```
// http://docs.oracle.com/javase/7/docs/api/java/util/package-summary.html
// http://docs.oracle.com/javase/7/docs/api/java/io/package-summary.html
// http://docs.oracle.com/javase/7/docs/api/java/awt/package-summary.html



package editortexto;

import java.awt.Color;
import java.awt.Font;
import java.awt.GraphicsEnvironment;
import java.awt.datatransfer.Clipboard;
import java.awt.datatransfer.DataFlavor;
import java.awt.datatransfer.StringSelection;
import java.awt.datatransfer.Transferable;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.font.TextAttribute;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Map;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JColorChooser;
import javax.swing.JFileChooser;
import javax.swing.JLabel;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;
import javax.swing.filechooser.FileView;


public class EditorTexto extends javax.swing.JFrame {
    Clipboard portapapeles = getToolkit().getSystemClipboard();
   //Fuente for defecto
    private Color color = Color.BLACK;
    private String tipo = "Times";
    private int tamanio = 12;
    private int estilo = Font.PLAIN;
    private JColorChooser colorChooser = new JColorChooser();

    public EditorTexto() {

        initComponents();

        //tamaños de fuente: botón y evento
        for (int i = 10; i <= 50; i=i+2){
            //Creación de item de tamaño
            final JMenuItem TAMANIO_FUENTE = new JMenuItem();
            TAMANIO_FUENTE.setText(""+i);
            //Evento asociado al item
            TAMANIO_FUENTE.addActionListener(new ActionListener(){
                public void actionPerformed (ActionEvent ae) {
                    int tamanioFuente = Integer.parseInt(TAMANIO_FUENTE.getText());
                    String tipoFuente = areaDeTexto.getFont().getFontName();
                    int estiloFuente = areaDeTexto.getFont().getStyle();
                    Font f = new Font(tipoFuente, estiloFuente, tamanioFuente);
                    areaDeTexto.setFont(f);
                }
            });
            //Adición del item a la lista de tamaños de Fuentes y del ComboBox
            barraSuperiorFuenteTamanio.add(TAMANIO_FUENTE);
            barraInferiorTamanioFuente.addItem(TAMANIO_FUENTE.getText());

        }

        //tipos de fuente: botón y evento
        String [] listaDeFuentes =
GraphicsEnvironment.getLocalGraphicsEnvironment().getAvailableFontFamilyNames();
        for (int i=0; i <listaDeFuentes.length; i++){
            final JMenuItem TIPO_FUENTE = new JMenuItem();
            TIPO_FUENTE.setText(listaDeFuentes[i]);
            TIPO_FUENTE.addActionListener(new ActionListener(){
                public void actionPerformed (ActionEvent ae) {
                    int estiloFuente = areaDeTexto.getFont().getStyle();
                    int tamanioFuente = areaDeTexto.getFont().getSize();
                    Font f = new Font(TIPO_FUENTE.getText(), estiloFuente, tamanioFuente);
                    areaDeTexto.setFont(f);
                }
            });
            barraSuperiorFuenteTipo.add(TIPO_FUENTE);
        }

        //no es correcto esto aquí, pero lo tengo que poner para que la fuente por defecto no salga enana.
        //se puede ver el problema comentando esta línea.
