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

**MODUL 9 PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK**

**(Graphical User Interface Bagian 2)**

**Penugasan**

Laporkan hasil praktikum berikut dengan hasil penugasan dan penjelasannya ke Dosen dalam bentuk file pdf dengan format nama <<nim>>\_modul9.

1. Lengkapi kode semua objek
2. Tangkapan layar hasil kode yang dilengkapi
3. Tangkapan layar hasil running

**Penyelesaian:**

Pada praktikum kali ini, diminta untuk melanjutkan praktikum yang ada pada modul 8 dan menambahkan beberapa file program baru, yaitu sebagai 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;

/\*\*

 \*

 \* @author U53R

 \*/

import java.util.ArrayList;

import java.util.List;

public class Database {

    public static Database instance;

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

    private Database(){

    }

    public static synchronized Database getInstance(){

        if(instance == null){

            instance = new Database();

        }

        return instance;

    }

    public void insertMahasiswa(Mahasiswa mahasiswa){

        data.add(mahasiswa);

    }

    public List<Mahasiswa> getListMahasiswa(){

        return data;

    }

}

**Mahasiswa.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;

/\*\*

 \*

 \* @author U53R

 \*/

import java.util.ArrayList;

public class Mahasiswa {

    private String nim,nama,jenisKelamin;

    private int umur;

    private String alamat, provinsi;

    private ArrayList<String> hobi;

    /\*\*

    \* @return the nim

    \*/

    public String getNim() {

        return nim;

    }

    /\*\*

    \* @param nim the nim to set

    \*/

    public void setNim(String nim) {

        this.nim = nim;

    }

    /\*\*

    \* @return the nama

    \*/

    public String getNama() {

        return nama;

    }

    /\*\*

    \* @param nama the nama to set

    \*/

    public void setNama(String nama) {

        this.nama = nama;

    }

    /\*\*

    \* @return the jenisKelamin

    \*/

    public String getJenisKelamin() {

        return jenisKelamin;

    }

    /\*\*

    \* @param jenisKelamin the jenisKelamin to set

    \*/

    public void setJenisKelamin(String jenisKelamin) {

        this.jenisKelamin = jenisKelamin;

    }

    /\*\*

     \* @return the umur

     \*/

    public int getUmur() {

        return umur;

    }

    /\*\*

    \* @param umur the umur to set

    \*/

    public void setUmur(int umur) {

        this.umur = umur;

    }

    /\*\*

    \* @return the alamat

    \*/

    public String getAlamat() {

        return alamat;

    }

    /\*\*

    \* @param alamat the alamat to set

    \*/

    public void setAlamat(String alamat) {

        this.alamat = alamat;

    }

    /\*\*

    \* @return the provinsi

    \*/

    public String getProvinsi() {

        return provinsi;

    }

    /\*\*

    \* @param provinsi the provinsi to set

    \*/

    public void setProvinsi(String provinsi) {

        this.provinsi = provinsi;

    }

    /\*\*

    \* @return the hobi

    \*/

    public ArrayList<String> getHobi() {

        return hobi;

