

LAMPIRAN KODING PROGRAM

```
package iinterface;
import javax.swing.ImageIcon;
import java.sql.ResultSet;
import javax.swing.JOptionPane;
import java.sql.Connection;
import java.sql.SQLException;
import java.sql.Statement;
import java.sql.*;
import javax.swing.table.DefaultTableModel;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import koneksi.database;
import java.text.SimpleDateFormat;
import javax.swing.*;
import java.sql.DriverManager;
import java.sql.ResultSet;
import javax.swing.JOptionPane;
import java.sql.Connection;
import java.sql.SQLException;
import java.sql.Statement;
import java.sql.*;
import javax.swing.table.DefaultTableModel;
import javax.swing.*;
import java.text.SimpleDateFormat;
import java.sql.SQLException;
import java.sql.Statement;
import java.awt.*;
```

```

import java.awt.event.*;
import javax.swing.table.TableRowSorter;
/**
 *
 * @author User
 */
public class form_mahasiswa extends javax.swing.JInternalFrame {
String data[]=new String[8];
database dbsetting;
String driver, dtbase, user,pass,userLogin;
    private Connection con;
    private Statement st;
    String Tanggal1;
    ImageIcon imageIcon;
    /** Creates new form form_mahasiswa */
public form_mahasiswa() {
    initComponents();
    dbsetting = new database();
    driver = dbsetting.SettingPanel("DBDriver");
    dtbase = dbsetting.SettingPanel("DBDatabase");
    user = dbsetting.SettingPanel("DBUsername");
    pass = dbsetting.SettingPanel("DBPassword");
    jTable1.setModel(tableModel);
    Table(jTable1, new int[]{ 100,200,100,100,100,300,100,100});
    setDefaultTable();
    tampil_jurusan();
Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
Dimension frameSize = getSize();
setLocation(
(screenSize.width - frameSize.width) / 3,
(screenSize.height - frameSize.height) / 5);
    }

```

```
private void bUTTON_FORM5ActionPerformed(java.awt.event.ActionEvent evt)
{
    this.dispose();
}
```

```
private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    if (jRadioButton1.isSelected()== true){
        jRadioButton2.setSelected(false);
        jRadioButton2.setEnabled(false);
    }else if (jRadioButton1.isSelected()== false){
        jRadioButton2.setSelected(false);
        jRadioButton2.setEnabled(true);
    }    // TODO add your handling code here:
}
```

```
private void jRadioButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    if (jRadioButton2.isSelected()== true){
        jRadioButton1.setSelected(false);
        jRadioButton1.setEnabled(false);
    }else if (jRadioButton2.isSelected()== false){
        jRadioButton1.setSelected(false);
        jRadioButton1.setEnabled(true);
    }    // TODO add your handling code here:
}
```

```
private void tanggal_lahirPropertyChange(java.beans.PropertyChangeEvent evt) {
    if(tanggal_lahir.getDate()!=null){
        SimpleDateFormat Format = new SimpleDateFormat("yyyy-MM-dd");
        Tanggal1=Format.format(tanggal_lahir.getDate());
    }
}
```

```
private void jComboBox1ItemStateChanged(java.awt.event.ItemEvent evt) {
    tampil_hasil_jurusan1();
}

private void jTextField3CaretUpdate(javax.swing.event.CaretEvent evt) {
    tampil_hasil_jurusan2();
}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {
    Tampil_tabel();
    Tampil();
}

private void BUTTON_FORM1ActionPerformed(java.awt.event.ActionEvent evt)
{
    bersih();
}

private void BUTTON_FORM2ActionPerformed(java.awt.event.ActionEvent evt)
{
    simpan_data(); // TODO add your handling code here:
}

private void BUTTON_FORM3ActionPerformed(java.awt.event.ActionEvent evt)
{
    ubah_data(); // TODO add your handling code here:
}

private void BUTTON_FORM4ActionPerformed(java.awt.event.ActionEvent evt)
{
    hapus_data(); // TODO add your handling code here:
}

private void jTextField5KeyTyped(java.awt.event.KeyEvent evt) {
```

```

        if(jTextField5.getText().length()>11)
            evt.consume();
            FilterHanyaAngka(evt);    // TODO add your handling code here:
    }

private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {
    if(jTextField1.getText().length()>9)
        evt.consume();
    FilterHanyaAngka(evt);// TODO add your handling code here:
}

private void jTextField2KeyTyped(java.awt.event.KeyEvent evt) {
    if(jTextField2.getText().length()>39)
        evt.consume();
        filterhanyahurut(evt);
}

private void jTextField7CaretUpdate(javax.swing.event.CaretEvent evt) {
    TableRowSorter sorter = new TableRowSorter(jTable1.getModel());
    jTable1.setRowSorter(sorter);
    sorter.setRowFilter(RowFilter.regexFilter("(?i)" + jTextField7.getText()));
    jTable1.setSelectionMode(1);    // TODO add your handling code here:
}

void filterhanyahurut (KeyEvent b){
    if (Character.isDigit(b.getKeyChar())) {
        b.consume();
        JOptionPane.showMessageDialog(null, "Maaf Data Yang Anda Inputkan Harus Huruf");
    }
}

public void FilterHanyaAngka(java.awt.event.KeyEvent evt){
    char c = evt.getKeyChar();