        barraInferiorTamanioFuente.setSelectedIndex(3);

    }//Fin Constructor


    /**
     * 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() {
        java.awt.GridBagConstraints gridBagConstraints;

        barraInferior = new javax.swing.JToolBar();
        barraInferiorAbrir = new javax.swing.JButton();
        barraInferiorGuardar = new javax.swing.JButton();
        barraInferiorNegrita = new javax.swing.JToggleButton();
        barraInferiorCursiva = new javax.swing.JToggleButton();
        barraInferiorNormal = new javax.swing.JToggleButton();
        barraInferiorSubrayar = new javax.swing.JButton();
        barraInferiorTamanioFuente = new javax.swing.JComboBox();
        barraInferiorColorFuente = new javax.swing.JToggleButton();
        barraInferiorColorFondo = new javax.swing.JToggleButton();
        barraInferiorCortar = new javax.swing.JToggleButton();
        barraInferiorCopiar = new javax.swing.JToggleButton();
```

```
        barraInferiorPegar = new javax.swing.JToggleButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        areaDeTexto = new javax.swing.JTextArea();
        barraSuperior = new javax.swing.JMenuBar();
        barraSuperiorArchivo = new javax.swing.JMenu();
        barraSuperiorArchivoAbrir = new javax.swing.JMenuItem();
        barraSuperiorArchivoGuardar = new javax.swing.JMenuItem();
        barraSuperiorArchivoSalir = new javax.swing.JMenuItem();
        barraSuperiorEdicion = new javax.swing.JMenu();
        barraSuperiorEdicionCortar = new javax.swing.JMenuItem();
        barraSuperiorEdicionCopiar = new javax.swing.JMenuItem();
        barraSuperiorEdicionPegar = new javax.swing.JMenuItem();
        barraSuperiorEdicionColorFondo = new javax.swing.JMenuItem();
        barraSuperiorFuente = new javax.swing.JMenu();
        jMenu3 = new javax.swing.JMenu();
        barraSuperiorFuenteNegrita = new javax.swing.JRadioButtonMenuItem();
        barraSuperiorFuenteCursiva = new javax.swing.JRadioButtonMenuItem();
        barraSuperiorFuenteSubrayado = new javax.swing.JRadioButtonMenuItem();
        barraSuperiorFuenteTachado = new javax.swing.JRadioButtonMenuItem();
        barraSuperiorFuenteNormal = new javax.swing.JRadioButtonMenuItem();
        barraSuperiorFuenteTipo = new javax.swing.JMenu();
        barraSuperiorFuenteTamanio = new javax.swing.JMenu();
        barraSuperiorFuenteColor = new javax.swing.JMenuItem();
        barraSuperiorInfo = new javax.swing.JMenu();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setTitle("Editor de Texto");
        getContentPane().setLayout(new java.awt.GridBagLayout());

        barraInferior.setBorder(javax.swing.BorderFactory.createEtchedBorder());
        barraInferior.setFloatable(false);
        barraInferior.setRollover(true);

        barraInferiorAbrir.setIcon(new javax.swing.ImageIcon(getClass().getResource("/editortexto/images/abrir.jpg")));
// NOI18N
        barraInferiorAbrir.setText("Abrir");
        barraInferiorAbrir.setFocusable(false);
        barraInferiorAbrir.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        barraInferiorAbrir.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
        barraInferiorAbrir.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraInferiorAbrirActionPerformed(evt);
            }
        });
        barraInferior.add(barraInferiorAbrir);

        barraInferiorGuardar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/guardar.jpg"))); // NOI18N
        barraInferiorGuardar.setText("Guardar");
        barraInferiorGuardar.setFocusable(false);
        barraInferiorGuardar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        barraInferiorGuardar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
        barraInferiorGuardar.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraInferiorGuardarActionPerformed(evt);
            }
        });
        barraInferior.add(barraInferiorGuardar);

        barraInferiorNegrita.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/negrita.png"))); // NOI18N
        barraInferiorNegrita.setText("Negrita");
        barraInferiorNegrita.setFocusable(false);
        barraInferiorNegrita.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        barraInferiorNegrita.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