    }

    /\*\*

    \* @param hobi the hobi to set

    \*/

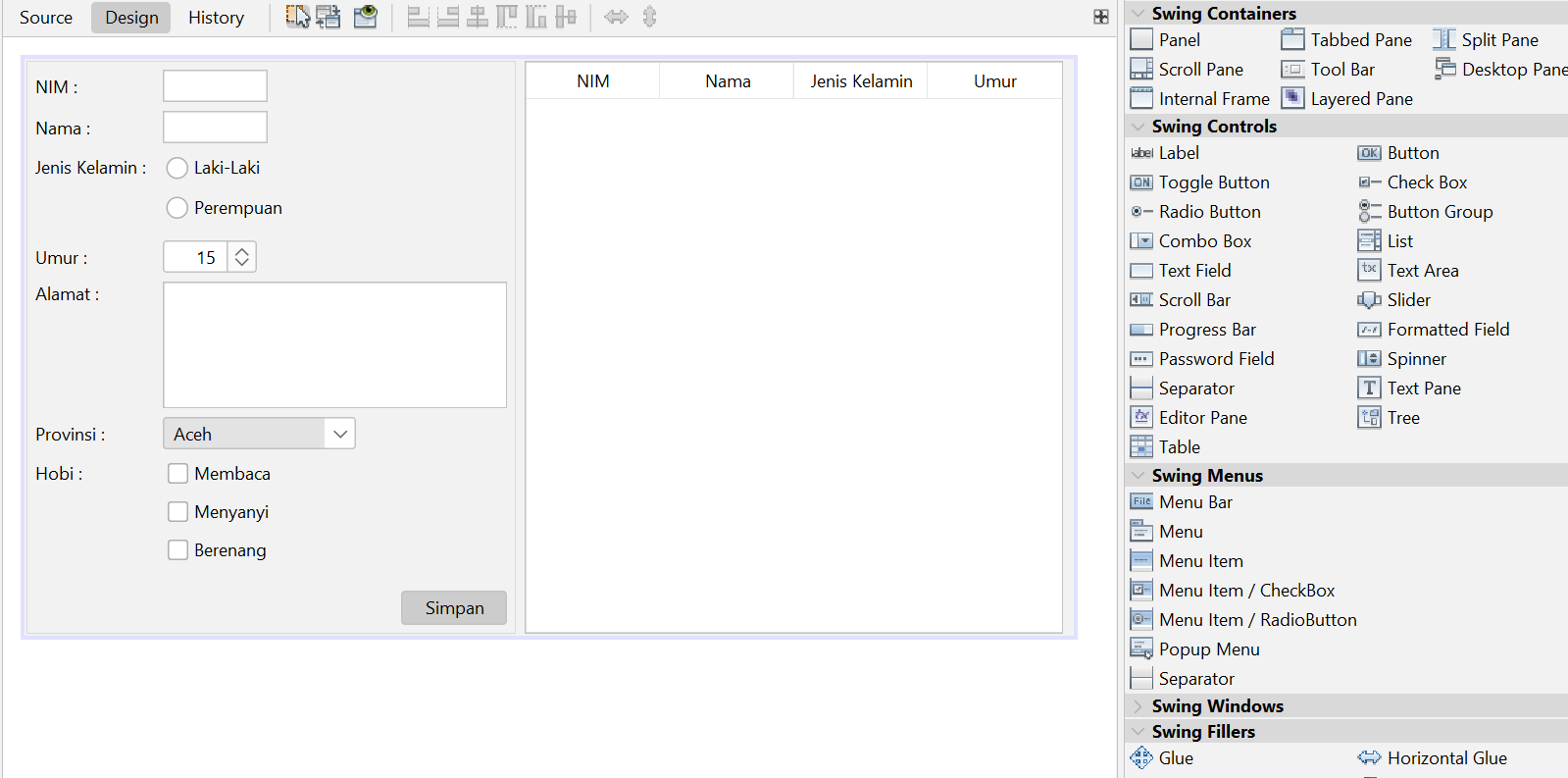
    public void setHobi(ArrayList<String> hobi) {

        this.hobi = hobi;

    }

}

Kemudian, memodifikasi **MainFrame.java** dengan menambahkan JTable baru yang terdiri dari kolom NIM, Nama, Jenis Kelamin, dan Umur seperti berikut ini:



Untuk mengkonsistenkan tampilan tabel dan data source, maka setelah melakukan simpan data, kita perlu melakukan refresh tabel tersebut. Untuk melakukan refresh atau load ulang tambahkan method berikut di source:

private void loadTableData(){

        DefaultTableModel dtm = (DefaultTableModel) mahasiswaTable.getModel();

        //refresh tabel

        while(dtm.getRowCount()>0){

            dtm.removeRow(0);

        }

        //isi tabel

        for(Mahasiswa mhs:

        Database.getInstance().getListMahasiswa()){

            dtm.addRow(new

            Object[]{mhs.getNim(),mhs.getNama(),mhs.getJenisKelamin(),mhs.getUmur()});

        }

    }

    private void clearForm(){

        nimTextField.setText("");

        namaTextField.setText("");

        //buttonGroup1.clearSelection();

        umurSpinner.setValue(15);

        alamatTextArea.setText("");

        provinsiComboBox.setSelectedIndex(-1);

        membacaCheckBox.setSelected(false);

        menyanyiCheckBox.setSelected(false);

        berenangCheckBox.setSelected(false);

    }

Kemudian untuk menambahkan jendela notifikasi, ubah kode simpanButtonActionPerformed pada **MainFrame.java** menjadi berikut ini:

private void simpanButtonActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_simpanButtonActionPerformed

        // TODO add your handling code here:

