

KODING PROGRAM

```
package view_program;

import java.awt.event.KeyEvent;

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 program_server.Koneksi;

import javax.swing.table.TableRowSorter;

import java.text.SimpleDateFormat;

/**
 *
 * @author sempak
 */
public class form_anggota extends javax.swing.JInternalFrame {

    String data[]=new String[8];

    DefaultTableModel tb2;

    Koneksi dbsetting;

    String driver, dtbase, user,pass,userLogin;

    private Connection con;

    private Statement st;

    String Tanggal1;

    String Tanggal2;

    ImageIcon imagelcon;
```

```

ResultSet rs;

Koneksi d;

/** Creates new form form_anggota */
public form_anggota() {
    initComponents();
    dbsetting = new Koneksi();
    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});
    setDefaultTable();
    tampil_kelas();
}

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 [] {"NIS", "Nama", "ID Kelas", "Tgl. Daftar", "Tgl. Akhir"}
){
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 anggota order by nis 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);
tableModel.addRow(data);
}
}

```

```

    } catch (Exception ex) {
        JOptionPane.showMessageDialog(null, "Koneksi database error");
    }
}

public void refresh() throws SQLException{
    String SQL = "select * from anggota order by nis 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);
                tableModel.addRow(data);
            }
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(null, "Connection database error");
        }
    }
}
}

```

```

public void bersih(){
nis.setText("");
nama.setText("");
tanggalakhir.setDate(null);
tanggaldaftar.setDate(null);
jComboBox1.setSelectedItem("Pilih");
idkelas.setText("");
}

public void tampil_kelas(){
try {
Class.forName(driver);
Connection kon = DriverManager.getConnection(dtbase,user,pass);
Statement stt = kon.createStatement();
String SQL = "SELECT * FROM kelas ORDER BY nmkelas ASC";
ResultSet res = stt.executeQuery(SQL);
while(res.next()){
jComboBox1.addItem(res.getString("nmkelas"));
}
} catch (Exception e) {
JOptionPane.showMessageDialog(null, "Error Connection"+e);
}
}

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

```

```

private void tanggalakhirPropertyChange(java.beans.PropertyChangeEvent evt) {
    if(tanggalakhir.getDate()!=null){
        SimpleDateFormat Format = new SimpleDateFormat("yyyy-MM-dd");
        Tanggal2=Format.format(tanggalakhir.getDate());
    } // TODO add your handling code here:
}

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {
    nis.setText(String.valueOf(jTable1.getValueAt(jTable1.getSelectedRow(), 0)));
    nama.setText(String.valueOf(jTable1.getValueAt(jTable1.getSelectedRow(), 1)));
    idkelas.setText(String.valueOf(jTable1.getValueAt(jTable1.getSelectedRow(), 2))); //
    TODO add your handling code here:
}

private void nisCaretUpdate(javax.swing.event.CaretEvent evt) {
    try{
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        String sql = "SELECT * FROM anggota where nis= '"+nis.getText()+"'";
        ResultSet rsProdi = stt.executeQuery(sql);
        if(rsProdi.next()){
            tanggaldaftar.setDate(rs.getDate("tanggaldaftar"));
            tanggalakhir.setDate(rs.getDate("tanggalakhir"));
        }
    }catch(SQLException ex){
        JOptionPane.showMessageDialog(null, ex);
    }
}

```

```
} // TODO add your handling code here:  
}
```

```
private void button_kotak1ActionPerformed(java.awt.event.ActionEvent evt) {  
    String s_1 = nis.getText();  
    String s_2 = nama.getText();  
    String s_3 = idkelas.getText();  
    String s_4 = Tanggal1;  
    String s_5 = Tanggal2;  
    //validasi isian  
    if (s_1.isEmpty()){  
        JOptionPane.showMessageDialog(null,"NIS Masih Kosong");  
        nis.requestFocus();  
        return;  
    }  
    if (s_2.isEmpty()){  
        JOptionPane.showMessageDialog(null,"Nama Masih Kosong");  
        nama.requestFocus();  
        return;  
    }  
    if (s_3.isEmpty()){  
        JOptionPane.showMessageDialog(null,"Id Kelas Masih Kosong");  
        idkelas.requestFocus();  
        return;  
    }  
    if (s_4.isEmpty()){  
        JOptionPane.showMessageDialog(null,"Tanggal Daftar Masih Kosong");  
        tanggaldaftar.requestFocus();  
    }  
}
```



```

        return;
    }
    if (s_5.isEmpty()){
        JOptionPane.showMessageDialog(null,"Tanggal Akhir Masih Kosong");
        tanggalakhir.requestFocus();
        return;
    }
    try{
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();

        int ok=JOptionPane.showConfirmDialog(rootPane, "Apakah Anda yakin Akan
Menyimpan Data ini???", "Question",JOptionPane.OK_OPTION);
        if (ok==0) {
            String sql = "INSERT INTO anggota VALUES("
                + s_1 + ","
                + s_2 + ","
                + s_3 + ","
                + s_4 + ","
                + s_5 + ")";
            int done = stt.executeUpdate(sql);

            data[0] = nis.getText();
            data[1] = nama.getText();
            data[2] = idkelas.getText();
            data[3] = Tanggal1;
            data[4] = Tanggal2;
            tableModel.insertRow(0, data);
            bersih();
            refresh();
        }
    }
}

