**Class about**

/\*

 \* To change this template, choose Tools | Templates

 \* and open the template in the editor.

 \*/

/\*

 \* About.java

 \*

 \* Created on Jan 17, 2011, 11:56:44 AM

 \*/

package JeditorTeks;

//atribut

/\*\*

 \*

 \* @author MazArie

 \*/

public class About extends javax.swing.JFrame {

 /\*\* Creates new form About \*/

 public About() {//method

 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() {

 setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

 getContentPane().setLayout(layout);

 layout.setHorizontalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGap(0, 400, Short.MAX\_VALUE)

 );

 layout.setVerticalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGap(0, 300, Short.MAX\_VALUE)

 );

 pack();

 }// </editor-fold>

 /\*\*

 \* @param args the command line arguments

 \*/

 public static void main(String args[]) {//method

 java.awt.EventQueue.invokeLater(new Runnable() {

 public void run() {//method baru

 new About().setVisible(true);

 }

 });

 }

 // Variables declaration - do not modify

 // End of variables declaration

}

**Class Main**

 \* To change this template, choose Tools | Templates

 \* and open the template in the editor.

 \*/

package JeditorTeks;//atribut

/\*\*

 \*

 \* @author MzArie

 \*/

public class Main {

 /\*\*

 \* @param args the command line arguments

 \*/

 public static void main(String[] args) {//method

 // TODO code application logic here

 new editortext().setVisible(true);

 }

}

**Class Editor**

 \* editortext.java

 \*

 \* Created on 23 april 11, 8:43:43

 \*/

package JeditorTeks;

//import com.asprise.util.pdf.PDFReader;

import com.asprise.util.pdf.PDFReader;

import java.awt.print.PrinterException;

import java.io.\*;

import javax.swing.\*;

import javax.swing.filechooser.FileFilter;

import org.apache.poi.poifs.filesystem.\*;

import org.apache.poi.hwpf.\*;

import org.apache.poi.hwpf.extractor.\*;

/\*\*

 \*

 \* @author MzArie

 \*/

public class editortext extends javax.swing.JFrame {

 /\*\* deklarasi property selain komponen ui \*/

 private font1 hruf1;

 private font2 hruf2;

 JFileChooser fc;

 File file;

 boolean fileBaru;

 final String[] EXT = {".java"};

 final String[] EXT1 = {".html"};

 final String[] EXT2 = {".txt"};

 final String[] EXT3 = {".ini"};

 final String[] EXT4 = {".doc"};

 final String[] EXT5 = {".pdf"};

 //deklarasi string2 apa saja yg bs dibuka oleh program

 public editortext() { //method

 initComponents();

 Title1.setText("Form 1");

 Title2.setText("Form 2");

 fc = new JFileChooser();

 fc.addChoosableFileFilter(new jenisFile(EXT));

 fc.addChoosableFileFilter(new jenisFile(EXT1));

 fc.addChoosableFileFilter(new jenisFile(EXT2));

 fc.addChoosableFileFilter(new jenisFile(EXT3));

 fc.addChoosableFileFilter(new jenisFile(EXT4));

 fc.addChoosableFileFilter(new jenisFile(EXT5));

 fc.setAcceptAllFileFilterUsed(true);

 //mengatur agar filechooser bisa menampilkan seluruh ekstensi lainya selain yg diatas

 fileBaru = true;

 }

 /\*\* 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() {

 jSeparator1 = new javax.swing.JSeparator();

 jScrollPane1 = new javax.swing.JScrollPane();

 Editor1 = new javax.swing.JEditorPane();

 jScrollPane2 = new javax.swing.JScrollPane();

 Editor2 = new javax.swing.JEditorPane();

 Title1 = new javax.swing.JLabel();

 Title2 = new javax.swing.JLabel();

 jMenuBar1 = new javax.swing.JMenuBar();

 File1 = new javax.swing.JMenu();

 NewFile1 = new javax.swing.JMenuItem();

 NewFile2 = new javax.swing.JMenuItem();

 Open1 = new javax.swing.JMenuItem();

 Open2 = new javax.swing.JMenuItem();

 Save1 = new javax.swing.JMenuItem();

 Save2 = new javax.swing.JMenuItem();

 SaveAs1 = new javax.swing.JMenuItem();

 SaveAs2 = new javax.swing.JMenuItem();

 Print1 = new javax.swing.JMenuItem();

 Print2 = new javax.swing.JMenuItem();

 jSeparator4 = new javax.swing.JSeparator();

 Exit = new javax.swing.JMenuItem();

 PDF = new javax.swing.JMenu();

 OpenPdf1 = new javax.swing.JMenuItem();

 Openpdf2 = new javax.swing.JMenuItem();

 Doc1 = new javax.swing.JMenu();

 OpenDoc1 = new javax.swing.JMenuItem();

 OpenDoc2 = new javax.swing.JMenuItem();

 Edit = new javax.swing.JMenu();

 Cut1 = new javax.swing.JMenuItem();

 Cut2 = new javax.swing.JMenuItem();

 Copy1 = new javax.swing.JMenuItem();

 Copy2 = new javax.swing.JMenuItem();

 Paste1 = new javax.swing.JMenuItem();

 Paste2 = new javax.swing.JMenuItem();

 Format = new javax.swing.JMenu();

 Font1 = new javax.swing.JMenuItem();

 Font2 = new javax.swing.JMenuItem();

 jMenu2 = new javax.swing.JMenu();

 jMenuItem1 = new javax.swing.JMenuItem();

 setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

 setBackground(new java.awt.Color(153, 0, 153));

 Editor1.setAutoscrolls(false);

 jScrollPane1.setViewportView(Editor1);

 jScrollPane2.setViewportView(Editor2);

 Title1.setText("Form 1");

 Title2.setText("Form 2");

 File1.setText("File");

 File1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 File1ActionPerformed(evt);

 }

 });

 NewFile1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_N, java.awt.event.InputEvent.CTRL\_MASK));

 NewFile1.setText("New File1");

 NewFile1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 NewFile1ActionPerformed(evt);

 }

 });

 File1.add(NewFile1);

 NewFile2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_N, java.awt.event.InputEvent.ALT\_MASK));

 NewFile2.setText("New File2");

 NewFile2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 NewFile2ActionPerformed(evt);

 }

 });

 File1.add(NewFile2);

 Open1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_O, java.awt.event.InputEvent.CTRL\_MASK));

 Open1.setText("Open File1");

 Open1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Open1ActionPerformed(evt);

 }

 });

 File1.add(Open1);

 Open2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_O, java.awt.event.InputEvent.ALT\_MASK));

 Open2.setText("Open File2");

 Open2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Open2ActionPerformed(evt);

 }

 });

 File1.add(Open2);

 Save1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_S, java.awt.event.InputEvent.CTRL\_MASK));

 Save1.setText("Save File1");

 Save1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Save1ActionPerformed(evt);

 }

 });

 File1.add(Save1);

 Save2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_S, java.awt.event.InputEvent.ALT\_MASK));

 Save2.setText("Save File2");

 Save2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Save2ActionPerformed(evt);

 }

 });

 File1.add(Save2);

 SaveAs1.setText("Save As 1");