        Mahasiswa mhs = new Mahasiswa();

        mhs.setNim(nimTextField.getText());

        mhs.setNama(namaTextField.getText());

        if(lakiRadioButton.isSelected()){

            mhs.setJenisKelamin("Laki-laki");

        }

        if(perempuanRadioButton.isSelected()){

            mhs.setJenisKelamin("Perempuan");

        }

        int umur = (Integer) umurSpinner.getValue();

        mhs.setUmur(umur);

        mhs.setAlamat(alamatTextArea.getText());

        mhs.setProvinsi(provinsiComboBox.getSelectedItem().toString());

        ArrayList<String> hobiList = new ArrayList<>();

        if(membacaCheckBox.isSelected()){

            hobiList.add("membaca");

        }

        if(menyanyiCheckBox.isSelected()){

            hobiList.add("menyanyi");

        }

        if(berenangCheckBox.isSelected()){

            hobiList.add("berenang");

        }

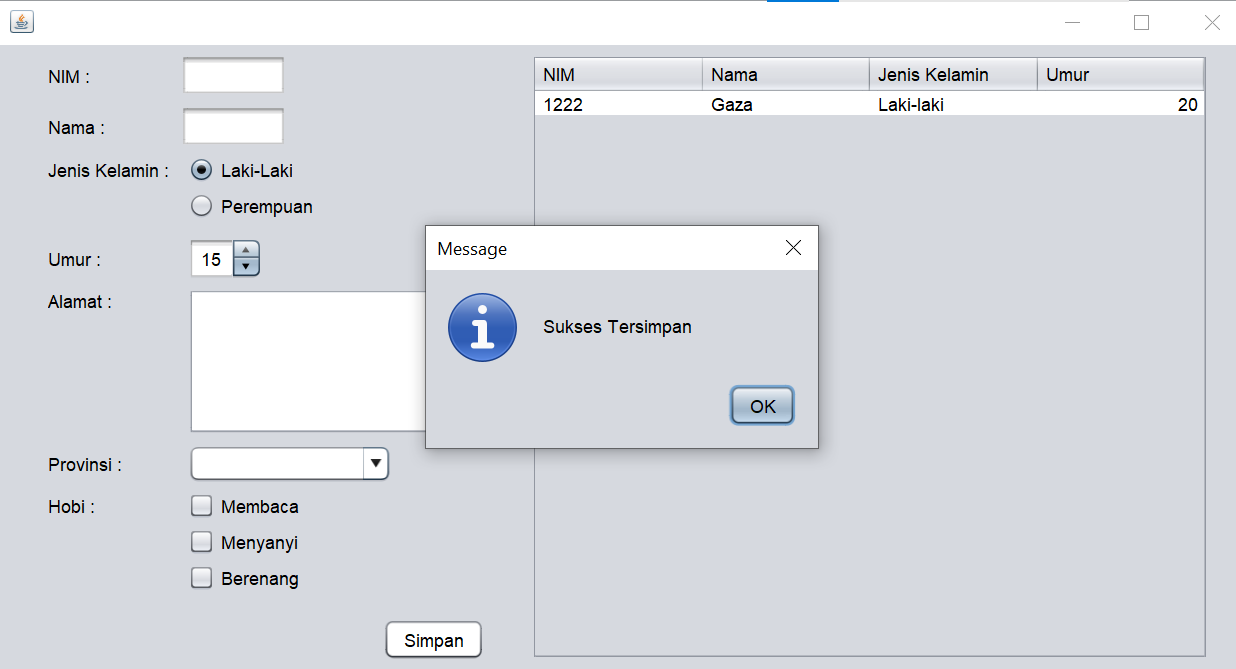
        Database.getInstance().insertMahasiswa(mhs);

        clearForm();

        JOptionPane.showMessageDialog(this, "Sukses Tersimpan");

        loadTableData();