        barraInferiorNegrita.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraInferiorNegritaActionPerformed(evt);
            }
        });
        barraInferior.add(barraInferiorNegrita);

        barraInferiorCursiva.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/cursiva3.jpg"))); // NOI18N
        barraInferiorCursiva.setText("Cursiva");
        barraInferiorCursiva.setFocusable(false);
        barraInferiorCursiva.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        barraInferiorCursiva.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
        barraInferiorCursiva.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraInferiorCursivaActionPerformed(evt);
            }
        });
        barraInferior.add(barraInferiorCursiva);

        barraInferiorNormal.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/normal.png"))); // NOI18N
        barraInferiorNormal.setText("Normal");
        barraInferiorNormal.setFocusable(false);
        barraInferiorNormal.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        barraInferiorNormal.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
        barraInferiorNormal.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraInferiorNormalActionPerformed(evt);
            }
        });
        barraInferior.add(barraInferiorNormal);

        barraInferiorSubrayar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/subrayar.jpg"))); // NOI18N
        barraInferiorSubrayar.setText("Subrayar");
        barraInferiorSubrayar.setFocusable(false);
        barraInferiorSubrayar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        barraInferiorSubrayar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
        barraInferiorSubrayar.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraInferiorSubrayarActionPerformed(evt);
            }
        });
        barraInferior.add(barraInferiorSubrayar);

        barraInferiorTamanioFuente.setMaximumRowCount(20);
        barraInferiorTamanioFuente.setToolTipText("Tamaño de Fuente");
        barraInferiorTamanioFuente.addActionListener(new java.awt.event.ActionListener() {
```

```
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraInferiorTamanioFuenteActionPerformed(evt);
      }
    });
    barraInferior.add(barraInferiorTamanioFuente);

    barraInferiorColorFuente.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/colorFuente2.jpg"))); // NOI18N
    barraInferiorColorFuente.setText("ColorTexto");
    barraInferiorColorFuente.setFocusable(false);
    barraInferiorColorFuente.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
    barraInferiorColorFuente.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
    barraInferiorColorFuente.addActionListener(new java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraInferiorColorFuenteActionPerformed(evt);
      }
    });
    barraInferior.add(barraInferiorColorFuente);

    barraInferiorColorFondo.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/colorFondo.png"))); // NOI18N
    barraInferiorColorFondo.setText("Fondo");
    barraInferiorColorFondo.setFocusable(false);
    barraInferiorColorFondo.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
    barraInferiorColorFondo.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
    barraInferiorColorFondo.addActionListener(new java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraInferiorColorFondoActionPerformed(evt);
      }
    });
    barraInferior.add(barraInferiorColorFondo);

    barraInferiorCortar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/cortar.jpg"))); // NOI18N
    barraInferiorCortar.setText("Cortar");
    barraInferiorCortar.setFocusable(false);
    barraInferiorCortar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
    barraInferiorCortar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
    barraInferiorCortar.addActionListener(new java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraInferiorCortarActionPerformed(evt);
      }
    });
    barraInferior.add(barraInferiorCortar);

    barraInferiorCopiar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/copiar2.jpg"))); // NOI18N
    barraInferiorCopiar.setText("Copiar");
    barraInferiorCopiar.setFocusable(false);
    barraInferiorCopiar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
    barraInferiorCopiar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