 SaveAs1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 SaveAs1ActionPerformed(evt);

 }

 });

 File1.add(SaveAs1);

 SaveAs2.setText("Save As 2");

 SaveAs2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 SaveAs2ActionPerformed(evt);

 }

 });

 File1.add(SaveAs2);

 Print1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_P, java.awt.event.InputEvent.CTRL\_MASK));

 Print1.setText("Print 1");

 Print1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Print1ActionPerformed(evt);

 }

 });

 File1.add(Print1);

 Print2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_P, java.awt.event.InputEvent.ALT\_MASK));

 Print2.setText("Print 2");

 Print2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Print2ActionPerformed(evt);

 }

 });

 File1.add(Print2);

 File1.add(jSeparator4);

 Exit.setText("Exit");

 Exit.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 ExitActionPerformed(evt);

 }

 });

 File1.add(Exit);

 jMenuBar1.add(File1);

 PDF.setText("PDF");

 OpenPdf1.setText("Open Pdf1");

 OpenPdf1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 OpenPdf1ActionPerformed(evt);

 }

 });

 PDF.add(OpenPdf1);

 Openpdf2.setText("Open Pdf2");

 Openpdf2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Openpdf2ActionPerformed(evt);

 }

 });

 PDF.add(Openpdf2);

 jMenuBar1.add(PDF);

 Doc1.setText("DOC");

 OpenDoc1.setText("Open Doc 1");

 OpenDoc1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 OpenDoc1ActionPerformed(evt);

 }

 });

 Doc1.add(OpenDoc1);

 OpenDoc2.setText("Open Doc 2");

 OpenDoc2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 OpenDoc2ActionPerformed(evt);

 }

 });

 Doc1.add(OpenDoc2);

 jMenuBar1.add(Doc1);

 Edit.setText("Edit");

 Cut1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_X, java.awt.event.InputEvent.CTRL\_MASK));

 Cut1.setText("Cut1");

 Cut1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Cut1ActionPerformed(evt);

 }

 });

 Edit.add(Cut1);

 Cut2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_X, java.awt.event.InputEvent.ALT\_MASK));

 Cut2.setText("Cut2");

 Cut2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Cut2ActionPerformed(evt);

 }

 });

 Edit.add(Cut2);

 Copy1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_C, java.awt.event.InputEvent.CTRL\_MASK));

 Copy1.setText("Copy1");

 Copy1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Copy1ActionPerformed(evt);

 }

 });

 Edit.add(Copy1);

 Copy2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_C, java.awt.event.InputEvent.ALT\_MASK));

 Copy2.setText("Copy2");

 Copy2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Copy2ActionPerformed(evt);

 }

 });

 Edit.add(Copy2);

 Paste1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_V, java.awt.event.InputEvent.CTRL\_MASK));

 Paste1.setText("Paste1");

 Paste1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Paste1ActionPerformed(evt);

 }

 });

 Edit.add(Paste1);

 Paste2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_V, java.awt.event.InputEvent.ALT\_MASK));

 Paste2.setText("Paste2");

 Paste2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Paste2ActionPerformed(evt);

 }

 });

 Edit.add(Paste2);

 jMenuBar1.add(Edit);

 Format.setText("Format");

 Font1.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_F, java.awt.event.InputEvent.CTRL\_MASK));

 Font1.setText("Font1");

 Font1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Font1ActionPerformed(evt);

 }

 });

 Format.add(Font1);

 Font2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK\_F, java.awt.event.InputEvent.ALT\_MASK));

 Font2.setText("Font 2");

 Font2.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 Font2ActionPerformed(evt);

 }

 });

 Format.add(Font2);

 jMenuBar1.add(Format);

 jMenu2.setText("Help");

 jMenuItem1.setText("About");

 jMenuItem1.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 jMenuItem1ActionPerformed(evt);

 }

 });

 jMenu2.add(jMenuItem1);

 jMenuBar1.add(jMenu2);

 setJMenuBar(jMenuBar1);

 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

 getContentPane().setLayout(layout);

 layout.setHorizontalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addGap(166, 166, 166)

 .addComponent(Title1, javax.swing.GroupLayout.DEFAULT\_SIZE, 60, Short.MAX\_VALUE)

 .addGap(343, 343, 343)

 .addComponent(Title2, javax.swing.GroupLayout.DEFAULT\_SIZE, 110, Short.MAX\_VALUE)

 .addGap(120, 120, 120))

 .addGroup(layout.createSequentialGroup()

 .addContainerGap()

 .addComponent(jScrollPane1)

 .addGap(10, 10, 10)

 .addComponent(jScrollPane2)

 .addContainerGap())

 );

 layout.setVerticalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

 .addContainerGap()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

 .addComponent(Title1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

 .addComponent(Title2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

 .addGap(18, 18, 18)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

 .addComponent(jScrollPane2, javax.swing.GroupLayout.DEFAULT\_SIZE, 383, Short.MAX\_VALUE)

 .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 383, Short.MAX\_VALUE))

 .addContainerGap())

 );

 pack();

 }// </editor-fold>

 private void NewFile1ActionPerformed(java.awt.event.ActionEvent evt) {

 //method:

 if (Editor1.getText().hashCode() != "".hashCode()) {

 int jwb = JOptionPane.showConfirmDialog(this, "Data Sudah Anda rubah \n Apakah akan di Simpan ?",

 "Konfirmasi", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.QUESTION\_MESSAGE);

 switch (jwb) {

 case 0:

 SaveAs1.doClick();

 break;

 case 1:

 Editor1.setText("");

 Title1.setText("Form 1");

 break;

 case 2:

 return;

 } //kondisi yg bernilai false

 }

 }

 private void NewFile2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method:

 if (Editor2.getText().hashCode() != "".hashCode()) {

 int jwb = JOptionPane.showConfirmDialog(this, "Data Sudah Anda rubah \n Apakah akan di Simpan ?",

 "Konfirmasi", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.QUESTION\_MESSAGE);

 switch (jwb) {

 case 0:

 SaveAs2.doClick();

 break;

 case 1:

 Editor2.setText("");

 Title2.setText("Form 2");

 break;

 case 2:

 return;

 }

 }

 }

 /\*\*

 \* Methode ini dipanggil saat mengklik menu File > Open File 1.

 \* Akan ada pengecekan di dalam kode program.