```

```
        if (!(c>='0' && c<='9' || (c == KeyEvent.VK_BACK_SPACE) || (c ==
KeyEvent.VK_DELETE)))){
            evt.consume();
OptionPane.showMessageDialog(null, "Maaf Data Yang Anda Inputkan Harus
Angka");
        }
    }
// Variables declaration - do not modify
private class_java.BUTTON_FORM bUTTON_FORM1;
private class_java.BUTTON_FORM bUTTON_FORM2;
private class_java.BUTTON_FORM bUTTON_FORM3;
private class_java.BUTTON_FORM bUTTON_FORM4;
private class_java.BUTTON_FORM bUTTON_FORM5;
private javax.swing.JComboBox jComboBox1;
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 jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JRadioButton jRadioButton1;
private javax.swing.JRadioButton jRadioButton2;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
private javax.swing.JTextField jTextField5;
```

```

private javax.swing.JTextField jTextField6;
private javax.swing.JTextField jTextField7;
private com.toedter.calendar.JDateChooser tanggal_lahir;
// End of variables declaration
int row=0;
private javax.swing.table.DefaultTableModel
tableModel=getDefaultTableModel();
private void Table(javax.swing.JTable tb, int lebar[] ) {
tb.setAutoResizeMode(JTable.AUTO_RESIZE_OFF);
int kolom=tb.getColumnCount();
for(int i=0;i < kolom;i++) {
javax.swing.table.TableColumn tbc=tb.getColumnModel().getColumn(i);
tbc.setPreferredWidth(lebar[i]);
tb.setRowHeight(17);
}}
private javax.swing.table.DefaultTableModel getDefaultTableModel() {
return new javax.swing.table.DefaultTableModel(
new Object[][] {},
new String [] {"NPM","Nama Mahasiswa","JK","KD
Jurusan","Tgl.Lahir","Alamat","Telp","Email"}
){
boolean[] canEdit = new boolean[]{
false, false, false, false
};
@Override
public boolean isCellEditable(int rowIndex, int columnIndex){
return canEdit[columnIndex];
}
};
}
private void setDefaultTable() {
String stat ="";

```

```

        try {
Class.forName(driver);
Connection con = DriverManager.getConnection(dtbase,user,pass);
Statement st = con.createStatement();
        String SQL = "SELECT * FROM tb_mahasiswa order by npm Asc";
        ResultSet res = st.executeQuery(SQL);
        while(res.next()){
data[0] = res.getString(1);
data[1] = res.getString(2);
data[2] = res.getString(3);
data[3] = res.getString(4);
data[4] = res.getString(5);
data[5] = res.getString(6);
data[6] = res.getString(7);
data[7] = res.getString(8);
tableModel.addRow(data);
        }
        } catch (Exception ex) {
JOptionPane.showMessageDialog(null, "Koneksi database error");
        }
}

public void refresh() throws SQLException{
        String SQL = "select * from tb_mahasiswa order by npm Asc";
        Connection con = DriverManager.getConnection(dtbase,user,pass);
        Statement st = con.createStatement();
        ResultSet res = st.executeQuery(SQL);
        if (res != null) {
DefaultTableModel model = (DefaultTableModel) jTable1.getModel();
        for (int i=tableModel.getRowCount()-1; i>=0; i--) {
                model.removeRow(i);
        }
        try {

```



```

        while(res.next()){
            data[0] = res.getString(1);
            data[1] = res.getString(2);
            data[2] = res.getString(3);
            data[3] = res.getString(4);
            data[4] = res.getString(5);
            data[5] = res.getString(6);
            data[6] = res.getString(7);
            data[7] = res.getString(8);
            tableModel.addRow(data);
        }
    } catch (Exception ex) {
        JOptionPane.showMessageDialog(null, "Connection database error");
    }
}

public void bersih(){
    jTextField1.setText("");
    jTextField2.setText("");
    jTextField3.setText("");
    jTextField4.setText("");
    jTextField5.setText("");
    jTextField6.setText("");
    tanggal_lahir.setDate(null);
    jComboBox1.setSelectedItem("-");
    jRadioButton1.setSelected(false);
    jRadioButton2.setSelected(false);
    jRadioButton1.setEnabled(true);
    jRadioButton2.setEnabled(true);
    jTextField2.requestFocus();
}

public void Tampil(){