    barraInferiorCopiar.addActionListener(new java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraInferiorCopiarActionPerformed(evt);
      }
    });
    barraInferior.add(barraInferiorCopiar);
```

```
        barraInferiorPegar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/editortexto/images/pegar.jpg"))); // NOI18N
        barraInferiorPegar.setText("Pegar");
        barraInferiorPegar.setFocusable(false);
        barraInferiorPegar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        barraInferiorPegar.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
        barraInferiorPegar.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraInferiorPegarActionPerformed(evt);
            }
        });
        barraInferior.add(barraInferiorPegar);

        gridBagConstraints = new java.awt.GridBagConstraints();
        gridBagConstraints.gridx = 0;
        gridBagConstraints.gridy = 0;
        gridBagConstraints.gridwidth = 2;
        gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;
        gridBagConstraints.ipadx = 163;
        gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;
        getContentPane().add(barraInferior, gridBagConstraints);

        areaDeTexto.setColumns(20);
        areaDeTexto.setFont(areaDeTexto.getFont().deriveFont((float)14));
        areaDeTexto.setLineWrap(true);
        areaDeTexto.setRows(5);
        areaDeTexto.setCursor(new java.awt.Cursor(java.awt.Cursor.TEXT_CURSOR));
        jScrollPane1.setViewportView(areaDeTexto);

        gridBagConstraints = new java.awt.GridBagConstraints();
        gridBagConstraints.gridx = 0;
        gridBagConstraints.gridy = 1;
        gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;
        gridBagConstraints.ipadx = 653;
        gridBagConstraints.ipady = 348;
        gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;
        gridBagConstraints.weightx = 1.0;
        gridBagConstraints.weighty = 1.0;
        gridBagConstraints.insets = new java.awt.Insets(11, 0, 11, 0);
        getContentPane().add(jScrollPane1, gridBagConstraints);

        barraSuperiorArchivo.setText("Archivo");
        barraSuperiorArchivo.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraSuperiorArchivoActionPerformed(evt);
            }
        });

        barraSuperiorArchivoAbrir.setText("Abrir");
        barraSuperiorArchivoAbrir.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                barraSuperiorArchivoAbrirActionPerformed(evt);
            }
        });
        barraSuperiorArchivo.add(barraSuperiorArchivoAbrir);
```

```
barraSuperiorArchivoGuardar.setText("Guardar");
barraSuperiorArchivoGuardar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraSuperiorArchivoGuardarActionPerformed(evt);
    }
});
barraSuperiorArchivo.add(barraSuperiorArchivoGuardar);

barraSuperiorArchivoSalir.setText("Salir");
barraSuperiorArchivoSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraSuperiorArchivoSalirActionPerformed(evt);
    }
});
barraSuperiorArchivo.add(barraSuperiorArchivoSalir);

barraSuperior.add(barraSuperiorArchivo);

barraSuperiorEdicion.setText("Edición");

barraSuperiorEdicionCortar.setText("Cortar");
barraSuperiorEdicionCortar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraSuperiorEdicionCortarActionPerformed(evt);
    }
});
barraSuperiorEdicion.add(barraSuperiorEdicionCortar);

barraSuperiorEdicionCopiar.setText("Copiar");
barraSuperiorEdicionCopiar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraSuperiorEdicionCopiarActionPerformed(evt);
    }
});
barraSuperiorEdicion.add(barraSuperiorEdicionCopiar);

barraSuperiorEdicionPegar.setText("Pegar");
barraSuperiorEdicionPegar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraSuperiorEdicionPegarActionPerformed(evt);
    }
});
barraSuperiorEdicion.add(barraSuperiorEdicionPegar);

barraSuperiorEdicionColorFondo.setText("Color de Fondo");
barraSuperiorEdicionColorFondo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        barraSuperiorEdicionColorFondoActionPerformed(evt);
    }
});
barraSuperiorEdicion.add(barraSuperiorEdicionColorFondo);

barraSuperior.add(barraSuperiorEdicion);

barraSuperiorFuente.setText("Fuente");

jMenu3.setText("Estilo");
```

```
barraSuperiorFuenteNegrita.setText("Negrita");
barraSuperiorFuenteNegrita.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      barraSuperiorFuenteNegritaActionPerformed(evt);