 \* @param evt

 \*/

 private void Open1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method:

 if (Editor1.getText().hashCode() != "".hashCode()) {

 int jwb = JOptionPane.showConfirmDialog(this, "Apakah anda ingin menimpan file yang telah dirubah?",

 "Konfirmasi", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.QUESTION\_MESSAGE);

 switch (jwb) {

 case 0:

 SaveAs1.doClick();

 break;

 case 1: //no,mk tidak simpan

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try { //jika benar

 Editor1.read(new FileInputStream(file), null);

 } catch (IOException e) {

 msg("Gagal Buka File : " + file.getName());

 return;

 }

 Title1.setText(" " + file.getName());

 fileBaru = false;

 }

 break;

 case 2:

 return;

 }

 } else { //jika editor 1 kosong

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try {

 Editor1.read(new FileInputStream(file), null);

 } catch (IOException e) {

 msg("Gagal Buka File : " + file.getName());

 return;

 }

 Title1.setText(" " + file.getName());

 fileBaru = false;

 }

 }

 }

 private void Save1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method:

 if (fc.showSaveDialog(this) == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 if (!file.exists() || OkToReplace()) {

 SimpanFile1();

 }

 }

 }

 private void SaveAs1ActionPerformed(java.awt.event.ActionEvent evt) {

 // menampilkan dialog utuk menyimpan file

 if (fc.showSaveDialog(this) == JFileChooser.APPROVE\_OPTION) { //jika disetujui mk:

 file = fc.getSelectedFile();

 if (!file.exists() || OkToReplace()) {

 SimpanFile1();

 }

 }

 }

 private void File1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method file:

 }

 private void Save2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method 2:

 if (fc.showSaveDialog(this) == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 if (!file.exists() || OkToReplace()) {

 SimpanFile2();

 }

 }

 }

 private void SaveAs2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method:

 if (fc.showSaveDialog(this) == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 if (!file.exists() || OkToReplace()) {

 SimpanFile2();

 }

 }

 }

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

 // method exit:

 System.exit(0);

 }

 private void Open2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method open:

 if (Editor2.getText().hashCode() != "".hashCode()) {

 int jwb = JOptionPane.showConfirmDialog(this, "apakah anda ingin menyimpan file yang telah dirubah?",

 "Konfirmasi", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.QUESTION\_MESSAGE);

 switch (jwb) {

 case 0:

 SaveAs2.doClick();

 break;

 case 1:

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try {

 Editor2.read(new FileInputStream(file), null);

 } catch (IOException e) {

 msg("Gagal Buka File : " + file.getName());

 return;

 }

 Title2.setText(" " + file.getName());

 fileBaru = false;

 }

 break;

 case 2:

 return;

 }

 } else { //jika editor2 kosong=

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try {

 Editor2.read(new FileInputStream(file), null);

 } catch (IOException e) {

 msg("Gagal Buka File : " + file.getName());

 return;

 }

 Title2.setText(" " + file.getName());

 fileBaru = false;

 }

 }

 }

 private void Cut1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method cut:

 Editor1.cut();

 }

 private void Cut2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method cut:

 Editor2.cut();

 }

 private void Copy1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method copy:

 Editor1.copy();

 }

 private void Copy2ActionPerformed(java.awt.event.ActionEvent evt) {

 // mthod copy2:

 Editor2.copy();

 }

 private void Paste1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method paste1:

 Editor1.paste();

 }

 private void Font1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method font:

 hruf1 = new font1(Editor1);

 hruf1.show();

 }

 private void Font2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method font2:

 hruf2 = new font2(Editor2);

 hruf2.show();

 }

 private void Pdf1ActionPerformed(java.awt.event.ActionEvent evt) {

 if (Editor1.getText().hashCode() != "".hashCode()) {

 int jwb = JOptionPane.showConfirmDialog(this, "Data Sudah Anda rubah \n Apakah akan di Simpan ?",

 "Konfirmasi", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.QUESTION\_MESSAGE);

 switch (jwb) {

 case 0:

 SaveAs1.doClick();

 break;

 case 1:

 Editor1.setText("");

 Title1.setText("Untitled1-NOTES");

 break;

 case 2:

 return;

 }

 }

 }

 private void Openpdf2ActionPerformed(java.awt.event.ActionEvent evt) {

 // TODO add your handling code here:

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try {

 Editor2.read(new FileInputStream(file), null);

 //readPdf2(file);

 } catch (IOException e) {

 //msg("Gagal Buka File : "+file.getName());

 return;

 }

 Title2.setText(" " + file.getName());

 fileBaru = false;

 }

 }

 private void OpenPdf1ActionPerformed(java.awt.event.ActionEvent evt) {

 // TODO add your handling code here:

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try {

 Editor1.read(new FileInputStream(file), null);

 // readPdf1(file);

 } catch (IOException e) {

 //msg("Gagal Buka File : "+file.getName());

 return;

 }

 Title1.setText(" " + file.getName());

 fileBaru = false;

 }

 }

 private void OpenDoc1ActionPerformed(java.awt.event.ActionEvent evt) {

 //method open doc

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try {

 Editor1.read(new FileInputStream(file), null);

 readDoc1(file);

 } catch (IOException e) {

 //msg("Gagal Buka File : "+file.getName());

 return;

 }

 Title1.setText(" " + file.getName());

 fileBaru = false;

 }

 }

 private void OpenDoc2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method doc 2

 int hasil = fc.showOpenDialog(this);

 if (hasil == JFileChooser.APPROVE\_OPTION) {

 file = fc.getSelectedFile();

 try {

 Editor2.read(new FileInputStream(file), null);

 readDoc2(file);

 } catch (IOException e) {

 //msg("Gagal Buka File : "+file.getName());

 return;

 }

 Title2.setText(" " + file.getName());

 fileBaru = false;

 }

 }

 private void Print1ActionPerformed(java.awt.event.ActionEvent evt) {

 //jika editor 1 tidak kosong

 if (!Editor1.getText().equals("")) {

 try {

 Editor1.print();

 } catch (PrinterException ex) {

 JOptionPane.showMessageDialog(this, "Data pada file 1 kosong");

 }

 }

 }

 private void Print2ActionPerformed(java.awt.event.ActionEvent evt) {

 // method print2

 if (!Editor2.getText().equals("")) {

 try {

 Editor2.print();

 } catch (PrinterException ex) {

 JOptionPane.showMessageDialog(this, "Data pada file 2 kosong");

 }

 }

 }

 private void Doc1ActionPerformed(java.awt.event.ActionEvent evt) {

 // method doc 1

 if (Editor1.getText().hashCode() != "".hashCode()) {

 int jwb = JOptionPane.showConfirmDialog(this, "Data Sudah Anda rubah \n Apakah akan di Simpan ?",

 "Konfirmasi", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.QUESTION\_MESSAGE);

 switch (jwb) {

 case 0:

 SaveAs1.doClick();

 break;

 case 1:

 Editor1.setText("");

 Title1.setText("Untitled1-NOTES");

 break;

 case 2:

 return;

 }

 }

 }

 private void Paste2ActionPerformed(java.awt.event.ActionEvent evt) {

 // memanggil method paste()pada editor 1

 Editor2.paste();

}

 private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {

 // membuat dan menampilkan menu help

 fHelp help = new fHelp(this, true);

 help.setVisible(true);

 }

 public static void main(String args[]) {

 java.awt.EventQueue.invokeLater(new Runnable() {

 public void run() {

 new editortext().setVisible(true);

 }

 });