```

```

try{
    String j_kelamin = "";
    Connection kon = DriverManager.getConnection(dtbase,user,pass);
    Statement stt = kon.createStatement();
    String sql = "SELECT * FROM tb_mahasiswa where npm =
"+jTextField1.getText()+"";
    stt = kon.createStatement();
    ResultSet rsProdi = stt.executeQuery(sql);
    while(rsProdi.next()){
        jTextField2.setText(rsProdi.getString("nama"));
        jTextField3.setText(rsProdi.getString("kode_jurusan"));
        tanggal_lahir.setDate(rsProdi.getDate("tgl_lahir"));
        jTextField4.setText(rsProdi.getString("alamat"));
        jTextField5.setText(rsProdi.getString("no_telp"));
        jTextField6.setText(rsProdi.getString("email"));
        j_kelamin = rsProdi.getString("jk");
    }
    if(j_kelamin.equals("jk")){
    }else{
        if(j_kelamin.equals("L")){
            jButton1.setSelected(true);
            jButton2.setEnabled(false);
            jButton2.setSelected(false);
        }else{
            jButton2.setSelected(true);
            jButton1.setEnabled(false);
            jButton1.setSelected(false);
        }
    }
}
} catch(SQLException ex){
    JOptionPane.showMessageDialog(null, ex);
}

```

```

    }
    }
    int row1 = 0;
    public void Tampil_tabel(){
        row1 = jTable1.getSelectedRow();
        jTextField1.setText(tableModel.getValueAt(row1, 0).toString());
    }
    public void simpan_data(){
        String s_1 = jTextField1.getText();
        String s_2 = jTextField2.getText();
        String s_3 = "";
        String s_4 = jTextField3.getText();
        String s_5 = Tanggal1;
        String s_6 = jTextField4.getText();
        String s_7 = jTextField5.getText();
        String s_8 = jTextField6.getText();
        //validasi isian
        if (s_1.isEmpty()){
            JOptionPane.showMessageDialog(null,"Kode Masih Kosong");
            jTextField1.requestFocus();
            return;
        }
        if (s_2.isEmpty()){
            JOptionPane.showMessageDialog(null,"Nama Masih Kosong");
            jTextField2.requestFocus();
            return;
        }
        if (jRadioButton1.isSelected()){
            s_3= "L";
        }
        else if (jRadioButton2.isSelected()){
            s_3 = "P";
        }
    }

```

```

}
if (s_4.isEmpty()){
    JOptionPane.showMessageDialog(null,"Bobot Awal Masih Kosong");
    jTextField3.requestFocus();
    return;
}
if (s_5.isEmpty()){
    JOptionPane.showMessageDialog(null,"Bobot Target Masih Kosong");
    jTextField4.requestFocus();
    return;
}
if (s_6.isEmpty()){
    JOptionPane.showMessageDialog(null,"KD Jenis Masih Kosong");
    jTextField5.requestFocus();
    return;
}
if (s_7.isEmpty()){
    JOptionPane.showMessageDialog(null,"Kode Kandang Masih Kosong");
    jTextField6.requestFocus();
    return;
}
try{
    Connection kon = DriverManager.getConnection(dtbase,user,pass);
    Statement stt = kon.createStatement();
    String sql = "INSERT INTO tb_mahasiswa VALUES("
        + s_1 + ","
        + s_2 + ","
        + s_3 + ","
        + s_4 + ","
        + s_5 + ","
        + s_6 + ","
        + s_7 + ","

```

```

        + s_8 + "));
int done = stt.executeUpdate(sql);
data[0] = jTextField1.getText();
data[1] = jTextField2.getText();
data[2] = s_3;
data[3] = jTextField3.getText();
data[4] = Tanggal1;
data[5] = jTextField4.getText();
data[6] = jTextField5.getText();
data[7] = jTextField6.getText();
tableModel.insertRow(0, data);
JOptionPane.showMessageDialog(null, "Data Sukses tersimpan" ,
"Informasi", JOptionPane.INFORMATION_MESSAGE);
bersih();
refresh();
} catch (Exception ex) {
    JOptionPane.showMessageDialog(null, "Data Gagal tersimpan" ,
    "Informasi", JOptionPane.INFORMATION_MESSAGE);
}
}

