**TUGAS PRA PERTEMUAN 11**

**OBJECT ORIENTED PROGRAMMING**

Dosen : Ibnu Santoso, SST, M.T.



Disusun Oleh :

Nama : Marsay Febrianto

NIM : (222112177)

Kelas : 2KS2

No : 23

Prodi : DIV Komputasi Statistik

**POLITEKNIK STATISTIKA STIS**

Jl. Otto Iskandardinata No. 64C, Jakarta 13330

Telp. (021) 8508812, 8191473, Fax. 8197577

Website : [www.stis.ac.id](http://www.stis.ac.id/), E-Mail : info@stis.ac.id

**Tahun Ajaran 2022/2023**

1. Class LoginPanel

import Data.koneksi;

import java.sql.Connection;

import java.sql.DriverManager;

import javax.swing.JOptionPane;

import javax.swing.JScrollPane;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

public class LoginPanel extends javax.swing.JPanel {

    private JScrollPane contentScrollPane;

    /\*\*

     \* Creates new form LoginPanel

     \*/

    public LoginPanel(JScrollPane contentScrollPane) {

        this.contentScrollPane = contentScrollPane;

        initComponents();

    }

    /\*\*

     \* This method is called from within the constructor to initialize the form.

     \* WARNING: Do NOT modify this code. The content of this method is always

     \* regenerated by the Form Editor.

     \*/

    @SuppressWarnings("unchecked")

    // <editor-fold defaultstate="collapsed" desc="Generated Code">

    private void initComponents() {

       …………………

    private void loginActionPerformed(java.awt.event.ActionEvent evt) {

        // TODO add your handling code here:

        try {

            koneksi conn = new koneksi();

            Connection con = conn.getCon();

            String sql = "Select \* from logindatabase where username=? and password=?";

            PreparedStatement pst = con.prepareStatement(sql);

            pst.setString(1, username.getText());

            pst.setString(2, password.getText());

            ResultSet rs = pst.executeQuery();

            if (rs.next()) {

                JOptionPane.showMessageDialog(null,"Hello "+username.getText());

                contentScrollPane.setViewportView(new HomePanel());

            } else {

                JOptionPane.showMessageDialog(null,"Username / Password wrong");

                username.setText("");

                password.setText("");

            }

        } catch(Exception e) {

            JOptionPane.showMessageDialog(null,e);

        }

    }

    // Variables declaration - do not modify

    private javax.swing.JButton daftar;

    private javax.swing.JLabel jLabel1;

    private javax.swing.JLabel jLabel2;

    private javax.swing.JLabel lbl\_password;

    private javax.swing.JLabel lbl\_username;

    private javax.swing.JButton login;

    private javax.swing.JPasswordField password;

    private javax.swing.JTextField username;

    // End of variables declaration

}

1. Class Entry Panel

import Services.Database;

import Data.Mahasiswa;

import Data.koneksi;

import EditCell.TableActionCellEditor;

import EditCell.TableActionCellRender;

import EditCell.TableActionEvent;

import java.awt.event.ActionListener;

import java.beans.PropertyChangeEvent;

import java.beans.PropertyChangeListener;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.util.ArrayList;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

import javax.swing.event.ChangeListener;

import javax.swing.table.DefaultTableModel;

/\*

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

 \* Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JPanel.java to edit this template

 \*/

/\*\*

 \*

 \* @author Second

 \*/

public class EntryPanel extends javax.swing.JPanel {

    /\*\*

     \* Creates new form EntryPanel

     \*/

    public EntryPanel() {

        initComponents();

        loadTableData();

        edit.setVisible(false);

        TableActionEvent event = new TableActionEvent() {

            @Override

            public void onEdit(int row) {

                nim.enableInputMethods(false);

                edit.setVisible(true);

                simpan.setVisible(false);

                System.out.println("Edit Row : " + row);

                Mahasiswa mhs = Database.getInstance().getListMahasiswa().get(row);

                nim.setText(mhs.getNim());

                nama.setText(mhs.getNama());

                umur.setValue(mhs.getUmur());

                alamat.setText(mhs.getAlamat());

                provinsi.setSelectedItem(mhs.getProvinsi());

            }

            @Override

            public void onDelete(int row) {

                if (mahasiswaTable.isEditing()) {

                    mahasiswaTable.getCellEditor().stopCellEditing();

                }

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

                System.out.println("delete Row : " + row + (String) model.getValueAt(row, 1));

                try {

                    koneksi con = new koneksi();