   }
});
jMenu3.add(barraSuperiorFuenteNegrita);

barraSuperiorFuenteCursiva.setText("Cursiva");
barraSuperiorFuenteCursiva.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      barraSuperiorFuenteCursivaActionPerformed(evt);
   }
});
jMenu3.add(barraSuperiorFuenteCursiva);

barraSuperiorFuenteSubrayado.setText("Subrayado");
barraSuperiorFuenteSubrayado.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      barraSuperiorFuenteSubrayadoActionPerformed(evt);
   }
});
jMenu3.add(barraSuperiorFuenteSubrayado);

barraSuperiorFuenteTachado.setText("Tachado");
barraSuperiorFuenteTachado.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      barraSuperiorFuenteTachadoActionPerformed(evt);
   }
});
jMenu3.add(barraSuperiorFuenteTachado);

barraSuperiorFuenteNormal.setText("Normal");
barraSuperiorFuenteNormal.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      barraSuperiorFuenteNormalActionPerformed(evt);
   }
});
jMenu3.add(barraSuperiorFuenteNormal);

barraSuperiorFuente.add(jMenu3);

barraSuperiorFuenteTipo.setText("Tipo");
barraSuperiorFuente.add(barraSuperiorFuenteTipo);

barraSuperiorFuenteTamanio.setText("Tamaño");
barraSuperiorFuente.add(barraSuperiorFuenteTamanio);

barraSuperiorFuenteColor.setText("Color");
barraSuperiorFuenteColor.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      barraSuperiorFuenteColorActionPerformed(evt);
   }
});
barraSuperiorFuente.add(barraSuperiorFuenteColor);
```

```
    barraSuperior.add(barraSuperiorFuente);

    barraSuperiorInfo.setText("Info");
    barraSuperiorInfo.addMouseListener(new java.awt.event.MouseAdapter() {
        public void mouseClicked(java.awt.event.MouseEvent evt) {
            barraSuperiorInfoMouseClicked(evt);
        }
    });
    barraSuperior.add(barraSuperiorInfo);

    setJMenuBar(barraSuperior);

    pack();
}// </editor-fold>

private void barraSuperiorFuenteCursivaActionPerformed(java.awt.event.ActionEvent evt) {
    String tipoFuente = areaDeTexto.getFont().getFontName();
    int tamanioFuente = areaDeTexto.getFont().getSize();
    Font fuente = new Font (tipoFuente, Font.ITALIC, tamanioFuente);
    areaDeTexto.setFont(fuente);

}

private void barraSuperiorArchivoAbrirActionPerformed(java.awt.event.ActionEvent evt) {
    JFileChooser file = new JFileChooser();
    JLabel nombreFichero = new JLabel();
    FileView view = new JavaFileView();
    file.setFileView(view);
    file.addChoosableFileFilter(new FiltroTxt());
    int opcion = file.showOpenDialog(this);
    if (opcion == JFileChooser.APPROVE_OPTION) {
        try {
            nombreFichero.setText(file.getSelectedFile().getName());
            FileReader reader = null;
            reader = new FileReader(file.getSelectedFile().getAbsoluteFile());
            try {
                areaDeTexto.read(reader, file.getSelectedFile().getName());
            } catch (IOException ex) {
                Logger.getLogger(EditorTexto.class.getName()).log(Level.SEVERE, null, ex);}
        } catch (FileNotFoundException ex) {
            Logger.getLogger(EditorTexto.class.getName()).log(Level.SEVERE, null, ex);}
    } else {
        nombreFichero.setText("");
    }
}

private void barraSuperiorArchivoGuardarActionPerformed(java.awt.event.ActionEvent evt) {
    FileWriter writer = null;
    JFileChooser filechooser = new JFileChooser();
    filechooser.showSaveDialog(this);
    if (filechooser.getSelectedFile() != null){ //si no se pulsa Cancelar
        String file = filechooser.getSelectedFile().getAbsolutePath()+".txt";
        try {
            writer = new FileWriter(file);
            areaDeTexto.write(writer);
        } catch (IOException exception) {
            System.err.println("Error al guardar"); exception.printStackTrace();
        } finally {
            if (writer != null) {
                try { writer.close();
                } catch (IOException exception) {
                    System.err.println("Error closing writer"); exception.printStackTrace();}
            }
        }//try
    }//if
}

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

private void barraSuperiorArchivoSalirActionPerformed(java.awt.event.ActionEvent evt) {
    if (JOptionPane.showConfirmDialog(this, "¿Desea Salir?", "Salir", JOptionPane.YES_NO_OPTION) == 0) {
        this.dispose();
        System.exit(0);
    }
}