    }//GEN-LAST:event\_simpanButtonActionPerformed

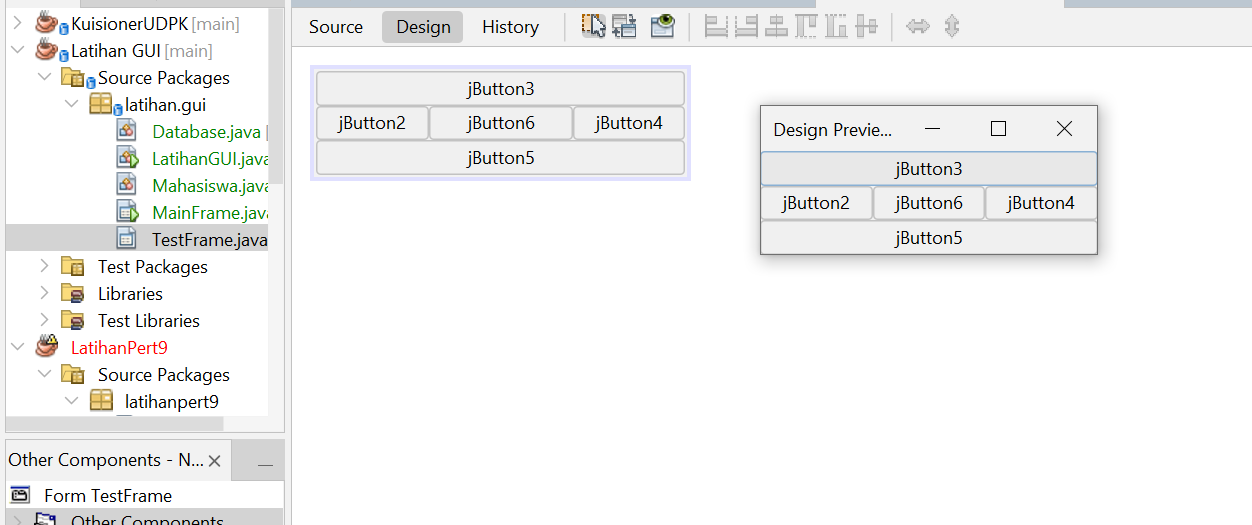
Ketika di-run maka akan muncul jendela sukses tersimpan

Pada pembahasan berikutnya kita akan membahas tentang layout pada Netbeans. Terdapat berbagai macam layout seperti BorderLayout, BoxLayout, dan lan-lain.

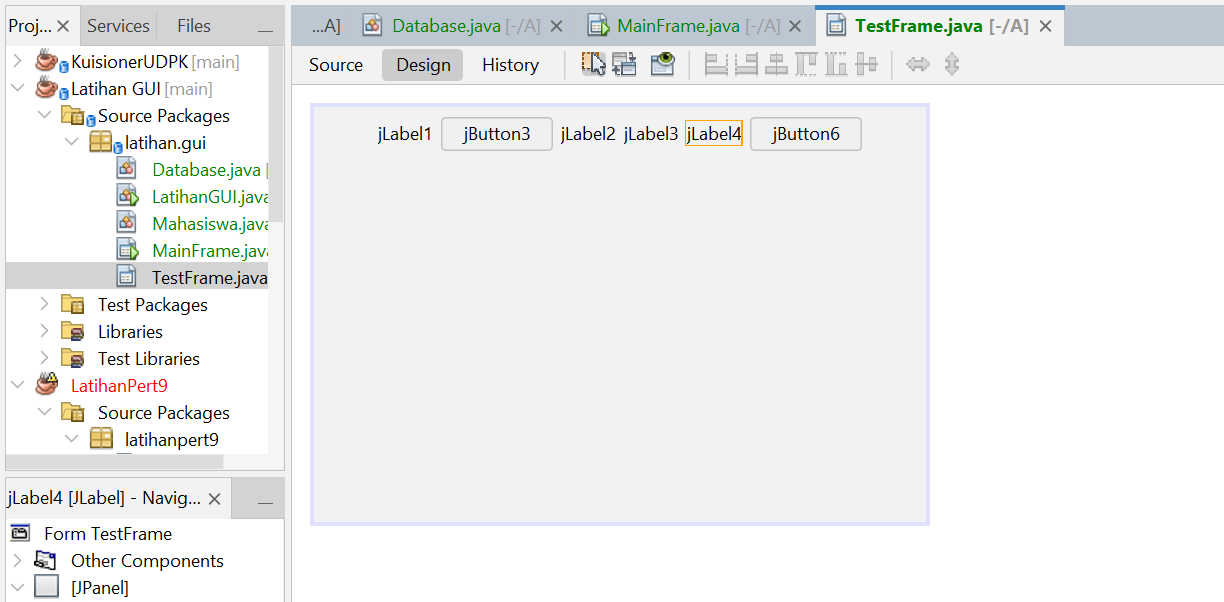
Pengaturan layout ini penting suaya kita bisa mengatur perilaku window ketika dikecilkan, dibesarkan sehingga window tetap berperilaku sesuai dengan yang kita inginkan.