                    String sql = "DELETE FROM mahasiswa WHERE nim = ?";

                    String nim = (String) model.getValueAt(row, 0);

                    PreparedStatement pst = con.getCon().prepareStatement(sql);

                    pst.setString(1, nim);

                    int rowsInserted = pst.executeUpdate();

                    if (rowsInserted >0) {

                        JOptionPane.showMessageDialog(null,nim+" telah dihapus");

                    } else {

                        JOptionPane.showMessageDialog(null,"nim tidak ada");

                    }

                    pst.close();

                    con.getCon().close();

                } catch (Exception ex) {

                    System.out.println(ex);

                }

                model.removeRow(row);

            }

        };

        mahasiswaTable.getColumnModel().getColumn(4).setCellRenderer(new TableActionCellRender());

        mahasiswaTable.getColumnModel().getColumn(4).setCellEditor(new TableActionCellEditor(event));

        loadTableData();

    }

    /\*\*

     \* This method is called from within the constructor to initialize the form.

     \* WARNING: Do NOT modify this code. The content of this method is always

     \* regenerated by the Form Editor.

     \*/

    @SuppressWarnings("unchecked")

    // <editor-fold defaultstate="collapsed" desc="Generated Code">

    private void initComponents() {

……………………

}

    private void editActionPerformed1(java.awt.event.ActionEvent evt) {

        // TODO add your handling code here:

        Mahasiswa mhs = getField();

        Database.getInstance().editMahasiswa(mhs);

        nim.enableInputMethods(true);

        edit.setVisible(false);

        simpan.setVisible(true);

        clearForm();

        loadTableData();

    }

    private Mahasiswa getField() {

        Mahasiswa mhs = new Mahasiswa();

        mhs.setNim(nim.getText());

        mhs.setNama(nama.getText());

        if(laki.isSelected()){

            mhs.setJenisKelamin("Laki-laki");

        }

        if(perempuan.isSelected()){

            mhs.setJenisKelamin("Perempuan");

        }

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

        mhs.setUmur(umurs);

        mhs.setAlamat(alamat.getText());

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

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

        if(membaca.isSelected()){

            hobiList.add("membaca");

        }

        if(menyanyi.isSelected()){

            hobiList.add("menyanyi");

        }

        if(berenang.isSelected()){

            hobiList.add("berenang");

        }

        return mhs;

    }

     private void loadTableData() {

        DefaultTableModel dtm = (DefaultTableModel)

                mahasiswaTable.getModel();

        //refresh tabel

        dtm.setRowCount(0);

        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() {

        nim.setText("");

        nama.setText("");

        umur.setValue(15);

        alamat.setText("");

        provinsi.setSelectedIndex(-1);

        menyanyi.setSelected(false);

        menyanyi.setSelected(false);

        berenang.setSelected(false);

    }

    // Variables declaration - do not modify

    private javax.swing.JTextArea alamat;

    private javax.swing.JCheckBox berenang;

    private javax.swing.JButton edit;

    private javax.swing.JLabel jLabel1;

    private javax.swing.JLabel jLabel2;

    private javax.swing.JLabel jLabel3;

    private javax.swing.JLabel jLabel4;

    private javax.swing.JLabel jLabel5;

    private javax.swing.JLabel jLabel7;

    private javax.swing.JLabel jLabel8;

    private javax.swing.JPanel jPanel1;

    private javax.swing.JPanel jPanel2;

    private javax.swing.JScrollPane jScrollPane1;

    private javax.swing.JScrollPane jScrollPane2;

    private javax.swing.JRadioButton laki;

    private javax.swing.JTable mahasiswaTable;

    private javax.swing.JCheckBox membaca;

    private javax.swing.JCheckBox menyanyi;

    private javax.swing.JTextField nama;

    private javax.swing.JTextField nim;

    private javax.swing.JRadioButton perempuan;

    private javax.swing.JComboBox<String> provinsi;

    private javax.swing.JButton simpan;

    private javax.swing.JSpinner umur;

    // End of variables declaration

}

1. Class Koneksi database

/\*

 \* 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 Data;

import java.sql.Connection;

import java.sql.DriverManager;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

 \*

 \* @author Second

 \*/

public class koneksi {

    private Connection con;

    public koneksi() throws Exception{

        Class.forName("com.mysql.cj.jdbc.Driver");

        con = DriverManager.getConnection("jdbc:mysql://localhost:3306/pbo\_t9","marsay","");

//        System.out.println("Database Terkoneksi");