```

```

        JOptionPane.showMessageDialog(this,
            "Data Berhasil Disimpan", "Informasi", 1);
    }
} catch (Exception ex) {
    JOptionPane.showMessageDialog(null, "Error Simpan");
} // TODO add your handling code here:
}

private void button_kotak2ActionPerformed(java.awt.event.ActionEvent evt) {
    String s_1 = nis.getText();
    String s_2 = nama.getText();
    String s_3 = idkelas.getText();
    String s_4 = Tanggal1;
    String s_5 = Tanggal2;
    //validasi isian
    if (s_1.isEmpty()) {
        JOptionPane.showMessageDialog(null, "NIS Masih Kosong");
        nis.requestFocus();
        return;
    }
    if (s_2.isEmpty()) {
        JOptionPane.showMessageDialog(null, "Nama Masih Kosong");
        nama.requestFocus();
        return;
    }
    if (s_3.isEmpty()) {
        JOptionPane.showMessageDialog(null, "Id Kelas Masih Kosong");
        idkelas.requestFocus();
    }
}

```

```

        return;
    }
    if (s_4.isEmpty()){
        JOptionPane.showMessageDialog(null,"Tanggal Daftar Masih Kosong");
        tanggaldaftar.requestFocus();
        return;
    }
    if (s_5.isEmpty()){
        JOptionPane.showMessageDialog(null,"Tanggal Akhir Masih Kosong");
        tanggalakhir.requestFocus();
        return;
    }
    try{
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();

        int ok=JOptionPane.showConfirmDialog(rootPane, "Apakah Anda yakin Akan
Membah Data ini???", "Question",JOptionPane.OK_OPTION);
        if (ok==0) {
            String sql = "UPDATE anggota SET nama="
                + s_2+ ",idkelas="
                + s_3 + ",tanggaldaftar="
                + s_4+ ",tanggalakhir="
                + s_5 + " WHERE nis="
                + s_1 + """;
            int done = stt.executeUpdate(sql);
            data[0] = nis.getText();
            data[1] = nama.getText();
            data[2] = idkelas.getText();

```

```

data[3] = Tanggal1;
data[4] = Tanggal2;
tableModel.removeRow(0);
tableModel.insertRow(0, data);
bersih();
refresh();

JOptionPane.showMessageDialog(this,
    "Data Berhasil Diubah", "Informasi", 1);
}
}catch(Exception ex){
    JOptionPane.showMessageDialog(null,"Error Update");
}    // TODO add your handling code here:
}

private void button_kotak3ActionPerformed(java.awt.event.ActionEvent evt) {
try{
    Connection kon = DriverManager.getConnection(dtbase,user,pass);
    Statement stt = kon.createStatement();

    int ok=JOptionPane.showConfirmDialog(rootPane, "Apakah Anda yakin Akan
Menghapus Data Tersebut???", "Question",JOptionPane.OK_OPTION);
    if (ok==0) {
        String sql ="DELETE FROM anggota WHERE nis="+nis.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!");
    } // TODO add your handling code here:
}

private void button_kotak4ActionPerformed(java.awt.event.ActionEvent evt) {
    bersih(); // TODO add your handling code here:
}

private void jTextField1CaretUpdate(javax.swing.event.CaretEvent evt) {
    TableRowSorter sorter = new TableRowSorter(jTable1.getModel());
    jTable1.setRowSorter(sorter);
    sorter.setRowFilter(RowFilter.regexFilter("(?i)" + jTextField1.getText()));
    jTable1.setSelectionMode(1); // TODO add your handling code here:
}

private void button_kotak5ActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose(); // TODO add your handling code here:
}

private void jComboBox1ItemStateChanged(java.awt.event.ItemEvent evt) {
    try{
        String status="";
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        String sql = "SELECT * FROM kelas where nmkelas="
        ""+jComboBox1.getSelectedItem()+"";
        ResultSet rsProdi = stt.executeQuery(sql);
    }
}

```

```

        if(rsProdi.next()){
            idkelas.setText(rsProdi.getString("idkelas"));
        }
    }catch(Exception ex){
    } // TODO add your handling code here:
}

```

```

private void idkelasCaretUpdate(javax.swing.event.CaretEvent evt) {
    try{
        String status="";
        Connection kon = DriverManager.getConnection(dtbase,user,pass);
        Statement stt = kon.createStatement();
        String sql = "SELECT * FROM kelas where idkelas= '"+idkelas.getText()+"'";
        ResultSet rsProdi = stt.executeQuery(sql);
        if(rsProdi.next()){
            JComboBox1.setSelectedItem(rsProdi.getString("nmkelas"));
        }
    }catch(Exception ex){
    } // TODO add your handling code here:
}

```

```

private void nisKeyTyped(java.awt.event.KeyEvent evt) {
    if(nis.getText().length(>9)
    evt.consume();
    FilterHanyaAngka(evt); // 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();

        JOptionPane.showMessageDialog(null, "Maaf Data Yang Anda Inputkan Harus
        Angka");
    }
}

// Variables declaration - do not modify
private class_java.button_kotak button_kotak1;
private class_java.button_kotak button_kotak2;
private class_java.button_kotak button_kotak3;
private class_java.button_kotak button_kotak4;
private class_java.button_kotak button_kotak5;
private javax.swing.JTextField idkelas;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JLabel jLabel1;
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.JPanel jPanel1;  
private javax.swing.JScrollPane jScrollPane1;  
private javax.swing.JTable jTable1;  
private javax.swing.JTextField jTextField1;  
private javax.swing.JTextField nama;  
private javax.swing.JTextField nis;  
private com.toedter.calendar.JDateChooser tanggalakhir;  
private com.toedter.calendar.JDateChooser tanggaldaftar;  
// End of variables declaration  
  
}
```