Tambahkan frame baru pada project dengan nama TestFrame

**Border Layout**



**Flow Layout**



Misal untuk program entri mahasiswa tadi akan kita buat menjadi 2 panel yaitu panel entri dan panel untuk menampilkan data. Pada package latihan.gui tambahkan new Jpanel form bernama **EntryPanel.**

* Copy paste semua elemen entri pada MainFrame ke panel tersebut.
* Kemudian tampilan pada MainFrame dapat kita bersihkan. Drag semua komponen kemudian delete
* Kemudian tambahkan JScrollPane pada MainFrame
* Kemudin edit method constructornya menjadi seprti berikut:

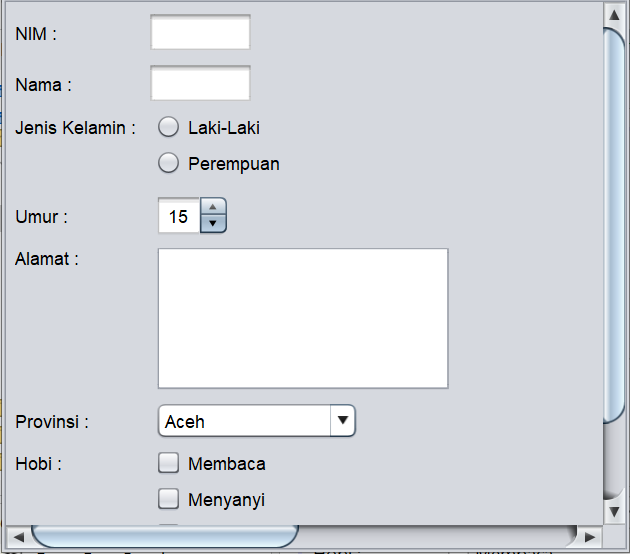
 public MainFrame() {

     initComponents();

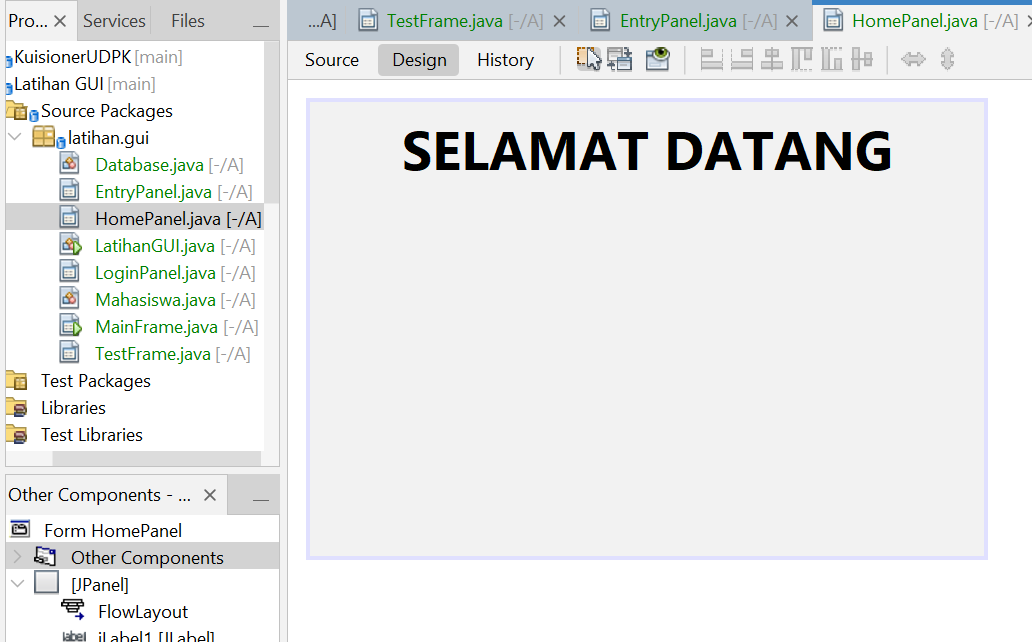
     contentScrollPane.setViewportView(new EntryPanel());