    }

    public Connection getCon() {

        return con;

    }

}

1. Class database

package Services;

import Data.Mahasiswa;

import Data.koneksi;

import java.beans.Statement;

import java.io.FileInputStream;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.util.ArrayList;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JFileChooser;

import javax.swing.JOptionPane;

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 static void loadFile() {

        JFileChooser fileChooser = new JFileChooser();

        if (fileChooser.showOpenDialog(null)==JFileChooser.APPROVE\_OPTION) {

            try {

                FileInputStream fis = new FileInputStream(fileChooser.getSelectedFile().toString());

                ObjectInputStream ois = new ObjectInputStream(fis);

                instance = (Database) ois.readObject();

                ois.close();

                fis.close();

            } catch (IOException ex) {

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

                System.err.println("gagal membaca file");

            } catch (ClassNotFoundException ex) {

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

                System.err.println("class tidak ditemukan/format file salah");

            }

        }

    }

    public void editMahasiswa(Mahasiswa m) {

        try {

            koneksi con = new koneksi();

            Connection conn = con.getCon();

            System.out.println("Database Terkoneksi");

            String sql = "UPDATE mahasiswa SET nama=?, jk=?, umur=?, alamat=?, provinsi=? WHERE nim=?";

            // Mempersiapkan pernyataan SQL dengan parameter tanda tanya (?)

            PreparedStatement stmt = conn.prepareStatement(sql);

            stmt.setString(1, m.getNama());

            stmt.setString(2, m.getJenisKelamin());

            stmt.setInt(3, m.getUmur());

            stmt.setString(4, m.getAlamat());

            stmt.setString(5, m.getProvinsi());

            stmt.setString(6, m.getNim());

            // Menjalankan pernyataan SQL untuk melakukan update

            int rowsUpdated = stmt.executeUpdate();

            if (rowsUpdated > 0) {

                System.out.println("Data mahasiswa berhasil diupdate.");

            } else {

                System.out.println("Gagal mengupdate data mahasiswa.");

            }

            // Menutup pernyataan dan koneksi

            stmt.close();

            conn.close();

        } catch (Exception ex) {

            System.err.println("Gagal connect ke database");

        }

    }

    public void insertMahasiswa(Mahasiswa m){

        try {

//            Class.forName("com.mysql.cj.jdbc.Driver");

//            Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/pbo\_t9","marsay","");

            koneksi con = new koneksi();

            String sql = "INSERT INTO mahasiswa (nim, nama, jk, umur, alamat, provinsi) VALUES (?,?,?,?,?,?)";

            PreparedStatement stmt = con.getCon().prepareStatement(sql);

            stmt.setString(1, m.getNim());

            stmt.setString(2, m.getNama());

            stmt.setString(3, m.getJenisKelamin());

            stmt.setInt(4, m.getUmur());

            stmt.setString(5, m.getAlamat());

            stmt.setString(6, m.getProvinsi());

            int rowsInserted = stmt.executeUpdate();

            if (rowsInserted >0) {

                JOptionPane.showMessageDialog(null,m.getNama()+" telah ditambahkan");

            } else {

                JOptionPane.showMessageDialog(null,"Isian tidak valid");

            }

            stmt.close();

            con.getCon().close();

        } catch(Exception e) {

            JOptionPane.showMessageDialog(null,e);

        }

    }

    public List<Mahasiswa> getListMahasiswa(){

        try {

            data = new ArrayList<>();

            koneksi conn = new koneksi();

//            Class.forName("com.mysql.cj.jdbc.Driver");

//            Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/pbo\_t9","marsay","");

            String sql = "SELECT \* FROM mahasiswa";

            PreparedStatement statement = conn.getCon().prepareStatement(sql);

            // Menjalankan pernyataan SQL

            ResultSet resultSet = statement.executeQuery();

            while (resultSet.next()) {

                Mahasiswa mahasiswa = new Mahasiswa();

                mahasiswa.setNim(resultSet.getString(1));

                mahasiswa.setNama(resultSet.getString(2));

                mahasiswa.setJenisKelamin(resultSet.getString(3));

                mahasiswa.setUmur(resultSet.getInt(4));

                mahasiswa.setAlamat(resultSet.getString(5));

                mahasiswa.setProvinsi(resultSet.getString(6));

                data.add(mahasiswa);

            }

            resultSet.close();

            statement.close();

            conn.getCon().close();

        } catch (Exception e) {

            e.printStackTrace();

        }

        return data;

    }

}