 }

 // Variables declaration - do not modify

 private javax.swing.JMenuItem Copy1;

 private javax.swing.JMenuItem Copy2;

 private javax.swing.JMenuItem Cut1;

 private javax.swing.JMenuItem Cut2;

 private javax.swing.JMenu Doc1;

 private javax.swing.JMenu Edit;

 private javax.swing.JEditorPane Editor1;

 private javax.swing.JEditorPane Editor2;

 private javax.swing.JMenuItem Exit;

 private javax.swing.JMenu File1;

 private javax.swing.JMenuItem Font1;

 private javax.swing.JMenuItem Font2;

 private javax.swing.JMenu Format;

 private javax.swing.JMenuItem NewFile1;

 private javax.swing.JMenuItem NewFile2;

 private javax.swing.JMenuItem Open1;

 private javax.swing.JMenuItem Open2;

 private javax.swing.JMenuItem OpenDoc1;

 private javax.swing.JMenuItem OpenDoc2;

 private javax.swing.JMenuItem OpenPdf1;

 private javax.swing.JMenuItem Openpdf2;

 private javax.swing.JMenu PDF;

 private javax.swing.JMenuItem Paste1;

 private javax.swing.JMenuItem Paste2;

 private javax.swing.JMenuItem Print1;

 private javax.swing.JMenuItem Print2;

 private javax.swing.JMenuItem Save1;

 private javax.swing.JMenuItem Save2;

 public javax.swing.JMenuItem SaveAs1;

 private javax.swing.JMenuItem SaveAs2;

 private javax.swing.JLabel Title1;

 private javax.swing.JLabel Title2;

 private javax.swing.JMenu jMenu2;

 private javax.swing.JMenuBar jMenuBar1;

 private javax.swing.JMenuItem jMenuItem1;

 private javax.swing.JScrollPane jScrollPane1;

 private javax.swing.JScrollPane jScrollPane2;

 private javax.swing.JSeparator jSeparator1;

 private javax.swing.JSeparator jSeparator4;

 // End of variables declaration

 private void msg(String pesan) { //menampilkan pesan dgn joption

 JOptionPane.showMessageDialog(this, pesan, "Informasi", JOptionPane.INFORMATION\_MESSAGE);

 }

 private boolean OkToReplace() {

 return (JOptionPane.showConfirmDialog(this, "File " + file.getName() + " Sudah Ada Apakah mau diganti ?",

 "Peringatan", JOptionPane.YES\_NO\_OPTION) == 0);

 //untu replace file return boolean tru jika yes dan false jka no

 }

 private void SimpanFile1() {

 PrintWriter pw = null;

 try {

 //menyimpan file

 pw = new PrintWriter(new BufferedWriter(new FileWriter(file)));

 } catch (IOException e) {

 msg("Gagal Simpan File : '" + file.getName());

 return;

 } //cetak texs editor 1

 pw.print(Editor1.getText());

 pw.close();

 //this.setTitle(file.getName()+" - NOTES");

 this.Title1.setText(file.getName() + "");

 fileBaru = false;

 }

 private void SimpanFile2() {

 PrintWriter pw = null;

 try {

 pw = new PrintWriter(new BufferedWriter(new FileWriter(file)));

 } catch (IOException e) {

 msg("Gagal Simpan File : '" + file.getName() + " ' ");

 return;

 }

 pw.print(Editor2.getText());

 pw.close();

 //this.setTitle(file.getName()+" - NOTES");

 this.Title2.setText(file.getName() + "");

 fileBaru = false;

 }

 private void readPdf1(File file) throws FileNotFoundException, IOException {

 PDFReader reader = new PDFReader(new FileInputStream(file));

 reader.open(); // open the file.

 int pages = reader.getNumberOfPages();

 String d = "";

 String e = "";

 for (int i = 0; i <= pages; i++) { //tampilkan file perpage kemudian gabungkan

 String c = reader.extractTextFromPage(i); //extrac text dari fil pdf

 d = e + c;

 e = d;

 Editor1.setText(d); //tampilkan semua text yang didapatkan dari editor1

 }

 reader.close();

 }

 private void readPdf2(File file) throws FileNotFoundException, IOException {

 PDFReader reader = new PDFReader(new FileInputStream(file));

 reader.open();

 int pages = reader.getNumberOfPages();

 String d = "";

 String e = "";

 String c = "";

 for (int i = 0; i <= pages; i++) {

 c = reader.extractTextFromPage(i);

 d = e + c;

 e = d;

 Editor2.setText(d);

 }

 }

 class jenisFile extends FileFilter {

 private String[] s;

//klas untuk filtering pada pembukaan dan penutupan file

 jenisFile(String[] sArg) {

 s = sArg;

 }

 //@Override

 public boolean accept(File fArg) {

 if (fArg.isDirectory()) {

 return true;

 }

 for (int i = 0; i < s.length; i++) {

 if (fArg.getName().toLowerCase().indexOf(s[i].toLowerCase()) > 0) {

 return true;

 }

 }

 return false;

 }

 //@Override

 public String getDescription() {

 String tmp = "";

 for (int i = 0; i < s.length; ++i) {

 tmp += "\*" + s[i] + " ";

 }

 return tmp;

 }

 }

 private void readDoc1(File file) {

 POIFSFileSystem fs = null;

 try { //membaca file doc1

 fs = new POIFSFileSystem(new FileInputStream(file));

 HWPFDocument doc = new HWPFDocument(fs);

 //membuat objek doc dari klas hwpfdDocument

 int pageNumber = 1;

 //mengestrak file per paragrap dari file doc/word

 WordExtractor we = new WordExtractor(doc);

 String[] paragraphs = we.getParagraphText();

 String c = "";

 String d = "";

 String e = "";

 for (int i = 0; i <= paragraphs.length; i++) { //

 c = paragraphs[i].toString().trim();

 d = e + "\n" + c;

 e = d;

 Editor1.setText(d);

 }

 } catch (Exception ex) {

 System.out.println(ex);

 }

 }

 private void readDoc2(File file) {

 POIFSFileSystem fs = null;

 try {

 fs = new POIFSFileSystem(new FileInputStream(file));

 HWPFDocument doc = new HWPFDocument(fs);

 int pageNumber = 1;

 WordExtractor we = new WordExtractor(doc);

 String[] paragraphs = we.getParagraphText();

 String c = "";

 String d = "";

 String e = "";

 for (int i = 0; i <= paragraphs.length; i++) {

 c = paragraphs[i].toString().trim();

 d = e + "\n" + c;

 e = d;

 Editor2.setText(d);

 }

 } catch (Exception ex) {

 System.out.println(ex);

 }

 }

}

Class fHelp

/\*

 \* To change this template, choose Tools | Templates

 \* and open the template in the editor.