}

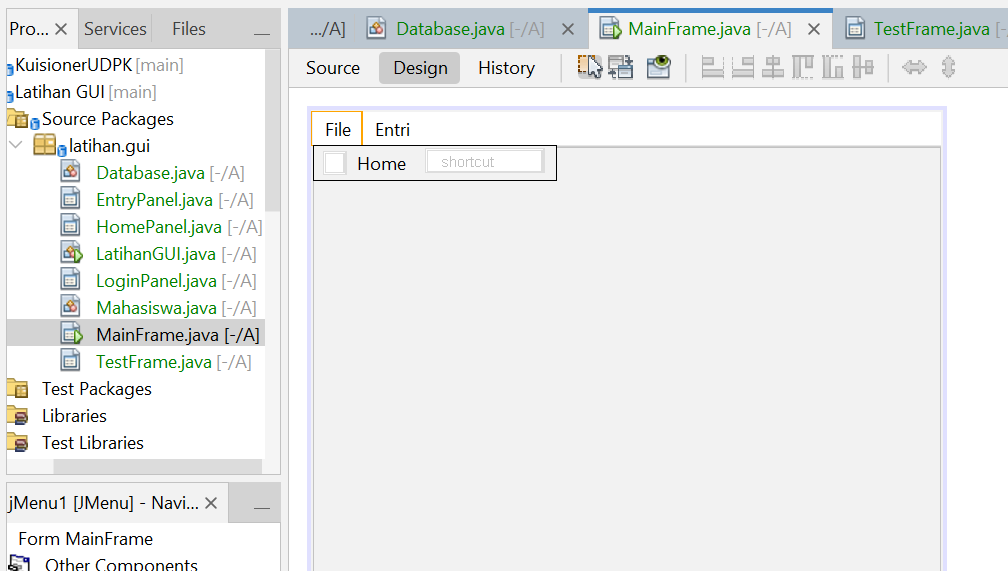
* Ketika dijalankan maka kita bisa melakukan resizing pada jendela frame.



Misal kita tambahkan HomePanel dengan ucapan selamat datang



Kemudian pada MainFrame kita tambahkan Jmenu. Tambahkan JmenuItem untuk File-> Home dan Entri-> Mahasiswa



Kemudian modifikasi action homeMenuItem dan entriMenuItem dengan menambahkan kode berikut

    private void homeMenuItemActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_homeMenuItemActionPerformed

        // TODO add your handling code here:

        contentScrollPane.setViewportView(new HomePanel());

    }//GEN-LAST:event\_homeMenuItemActionPerformed

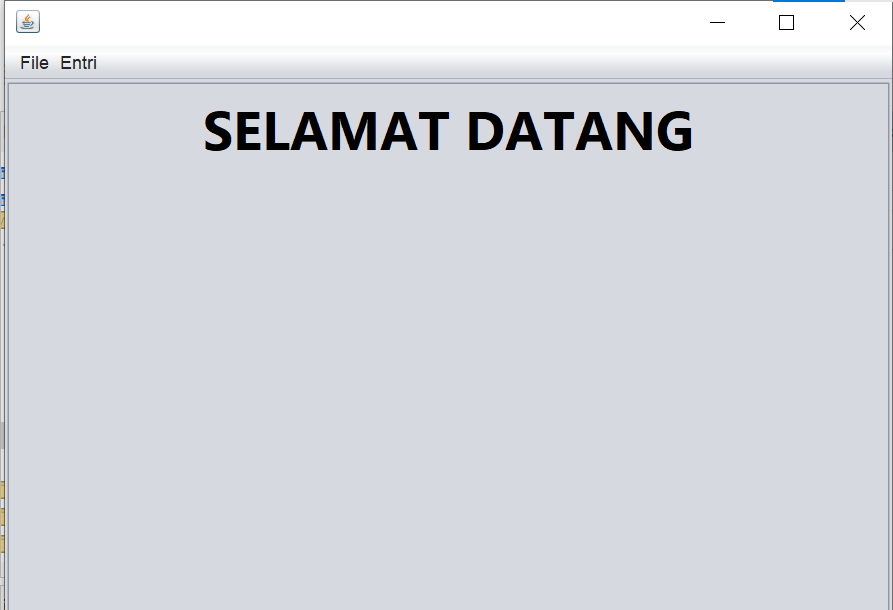
    private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jMenuItem2ActionPerformed