private void barraSuperiorEdicionCortarActionPerformed(java.awt.event.ActionEvent evt) {
    StringSelection ts = new StringSelection(areaDeTexto.getSelectedText());
    portapapeles.setContents(ts, ts);
    areaDeTexto.replaceRange("", areaDeTexto.getSelectionStart(), areaDeTexto.getSelectionEnd());
}

private void barraSuperiorEdicionCopiarActionPerformed(java.awt.event.ActionEvent evt) {
    StringSelection ts = new StringSelection(areaDeTexto.getSelectedText());
    portapapeles.setContents(ts, ts);
}

private void barraSuperiorEdicionPegarActionPerformed(java.awt.event.ActionEvent evt) {
    Transferable tf = portapapeles.getContents(this);
    try {
        String cadena = "";
        cadena = (String) tf.getTransferData(DataFlavor.stringFlavor);
        areaDeTexto.insert(cadena, areaDeTexto.getSelectionStart());
    } catch (Exception ee) {
    }
}

private void barraSuperiorEdicionColorFondoActionPerformed(java.awt.event.ActionEvent evt) {
    areaDeTexto.setBackground(colorChooser.showDialog(this, "Elija color de fondo", color));
}

private void barraSuperiorFuenteNegritaActionPerformed(java.awt.event.ActionEvent evt) {
    String tipoFuente = areaDeTexto.getFont().getFontName();
    int tamanioFuente = areaDeTexto.getFont().getSize();
    Font fuente = new Font (tipoFuente, Font.BOLD, tamanioFuente);
    areaDeTexto.setFont(fuente);
}

private void barraSuperiorFuenteSubrayadoActionPerformed(java.awt.event.ActionEvent evt) {
    Map atributosFuente = areaDeTexto.getFont().getAttributes();
    atributosFuente.put(TextAttribute.UNDERLINE,TextAttribute.UNDERLINE_ON);
    Font fuente = new Font(atributosFuente);
    areaDeTexto.setFont(fuente);
}

private void barraSuperiorFuenteNormalActionPerformed(java.awt.event.ActionEvent evt) {
    String tipoFuente = areaDeTexto.getFont().getFontName();
    int tamanioFuente = areaDeTexto.getFont().getSize();
    Font fuente = new Font (tipo, Font.PLAIN, tamanioFuente);
    areaDeTexto.setFont(fuente);
}

private void barraSuperiorFuenteTachadoActionPerformed(java.awt.event.ActionEvent evt) {
    Map atributosFuente = areaDeTexto.getFont().getAttributes();
    atributosFuente.put(TextAttribute.STRIKETHROUGH, TextAttribute.STRIKETHROUGH_ON);
    Font fuente = new Font(atributosFuente);
    areaDeTexto.setFont(fuente);
}

private void barraSuperiorFuenteColorActionPerformed(java.awt.event.ActionEvent evt) {
    areaDeTexto.setForeground(colorChooser.showDialog(this, "Color de Fuente",color));
}

private void barraSuperiorInfoMouseClicked(java.awt.event.MouseEvent evt) {
    String mensaje = "\n\n"+
                "Aplicación: Editor de Texto\n"+
                "Fecha: Noviembre 2014\n"+
                "Autor: Verónica Santamaría\n\n";
    JOptionPane.showMessageDialog(this, mensaje,"Información",JOptionPane.INFORMATION_MESSAGE);
}

private void barraInferiorAbrirActionPerformed(java.awt.event.ActionEvent evt) {
    barraSuperiorArchivoAbrir.doClick();
}

private void barraInferiorGuardarActionPerformed(java.awt.event.ActionEvent evt) {
    barraSuperiorArchivoGuardar.doClick();
}

private void barraInferiorNegritaActionPerformed(java.awt.event.ActionEvent evt) {
    barraSuperiorFuenteNegrita.doClick();
}

private void barraInferiorCursivaActionPerformed(java.awt.event.ActionEvent evt) {
    barraSuperiorFuenteCursiva.doClick();
}

private void barraInferiorNormalActionPerformed(java.awt.event.ActionEvent evt) {
    barraSuperiorFuenteNormal.doClick();
}

private void barraInferiorTamanioFuenteActionPerformed(java.awt.event.ActionEvent evt) {
    Object item = barraInferiorTamanioFuente.getSelectedItem();
    int tamanioFuente = Integer.parseInt(item.toString());
    String tipoFuente = areaDeTexto.getFont().getFontName();
    int estiloFuente = areaDeTexto.getFont().getStyle();
    Font f = new Font(tipoFuente, estiloFuente, tamanioFuente);
    areaDeTexto.setFont(f);
```

```
}

private void barraInferiorColorFuenteActionPerformed(java.awt.event.ActionEvent evt) {
   barraSuperiorFuenteColor.doClick();
}

private void barraInferiorColorFondoActionPerformed(java.awt.event.ActionEvent evt) {
   barraSuperiorEdicionColorFondo.doClick();
}

private void barraInferiorCortarActionPerformed(java.awt.event.ActionEvent evt) {
   barraSuperiorEdicionCortar.doClick();
}

private void barraInferiorCopiarActionPerformed(java.awt.event.ActionEvent evt) {