 \*/

/\*

 \* fHelp.java

 \*

 \* Created on 13 Jul 11, 22:06:44

 \*/

package JeditorTeks;

/\*\*

 \*

 \* @author hp mini

 \*/

public class fHelp extends javax.swing.JDialog {

 /\*\* Creates new form fHelp \*/

 public fHelp(java.awt.Frame parent, boolean modal) {

 super(parent, modal);

 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() {

 btnOK = new javax.swing.JButton();

 jScrollPane1 = new javax.swing.JScrollPane();

 jTextArea1 = new javax.swing.JTextArea();

 setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE\_ON\_CLOSE);

 btnOK.setText("OK");

 btnOK.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 btnOKActionPerformed(evt);

 }

 });

 jTextArea1.setColumns(20);

 jTextArea1.setEditable(false);

 jTextArea1.setRows(5);

 jTextArea1.setText("aplikasi ini digunakan untuk membuka \ndan amengedit file dalam bentuk \nekstensi \*.txt, \*.html, \*.java, \*.pdf dan \n\*.doc. kegunaan aplikasi ini cukup sederhana\npengguna dapat menggunakan menu-menu yang ada.\n\n\n\t\tSELAMAT MENCOBA\n\n\t\t\t\t (Ariyanto)");

 jScrollPane1.setViewportView(jTextArea1);

 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

 getContentPane().setLayout(layout);

 layout.setHorizontalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGap(0, 400, Short.MAX\_VALUE)

 .addGroup(layout.createSequentialGroup()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addContainerGap()

 .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 380, Short.MAX\_VALUE))

 .addGroup(layout.createSequentialGroup()

 .addGap(168, 168, 168)

 .addComponent(btnOK)))

 .addContainerGap())

 );

 layout.setVerticalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGap(0, 300, Short.MAX\_VALUE)

 .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

 .addContainerGap()

 .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 214, Short.MAX\_VALUE)

 .addGap(18, 18, 18)

 .addComponent(btnOK)

 .addGap(34, 34, 34))

 );

 pack();

 }// </editor-fold>

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

 // TODO add your handling code here:

 this.dispose();

}

 /\*\*

 \* @param args the command line arguments

 \*/

 public static void main(String args[]) {

 java.awt.EventQueue.invokeLater(new Runnable() {

 public void run() {

 fHelp dialog = new fHelp(new javax.swing.JFrame(), true);

 dialog.addWindowListener(new java.awt.event.WindowAdapter() {

 public void windowClosing(java.awt.event.WindowEvent e) {

 System.exit(0);

 }

 });

 dialog.setVisible(true);

 }

 });

 }

 // Variables declaration - do not modify

 private javax.swing.JButton btnOK;

 private javax.swing.JScrollPane jScrollPane1;

 private javax.swing.JTextArea jTextArea1;

 // End of variables declaration

}

**Class font 1**

/\*

 \* To change this template, choose Tools | Templates

 \* and open the template in the editor.

 \*/

/\*

 \* huruf1.java

 \*

 \* Created on 19 Okt 10, 0:21:40

 \*/

package JeditorTeks;

import java.awt.Font;

import javax.swing.JList;

import javax.swing.SwingConstants;

import javax.swing.event.ListSelectionEvent;

import javax.swing.event.ListSelectionListener;

import javax.swing.text.JTextComponent;

/\*\*

 \*

 \* @author MzArie

 \*/

public class font1 extends javax.swing.JFrame {

 /\*\* Creates new form huruf1 \*/

 private JTextComponent textcomponent;

 private JList list\_font;

 //list\_font = new JList(GraphicsEnvironment.getLocalGraphicsEnvironment().getAvailableFontFamilyNames());

 public font1(JTextComponent textcomponent) {

 initComponents();

 Sampel.setHorizontalAlignment(SwingConstants.CENTER);

 FontName.setSelectedValue("Arial", true);

 FontStyle.setSelectedValue("PLAIN", true);

 FontSize.setSelectedValue("12", true);

 this.textcomponent = textcomponent;

 }

 /\*\* 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() {

 jLabel2 = new javax.swing.JLabel();

 jScrollPane1 = new javax.swing.JScrollPane();

 FontName = new javax.swing.JList();

 jScrollPane2 = new javax.swing.JScrollPane();

 FontStyle = new javax.swing.JList();

 jLabel3 = new javax.swing.JLabel();

 jLabel1 = new javax.swing.JLabel();

 NameStyle = new javax.swing.JTextField();

 jLabel4 = new javax.swing.JLabel();

 NameSize = new javax.swing.JTextField();

 NameFont = new javax.swing.JTextField();

 jScrollPane3 = new javax.swing.JScrollPane();

 FontSize = new javax.swing.JList();

 Ok = new javax.swing.JButton();

 Cancel = new javax.swing.JButton();

 Sampel = new javax.swing.JTextField();

 setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

 setTitle("FONT");

 jLabel2.setText("Font Style : ");

 FontName.setModel(new javax.swing.AbstractListModel() {

 String[] strings = { "Agency FB", "Algerian", "Arial", "Arial Black", "Arial Narrow", "Arial Rounded MT Bold", "Arial Unicode MS", "Baskerville Old Face", "Bauhaus 93", "Bell MT", "Berlin Sans FB", "Berlin Sans FB Demi", "Bernard MT Condensed", "Blackadder ITC", "Bodoni MT", "Bodoni MT Black", "Bodoni MT Condensed", "Bodoni MT Poster Compressed", "Book Antiqua", "Bookman Old Style", "Britannic Bold", "Broadway", "Brush Script MT", "Calibri", "Californian FB", "Calisto MT", "Cambria", "Cambria Math", "Candara", "Castellar", "Centaur", "Century", "Century Gothic", "Century Schoolbook", "Chiller", "Colonna MT", "Comic Sans MS", "Consolas", "Constantia", "Cooper Black", "Copperplate Gothic Bold", "Copperplate Gothic Light", "Corbel", "Courier", "Courier New", "Curlz MT", "Edwardian Script ITC", "Elephant", "Engravers MT", "Eras Bold ITC", "Eras Demi ITC", "Eras Light ITC", "Eras Medium ITC", "Estrangelo Edessa", "Felix Titling", "Fixedsys", "Footlight MT Light", "Forte", "Franklin Gothic Book", "Franklin Gothic Demi", "Franklin Gothic Demi Cond", "Franklin Gothic Heavy", "Franklin Gothic Medium", "Franklin Gothic Medium Cond", "Freestyle Script", "French Script MT", "Garamond", "Gautami", "Georgia", "Gigi", "Gill Sans MT", "Gill Sans MT Condensed", "Gill Sans MT Ext Condensed Bold", "Gill Sans Ultra Bold", "Gill Sans Ultra Bold Condensed", "Gloucester MT Extra Condensed", "Goudy Old Style", "Goudy Stout", "Haettenschweiler", "Harlow Solid Italic", "Harrington", "High Tower Text", "Impact", "Imprint MT Shadow", "Informal Roman", "Jokerman", "Juice ITC", "Kristen ITC", "Kunstler Script", "Latha", "Lucida Bright", "Lucida Calligraphy", "Lucida Console", "Lucida Fax", "Lucida Handwriting", "Lucida Sans", "Lucida Sans Typewriter", "Lucida Sans Unicode", "Magneto", "Maiandra GD", "Mangal", "Matura MT Script Capitals", "Microsoft Sans Serif", "Mistral", "Modern", "Modern No. 20", "Monotype Corsiva", "MS Mincho", "MS Reference Sans Serif", "MS Sans Serif", "MS Serif", "Niagara Engraved", "Niagara Solid", "OCR A Extended", "Old English Text MT", "Onyx", "Palace Script MT", "Palatino Linotype", "Papyrus", "Perpetua", "Perpetua Titling MT", "Poor Richard", "Pristina", "Ravie", "Rockwell", "Rockwell Condensed", "Rockwell Extra Bold", "Roman", "Script", "Script MT Bold", "Segoe UI", "Showcard Gothic", "Small Fonts", "Snap ITC", "Stencil", "Sylfaen", "System", "Tahoma", "Tempus Sans ITC", "Terminal", "Times New Roman", "Trebuchet MS", "Tunga", "Tw Cen MT", "Tw Cen MT Condensed", "Tw Cen MT Condensed Extra Bold", "Verdana", "Viner Hand ITC", "Vivaldi", "Vladimir Script", "Wide Latin", "WST\_Czec", " " };

 public int getSize() { return strings.length; }

 public Object getElementAt(int i) { return strings[i]; }

 });