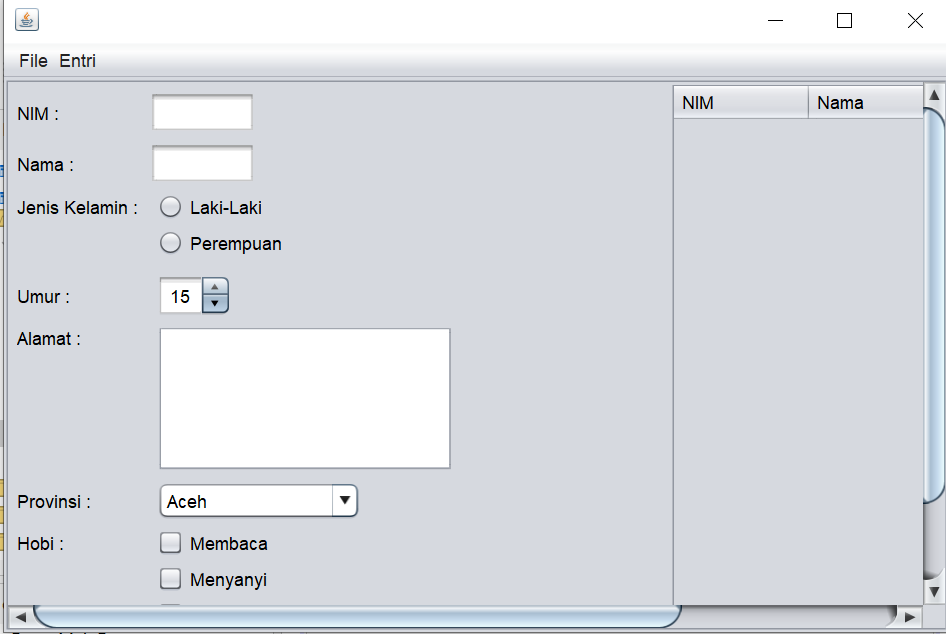
        // TODO add your handling code here:

        contentScrollPane.setViewportView(new EntryPanel());

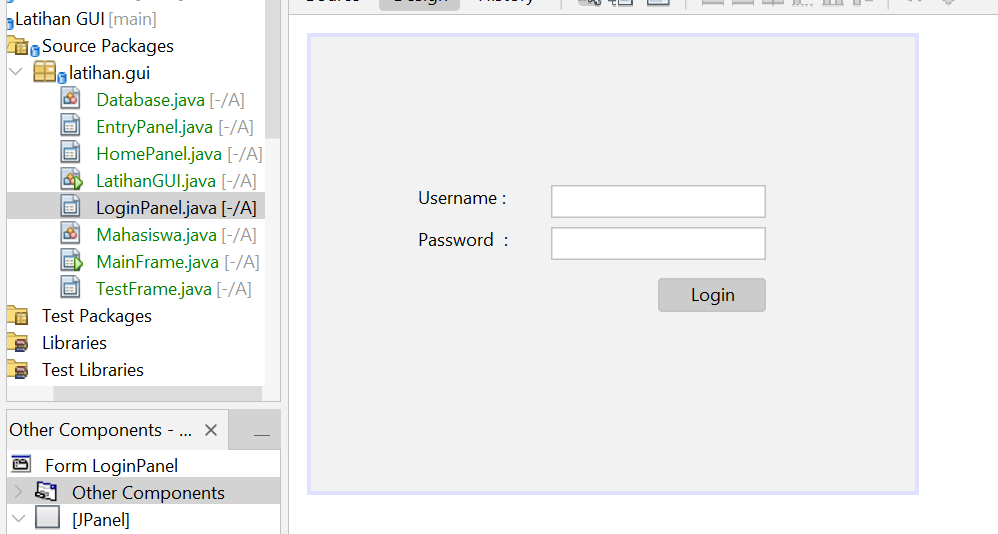
    }//GEN-LAST:event\_jMenuItem2ActionPerformed

Sehingga tampilan panel pada frame dapat diganti-ganti





Misal satu lagi kita tambahkan login panel



Kemudian tambahkan kode berikut pada awal source **LoginPanel**

private JScrollPane contentScrollPane;

        /\*\*

        \* Creates new form LoginPanel

        \*/

    public LoginPanel(JScrollPane contentScrollPane) {

        this.contentScrollPane = contentScrollPane;

        initComponents();

    }

Ubah method constructor pada MainFrame.java menjadi berikut ini:

    public MainFrame() {

        initComponents();

        contentScrollPane.setViewportView(new LoginPanel(contentScrollPane));

    }

Jalankan program, jika kita klik tombol login maka akan muncul tampilan home panel

