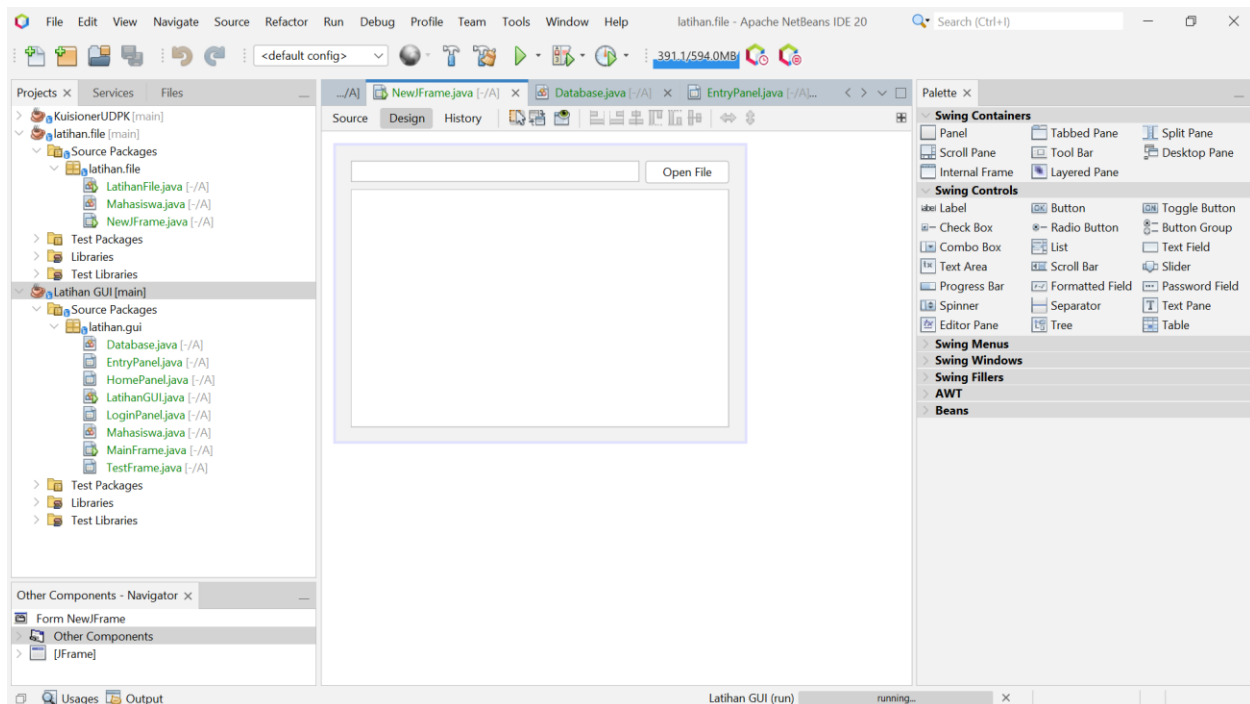
```
9      public class LatihanFile {
10      public static void main(String[] args) {
11      try {
12          FileOutputStream fos = new
13          FileOutputStream("D:\\POLSTAT STIS\\Semester 4\\Pemrograman Objek\\Pertemuan 10\\javaio.txt");
14          String str = "Java Input Output";
15          fos.write(str.getBytes());
16          fos.close();
17      } catch (FileNotFoundException ex) {
18          Logger.getLogger(LatihanFile.class.getName()).log(Level.SEVERE,
19              null, ex);
20      } catch (IOException ex) {
21          Logger.getLogger(LatihanFile.class.getName()).log(Level.SEVERE,null, ex);
22      }
23      }
24  }
```