 FontName.addMouseListener(new java.awt.event.MouseAdapter() {

 public void mouseClicked(java.awt.event.MouseEvent evt) {

 FontNameMouseClicked(evt);

 }

 });

 jScrollPane1.setViewportView(FontName);

 FontStyle.setModel(new javax.swing.AbstractListModel() {

 String[] strings = { "PLAIN", "BOLD", "ITALIC", "BOLD ITALIC" };

 public int getSize() { return strings.length; }

 public Object getElementAt(int i) { return strings[i]; }

 });

 FontStyle.addMouseListener(new java.awt.event.MouseAdapter() {

 public void mouseClicked(java.awt.event.MouseEvent evt) {

 FontStyleMouseClicked(evt);

 }

 });

 jScrollPane2.setViewportView(FontStyle);

 jLabel3.setText("Size : ");

 jLabel1.setText("Font :");

 jLabel4.setText("Sample");

 FontSize.setModel(new javax.swing.AbstractListModel() {

 String[] strings = { "8", "10", "12", "14", "16", "18", "20", "22", "24", "26", "28", "32", "48", "72" };

 public int getSize() { return strings.length; }

 public Object getElementAt(int i) { return strings[i]; }

 });

 FontSize.addMouseListener(new java.awt.event.MouseAdapter() {

 public void mouseClicked(java.awt.event.MouseEvent evt) {

 FontSizeMouseClicked(evt);

 }

 });

 FontSize.addAncestorListener(new javax.swing.event.AncestorListener() {

 public void ancestorMoved(javax.swing.event.AncestorEvent evt) {

 }

 public void ancestorAdded(javax.swing.event.AncestorEvent evt) {

 FontSizeAncestorAdded(evt);

 }

 public void ancestorRemoved(javax.swing.event.AncestorEvent evt) {

 }

 });

 jScrollPane3.setViewportView(FontSize);

 Ok.setText("Ok");

 Ok.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 OkActionPerformed(evt);

 }

 });

 Cancel.setText("Cancel");

 Cancel.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 CancelActionPerformed(evt);

 }

 });

 Sampel.setText("AABBCCDD");

 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

 getContentPane().setLayout(layout);

 layout.setHorizontalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addContainerGap()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addComponent(Sampel, javax.swing.GroupLayout.PREFERRED\_SIZE, 386, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addContainerGap())

 .addGroup(layout.createSequentialGroup()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

 .addGroup(javax.swing.GroupLayout.Alignment.LEADING, layout.createSequentialGroup()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

 .addComponent(jLabel1)

 .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 158, Short.MAX\_VALUE)

 .addComponent(NameFont))

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

 .addGroup(layout.createSequentialGroup()

 .addGap(20, 20, 20)

 .addComponent(jLabel2))

 .addComponent(NameStyle)

 .addComponent(jScrollPane2))

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addGap(37, 37, 37)

 .addComponent(jLabel3))

 .addGroup(layout.createSequentialGroup()

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addComponent(NameSize, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, 56, Short.MAX\_VALUE)

 .addComponent(jScrollPane3, javax.swing.GroupLayout.DEFAULT\_SIZE, 48, Short.MAX\_VALUE)))))

 .addComponent(jLabel4, javax.swing.GroupLayout.Alignment.LEADING))

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 44, Short.MAX\_VALUE)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addComponent(Ok)

 .addComponent(Cancel))

 .addGap(83, 83, 83))))

 );

 layout.setVerticalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addContainerGap()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

 .addComponent(jLabel1)

 .addComponent(jLabel2)

 .addComponent(jLabel3))

 .addGap(18, 18, 18)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

 .addComponent(NameFont, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(NameStyle, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(NameSize, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

 .addComponent(jLabel4))

 .addGroup(layout.createSequentialGroup()

 .addComponent(Ok)

 .addGap(18, 18, 18)

 .addComponent(Cancel))

 .addComponent(jScrollPane3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

 .addComponent(Sampel, javax.swing.GroupLayout.PREFERRED\_SIZE, 47, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addContainerGap(109, Short.MAX\_VALUE))

 );

 pack();

 }// </editor-fold>

 private void FontStyleMouseClicked(java.awt.event.MouseEvent evt) {

 // TODO add your handling code here:

 NameStyle.setText(FontStyle.getSelectedValue().toString().trim());

 FontStyle.addListSelectionListener(new ListSelectionListener(){

 public void valueChanged(ListSelectionEvent e){

 tampil();

 }

 });

}

 private void FontNameMouseClicked(java.awt.event.MouseEvent evt) {

 // TODO add your handling code here:

 FontName.setSelectedValue("ARIAL", true);

 NameFont.setText(FontName.getSelectedValue().toString().trim());

 FontName.addListSelectionListener(new ListSelectionListener(){

 public void valueChanged(ListSelectionEvent e) {

 tampil();

 }

 });

}

 private void FontSizeMouseClicked(java.awt.event.MouseEvent evt) {

 // TODO add your handling code here:

 NameSize.setText(FontSize.getSelectedValue().toString().trim());

 FontSize.addListSelectionListener(new ListSelectionListener(){

 public void valueChanged(ListSelectionEvent e){

 tampil();

 }

 });

}

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

 // TODO add your handling code here:

 pasang();

 dispose();

 }

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

 // TODO add your handling code here:

 dispose();

 }

 private void FontSizeAncestorAdded(javax.swing.event.AncestorEvent evt) {

 // TODO add your handling code here:

 }

 /\*\*

 \* @param args the command line arguments

 \*/

 /\*public static void main(String args[]) {

 java.awt.EventQueue.invokeLater(new Runnable() {

 public void run() {

 new huruf1();

 }

 });

 }\*/

 // Variables declaration - do not modify

 private javax.swing.JButton Cancel;

 private javax.swing.JList FontName;

 private javax.swing.JList FontSize;

 private javax.swing.JList FontStyle;

 private javax.swing.JTextField NameFont;

 private javax.swing.JTextField NameSize;

 private javax.swing.JTextField NameStyle;

 private javax.swing.JButton Ok;

 private javax.swing.JTextField Sampel;

 private javax.swing.JLabel jLabel1;

 private javax.swing.JLabel jLabel2;

 private javax.swing.JLabel jLabel3;

 private javax.swing.JLabel jLabel4;

 private javax.swing.JScrollPane jScrollPane1;

 private javax.swing.JScrollPane jScrollPane2;

 private javax.swing.JScrollPane jScrollPane3;

 // End of variables declaration

 public void tampil(){

 try{

 String family = FontName.getSelectedValue().toString();

 int size = Integer.parseInt(FontSize.getSelectedValue().toString());

 int style = 0;

 if(FontStyle.getSelectedValue().toString().equals("PLAIN")){

 style = Font.PLAIN;