```

```

public void ubah_data(){
    String s_1 = jTextField1.getText();
    String s_2 = jTextField2.getText();
    String s_3 = "";
    String s_4 = jTextField3.getText();
    String s_5 = Tanggal1;
    String s_6 = jTextField4.getText();
    String s_7 = jTextField5.getText();
    String s_8 = jTextField6.getText();
    //validasi isian
    if (s_1.isEmpty()){

```

```
JOptionPane.showMessageDialog(null,"Kode Masih Kosong");
jTextField1.requestFocus();
return;
}
if (s_2.isEmpty()){
    JOptionPane.showMessageDialog(null,"Nama Masih Kosong");
    jTextField2.requestFocus();
    return;
}
if (jRadioButton1.isSelected()){
    s_3= "L";
}
else if (jRadioButton2.isSelected()){
    s_3 = "P";
}
if (s_4.isEmpty()){
    JOptionPane.showMessageDialog(null,"Bobot Awal Masih Kosong");
    jTextField3.requestFocus();
    return;
}
if (s_5.isEmpty()){
    JOptionPane.showMessageDialog(null,"Bobot Target Masih Kosong");
    jTextField4.requestFocus();
    return;
}
if (s_6.isEmpty()){
    JOptionPane.showMessageDialog(null,"KD Jenis Masih Kosong");
    jTextField5.requestFocus();
    return;
}
if (s_7.isEmpty()){
    JOptionPane.showMessageDialog(null,"Kode Kandang Masih Kosong");
```

```

        jTextField6.requestFocus();
        return;
    }
    try{
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        String sql = "UPDATE tb_mahasiswa SET nama="
            + s_2 + ",jk="
            + s_3 + ",kode_jurusan="
            + s_4 + ",tgl_lahir="
            + s_5 + ",alamat="
            + s_6 + ",no_telp="
            + s_7 + ",email="
            + s_8 + " WHERE npm="
            + s_1 + """;
        int done = stt.executeUpdate(sql);
        data[0] = jTextField1.getText();
        data[1] = jTextField2.getText();
        data[2] = s_3;
        data[3] = jTextField3.getText();
        data[4] = Tanggal1;
        data[5] = jTextField4.getText();
        data[6] = jTextField5.getText();
        data[7] = jTextField6.getText();
        tableModel.removeRow(0);
        tableModel.insertRow(0, data);
        JOptionPane.showMessageDialog(null, "Data BERHASIL Diubah" ,
        "Informasi", JOptionPane.INFORMATION_MESSAGE);
        bersih();
        refresh();
    }catch(Exception ex){

```

```

        JOptionPane.showMessageDialog(null, "Data GAGAL Diubah" ,
        "Informasi", JOptionPane.INFORMATION_MESSAGE);
    }
}
public void hapus_data(){
    try{
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        int ok=JOptionPane.showConfirmDialog(rootPane, "Apakah Anda yakin
        Akan Menghapus Data Ini???", "Question",JOptionPane.OK_OPTION);
        if (ok==0) {
            String sql ="DELETE FROM tb_mahasiswa WHERE
            npm='"+jTextField1.getText()+"''";
            int done = stt.executeUpdate(sql);
            bersih();
            refresh();
            JOptionPane.showMessageDialog(this,
            "Data Berhasil Dihapus", "Informasi", 1);
        }
    }catch(Exception ex){
        JOptionPane.showMessageDialog(null, "Data Tidak Dapat Dihapus!");
    }
}
public void tampil_jurusan(){
    try {
        Class.forName(driver);
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        String SQL = "SELECT * FROM tb_jurusan ORDER BY jurusan ASC";
        ResultSet res = stt.executeQuery(SQL);
        while(res.next()){
            JComboBox1.addItem(res.getString("jurusan"));
        }
    }
}

```



```

    }
    } catch (Exception e) {
    JOptionPane.showMessageDialog(null, "Error Connection"+e);
    }
}

public void tampil_hasil_jurusan1(){
    try{
        String status="";
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        String sql = "SELECT * FROM tb_jurusan where jurusan="
        ""+jComboBox1.getSelectedItem()+"";
        ResultSet rsProdi = stt.executeQuery(sql);
        if(rsProdi.next()){
            jTextField3.setText(rsProdi.getString("kode_jurusan"));
        }}catch(Exception ex){
        }
    }

public void tampil_hasil_jurusan2(){
    try{
        String status="";
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        String sql = "SELECT * FROM tb_jurusan where kode_jurusan="
        ""+jTextField3.getText()+"";
        ResultSet rsProdi = stt.executeQuery(sql);
        if(rsProdi.next()){
            jComboBox1.setSelectedItem(rsProdi.getString("jurusan"));
        }}catch(Exception ex){
        }
    }
}

```