   barraSuperiorEdicionCopiar.doClick();
}

private void barraInferiorPegarActionPerformed(java.awt.event.ActionEvent evt) {
   barraSuperiorEdicionPegar.doClick();
}

private void barraInferiorSubrayarActionPerformed(java.awt.event.ActionEvent evt) {
   barraSuperiorFuenteSubrayado.doClick();
}


class JavaFileView extends FileView {
   public String getName(File file) {
      String filename = file.getName();
      if (filename.endsWith(".java")) {
         String name = filename + " : " + file.length();
         return name;
      }
      return null;}
}//FileView

class FiltroTxt extends javax.swing.filechooser.FileFilter {
   public boolean accept(java.io.File f) {
      if (f.isDirectory()) {
         return false;}
      String nombre = f.getName().toLowerCase();
      if (nombre != null) {
         if (nombre.endsWith(".txt")) {
            return true; }}
      return false;
   }
   public String getDescription() {
      return "*.txt ";}
}//FiltroTxt

public static void main(String args[]) {
   /* Set the Nimbus look and feel */
   //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
   /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
    * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
```

```
  try {
    for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
      if ("Nimbus".equals(info.getName())) {
        javax.swing.UIManager.setLookAndFeel(info.getClassName());
        break;
      }
    }
  } catch (ClassNotFoundException ex) {
    java.util.logging.Logger.getLogger(EditorTexto.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
  } catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(EditorTexto.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
  } catch (IllegalAccessException ex) {
    java.util.logging.Logger.getLogger(EditorTexto.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
  } catch (javax.swing.UnsupportedLookAndFeelException ex) {
    java.util.logging.Logger.getLogger(EditorTexto.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
  }
  //</editor-fold>

  /* Create and display the form */
  java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
      new EditorTexto().setVisible(true);
    }
  });
}//Fin MAIN

// Variables declaration - do not modify
private javax.swing.JTextArea areaDeTexto;
private javax.swing.JToolBar barraInferior;
private javax.swing.JButton barraInferiorAbrir;
private javax.swing.JToggleButton barraInferiorColorFondo;
private javax.swing.JToggleButton barraInferiorColorFuente;
private javax.swing.JToggleButton barraInferiorCopiar;
private javax.swing.JToggleButton barraInferiorCortar;
private javax.swing.JToggleButton barraInferiorCursiva;
private javax.swing.JButton barraInferiorGuardar;
private javax.swing.JToggleButton barraInferiorNegrita;
private javax.swing.JToggleButton barraInferiorNormal;
private javax.swing.JToggleButton barraInferiorPegar;
private javax.swing.JButton barraInferiorSubrayar;
private javax.swing.JComboBox barraInferiorTamanioFuente;
private javax.swing.JMenuBar barraSuperior;
private javax.swing.JMenu barraSuperiorArchivo;
private javax.swing.JMenuItem barraSuperiorArchivoAbrir;
private javax.swing.JMenuItem barraSuperiorArchivoGuardar;
private javax.swing.JMenuItem barraSuperiorArchivoSalir;
private javax.swing.JMenu barraSuperiorEdicion;
private javax.swing.JMenuItem barraSuperiorEdicionColorFondo;
private javax.swing.JMenuItem barraSuperiorEdicionCopiar;
private javax.swing.JMenuItem barraSuperiorEdicionCortar;
private javax.swing.JMenuItem barraSuperiorEdicionPegar;
private javax.swing.JMenu barraSuperiorFuente;
private javax.swing.JMenuItem barraSuperiorFuenteColor;
private javax.swing.JRadioButtonMenuItem barraSuperiorFuenteCursiva;
private javax.swing.JRadioButtonMenuItem barraSuperiorFuenteNegrita;
private javax.swing.JRadioButtonMenuItem barraSuperiorFuenteNormal;
private javax.swing.JRadioButtonMenuItem barraSuperiorFuenteSubrayado;
```

```
    private javax.swing.JRadioButtonMenuItem barraSuperiorFuenteTachado;
    private javax.swing.JMenu barraSuperiorFuenteTamanio;
    private javax.swing.JMenu barraSuperiorFuenteTipo;
    private javax.swing.JMenu barraSuperiorInfo;
    private javax.swing.JMenu jMenu3;
    private javax.swing.JScrollPane jScrollPane1;
    // End of variables declaration


}//FIN EditorTexto
```