 }else if(FontStyle.getSelectedValue().toString().equals("BOLD")){

 style = Font.BOLD;

 }else if (FontStyle.getSelectedValue().toString().equals("ITALIC")){

 style = Font.ITALIC;

 }else{

 style = Font.BOLD + Font.ITALIC;

 }

 Font font = new Font(family, style, size);

 Sampel.setFont(font);

 }catch(NullPointerException e){

 }

 }

 public void pasang(){

 try{

 String family = FontName.getSelectedValue().toString();

 int size = Integer.parseInt(FontSize.getSelectedValue().toString());

 int style = 0;

 if(FontStyle.getSelectedValue().toString().equals("PLAIN")){

 style = Font.PLAIN;

 }else if(FontStyle.getSelectedValue().toString().equals("BOLD")){

 style = Font.BOLD;

 }else if (FontStyle.getSelectedValue().toString().equals("ITALIC")){

 style = Font.ITALIC;

 }else{

 style = Font.BOLD + Font.ITALIC;

 }

 Font font = new Font(family, style, size);

 textcomponent.setFont(font);

 }catch(NullPointerException e){

 }

 }

}

**Class font2**

/\*

 \* To change this template, choose Tools | Templates

 \* and open the template in the editor.

 \*/

/\*

 \* huruf2.java

 \*

 \* Created on Nov 5, 2010, 1:28:32 AM

 \*/

package JeditorTeks;

import java.awt.Font;

import javax.swing.JList;

import javax.swing.SwingConstants;

import javax.swing.event.ListSelectionEvent;

import javax.swing.event.ListSelectionListener;

import javax.swing.text.JTextComponent;

/\*\*

 \*

 \* @author MzArie

 \*

 \*/

public class font2 extends javax.swing.JFrame {

 /\*\* Creates new form huruf2 \*/

 private JTextComponent textcomponent;

 private JList list\_font;

 public font2(JTextComponent textcomponent) {

 initComponents();

 Sampel.setHorizontalAlignment(SwingConstants.CENTER);

 FontName.setSelectedValue("Arial", true);

 FontStyle.setSelectedValue("PLAIN", true);

 FontSize.setSelectedValue("12", true);

 this.textcomponent = textcomponent;

 }

 /\*\* 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() {

 jScrollPane3 = new javax.swing.JScrollPane();

 FontSize = new javax.swing.JList();

 NameFont = new javax.swing.JTextField();

 NameStyle = new javax.swing.JTextField();

 jLabel1 = new javax.swing.JLabel();

 NameSize = new javax.swing.JTextField();

 jLabel4 = new javax.swing.JLabel();

 jScrollPane1 = new javax.swing.JScrollPane();

 FontName = new javax.swing.JList();

 jLabel2 = new javax.swing.JLabel();

 jLabel3 = new javax.swing.JLabel();

 jScrollPane2 = new javax.swing.JScrollPane();

 FontStyle = new javax.swing.JList();

 Ok = new javax.swing.JButton();

 Cancel = new javax.swing.JButton();

 Sampel = new javax.swing.JTextField();

 setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

 setTitle("FONT");

 FontSize.setModel(new javax.swing.AbstractListModel() {

 String[] strings = { "8", "10", "12", "14", "16", "18", "20", "22", "24", "26", "28", "32", "48", "72" };

 public int getSize() { return strings.length; }

 public Object getElementAt(int i) { return strings[i]; }

 });

 FontSize.addMouseListener(new java.awt.event.MouseAdapter() {

 public void mouseClicked(java.awt.event.MouseEvent evt) {

 FontSizeMouseClicked(evt);

 }

 });

 jScrollPane3.setViewportView(FontSize);

 jLabel1.setText("Font :");

 jLabel4.setText("Sample");

 FontName.setModel(new javax.swing.AbstractListModel() {

 String[] strings = { "Agency FB", "Algerian", "Arial", "Arial Black", "Arial Narrow", "Arial Rounded MT Bold", "Arial Unicode MS", "Baskerville Old Face", "Bauhaus 93", "Bell MT", "Berlin Sans FB", "Berlin Sans FB Demi", "Bernard MT Condensed", "Blackadder ITC", "Bodoni MT", "Bodoni MT Black", "Bodoni MT Condensed", "Bodoni MT Poster Compressed", "Book Antiqua", "Bookman Old Style", "Britannic Bold", "Broadway", "Brush Script MT", "Calibri", "Californian FB", "Calisto MT", "Cambria", "Cambria Math", "Candara", "Castellar", "Centaur", "Century", "Century Gothic", "Century Schoolbook", "Chiller", "Colonna MT", "Comic Sans MS", "Consolas", "Constantia", "Cooper Black", "Copperplate Gothic Bold", "Copperplate Gothic Light", "Corbel", "Courier", "Courier New", "Curlz MT", "Edwardian Script ITC", "Elephant", "Engravers MT", "Eras Bold ITC", "Eras Demi ITC", "Eras Light ITC", "Eras Medium ITC", "Estrangelo Edessa", "Felix Titling", "Fixedsys", "Footlight MT Light", "Forte", "Franklin Gothic Book", "Franklin Gothic Demi", "Franklin Gothic Demi Cond", "Franklin Gothic Heavy", "Franklin Gothic Medium", "Franklin Gothic Medium Cond", "Freestyle Script", "French Script MT", "Garamond", "Gautami", "Georgia", "Gigi", "Gill Sans MT", "Gill Sans MT Condensed", "Gill Sans MT Ext Condensed Bold", "Gill Sans Ultra Bold", "Gill Sans Ultra Bold Condensed", "Gloucester MT Extra Condensed", "Goudy Old Style", "Goudy Stout", "Haettenschweiler", "Harlow Solid Italic", "Harrington", "High Tower Text", "Impact", "Imprint MT Shadow", "Informal Roman", "Jokerman", "Juice ITC", "Kristen ITC", "Kunstler Script", "Latha", "Lucida Bright", "Lucida Calligraphy", "Lucida Console", "Lucida Fax", "Lucida Handwriting", "Lucida Sans", "Lucida Sans Typewriter", "Lucida Sans Unicode", "Magneto", "Maiandra GD", "Mangal", "Matura MT Script Capitals", "Microsoft Sans Serif", "Mistral", "Modern", "Modern No. 20", "Monotype Corsiva", "MS Mincho", "MS Reference Sans Serif", "MS Sans Serif", "MS Serif", "Niagara Engraved", "Niagara Solid", "OCR A Extended", "Old English Text MT", "Onyx", "Palace Script MT", "Palatino Linotype", "Papyrus", "Perpetua", "Perpetua Titling MT", "Poor Richard", "Pristina", "Ravie", "Rockwell", "Rockwell Condensed", "Rockwell Extra Bold", "Roman", "Script", "Script MT Bold", "Segoe UI", "Showcard Gothic", "Small Fonts", "Snap ITC", "Stencil", "Sylfaen", "System", "Tahoma", "Tempus Sans ITC", "Terminal", "Times New Roman", "Trebuchet MS", "Tunga", "Tw Cen MT", "Tw Cen MT Condensed", "Tw Cen MT Condensed Extra Bold", "Verdana", "Viner Hand ITC", "Vivaldi", "Vladimir Script", "Wide Latin", "WST\_Czec", " " };

 public int getSize() { return strings.length; }

 public Object getElementAt(int i) { return strings[i]; }

 });