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



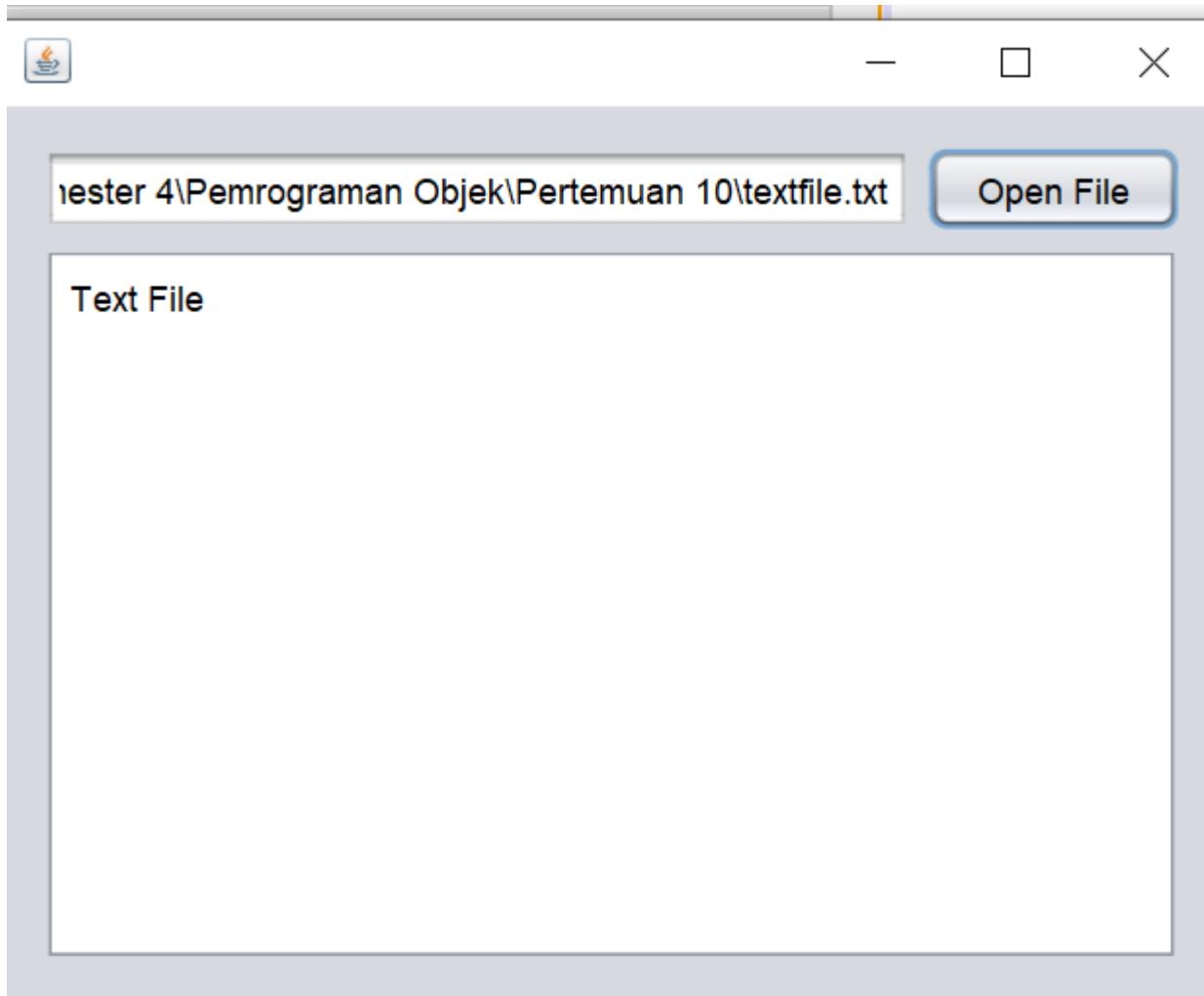
C. GUI

Buatlah rancangan GUI sebagai berikut:



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

Maka, ketika dijalankan file tersebut. Berikut adalah tampilan layarnya



MEMBACA DAN MENULIS OBJECT

Untuk membaca object, kita menggunakan kelas `ObjectOutputStream` dan untuk membaca object kita menggunakan kelas `ObjectInputStream`. Buka kembali project sebelumnya, kopikan kelas mahasiswa ke project yang sekarang. Kemudian kelas mahasiswa ini harus mengimplementasikan interface `serializable`.

```
package latihan.file;

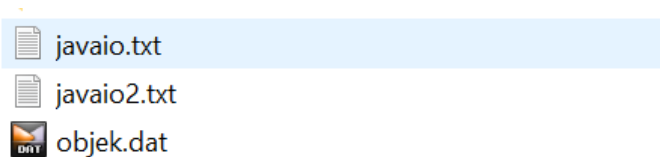
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.ObjectOutputStream;
import java.util.logging.Level;
import java.util.logging.Logger;
```

```

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.



D. 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:

```

run:
1234
Andi
BUILD SUCCESSFUL (total time: 0 seconds)

```

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.

The screenshot shows a Java Swing window titled "File Entri". The window contains a form on the left for entering student data and a table on the right displaying the stored data.

Form Fields:

- NIM :
- Nama :
- Jenis Kelamin : ☐ Laki-Laki ☒ Perempuan
- Umur : (with up/down arrows)
- Alamat :
- Provinsi : (with a dropdown arrow)
- Hobi : ☐ Membaca ☐ Menyanyi ☐ Berenang
-

Table Data:

NIM	Nama	Jenis Kelamin	Umur
2221212	Gazali	Laki-laki	15
212121	Hafis	Perempuan	15