 FontName.addMouseListener(new java.awt.event.MouseAdapter() {

 public void mouseClicked(java.awt.event.MouseEvent evt) {

 FontNameMouseClicked(evt);

 }

 });

 jScrollPane1.setViewportView(FontName);

 jLabel2.setText("Font Style : ");

 jLabel3.setText("Size : ");

 FontStyle.setModel(new javax.swing.AbstractListModel() {

 String[] strings = { "PLAIN", "BOLD", "ITALIC", "BOLD ITALIC" };

 public int getSize() { return strings.length; }

 public Object getElementAt(int i) { return strings[i]; }

 });

 FontStyle.addMouseListener(new java.awt.event.MouseAdapter() {

 public void mouseClicked(java.awt.event.MouseEvent evt) {

 FontStyleMouseClicked(evt);

 }

 });

 jScrollPane2.setViewportView(FontStyle);

 Ok.setText("Ok");

 Ok.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 OkActionPerformed(evt);

 }

 });

 Cancel.setText("Cancel");

 Cancel.addActionListener(new java.awt.event.ActionListener() {

 public void actionPerformed(java.awt.event.ActionEvent evt) {

 CancelActionPerformed(evt);

 }

 });

 Sampel.setText("AABBCCDD");

 javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

 getContentPane().setLayout(layout);

 layout.setHorizontalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addContainerGap()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

 .addComponent(Sampel, javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(javax.swing.GroupLayout.Alignment.LEADING, layout.createSequentialGroup()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

 .addGroup(javax.swing.GroupLayout.Alignment.LEADING, layout.createSequentialGroup()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

 .addComponent(NameFont)

 .addComponent(jLabel1)

 .addGroup(layout.createSequentialGroup()

 .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 158, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

 .addGroup(layout.createSequentialGroup()

 .addGap(20, 20, 20)

 .addComponent(jLabel2))

 .addGroup(layout.createSequentialGroup()

 .addGap(18, 18, 18)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(NameStyle))))

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addGap(37, 37, 37)

 .addComponent(jLabel3))

 .addGroup(layout.createSequentialGroup()

 .addGap(18, 18, 18)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addComponent(jScrollPane3, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(NameSize)))))

 .addComponent(jLabel4, javax.swing.GroupLayout.Alignment.LEADING))

 .addGap(18, 18, 18)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addComponent(Ok)

 .addComponent(Cancel))))

 .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

 );

 layout.setVerticalGroup(

 layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addContainerGap()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

 .addComponent(jLabel1)

 .addComponent(jLabel2)

 .addComponent(jLabel3))

 .addGap(18, 18, 18)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

 .addComponent(NameFont, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(NameStyle, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(NameSize, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addGroup(layout.createSequentialGroup()

 .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

 .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

 .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

 .addComponent(jLabel4))

 .addComponent(jScrollPane3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))

 .addGroup(layout.createSequentialGroup()

 .addGap(44, 44, 44)

 .addComponent(Ok)

 .addGap(18, 18, 18)

 .addComponent(Cancel)))

 .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

 .addComponent(Sampel, javax.swing.GroupLayout.DEFAULT\_SIZE, 64, Short.MAX\_VALUE)

 .addContainerGap())

 );

 pack();

 }// </editor-fold>

 private void FontSizeMouseClicked(java.awt.event.MouseEvent evt) {

 // TODO add your handling code here:

 NameSize.setText(FontSize.getSelectedValue().toString().trim());

 FontSize.addListSelectionListener(new ListSelectionListener(){

 public void valueChanged(ListSelectionEvent e){

 tampil();

 }

 });

}

 private void FontNameMouseClicked(java.awt.event.MouseEvent evt) {

 // TODO add your handling code here:

 FontName.setSelectedValue("ARIAL", true);

 NameFont.setText(FontName.getSelectedValue().toString().trim());

 FontName.addListSelectionListener(new ListSelectionListener(){

 public void valueChanged(ListSelectionEvent e) {

 tampil();

 }

 });

}

 private void FontStyleMouseClicked(java.awt.event.MouseEvent evt) {

 // TODO add your handling code here:

 NameStyle.setText(FontStyle.getSelectedValue().toString().trim());

 FontStyle.addListSelectionListener(new ListSelectionListener(){

 public void valueChanged(ListSelectionEvent e){

 tampil();

 }

 });

}

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

 // TODO add your handling code here:

 pasang();

 dispose();

}

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

 // TODO add your handling code here:

 dispose();

}

 /\*\*

 \* @param args the command line arguments

 \*/

 /\*public static void main(String args[]) {

 java.awt.EventQueue.invokeLater(new Runnable() {

 public void run() {

 new huruf2().setVisible(true);

 }

 });

 }\*/

 // Variables declaration - do not modify

 private javax.swing.JButton Cancel;

 private javax.swing.JList FontName;

 private javax.swing.JList FontSize;

 private javax.swing.JList FontStyle;

 private javax.swing.JTextField NameFont;

 private javax.swing.JTextField NameSize;

 private javax.swing.JTextField NameStyle;

 private javax.swing.JButton Ok;

 private javax.swing.JTextField Sampel;

 private javax.swing.JLabel jLabel1;

 private javax.swing.JLabel jLabel2;

 private javax.swing.JLabel jLabel3;

 private javax.swing.JLabel jLabel4;

 private javax.swing.JScrollPane jScrollPane1;

 private javax.swing.JScrollPane jScrollPane2;

 private javax.swing.JScrollPane jScrollPane3;

 // End of variables declaration

 public void tampil(){

 try{

 String family = FontName.getSelectedValue().toString();

 int size = Integer.parseInt(FontSize.getSelectedValue().toString());

 int style = 0;

 if(FontStyle.getSelectedValue().toString().equals("PLAIN")){

 style = Font.PLAIN;

 }else if(FontStyle.getSelectedValue().toString().equals("BOLD")){

 style = Font.BOLD;

 }else if (FontStyle.getSelectedValue().toString().equals("ITALIC")){

 style = Font.ITALIC;

 }else{

 style = Font.BOLD + Font.ITALIC;

 }

 Font font = new Font(family, style, size);

 Sampel.setFont(font);

 }catch(NullPointerException e){

 }

 }

 public void pasang(){

 try{

 String family = FontName.getSelectedValue().toString();

 int size = Integer.parseInt(FontSize.getSelectedValue().toString());

 int style = 0;

 if(FontStyle.getSelectedValue().toString().equals("PLAIN")){

 style = Font.PLAIN;

 }else if(FontStyle.getSelectedValue().toString().equals("BOLD")){

 style = Font.BOLD;

 }else if (FontStyle.getSelectedValue().toString().equals("ITALIC")){

 style = Font.ITALIC;

 }else{

 style = Font.BOLD + Font.ITALIC;

 }

 Font font = new Font(family, style, size);

 textcomponent.setFont(font);

 }catch(NullPointerException e){

 }

 }

}