



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**AD2 de Programação III**

**2º semestre de 2007**

**Nome:**

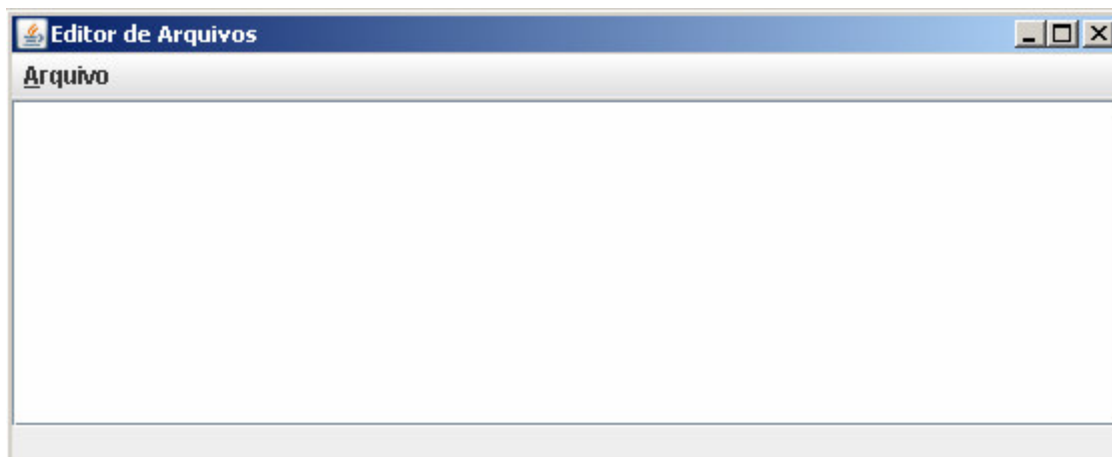
**Matrícula:**

**Pólo:**

**Exercício:**

Escreva um editor de arquivos simplificado (com as funcionalidades inerentes a um aplicativo que serve para editar textos) em Java, como mostrado na figura abaixo. Este editor pode ser encarado como o aplicativo **Notepad** do Sistema Operacional Windows, somente com algumas opções do Menu **Arquivo**.

Quando você clicar no menu **Arquivo** do seu aplicativo deve ser possível: abrir um arquivo novo (opção **Novo**), abrir um arquivo existente (opção **Abrir**), salvar (opção **Salvar**), salvar com outro nome (opção **Salvar como**) e sair do aplicativo (opção **Sair**).



```
import java.awt.BorderLayout;  
import java.awt.Font;  
import java.awt.Image;  
import java.awt.Toolkit;  
import java.awt.event.ActionEvent;  
import java.awt.Dimension;  
import java.awt.event.ActionListener;  
import java.io.*;  
import javax.swing.*;
```

```

public class Editor extends JFrame {

    JMenuBar menuPrincipal;
    JMenu mnuArquivo;
    JMenuItem mnuNovo, mnuAbrir, mnuSalvar, mnuSalvarComo, mnuSair;
    JTextArea txt;
    File file;
    StatusBar stb;
    JScrollPane scroll;
    String textoSalvo = new String("");

    MenuListener ml = new MenuListener(this);

    public Editor(){
        menuPrincipal = new JMenuBar();
        mnuArquivo = new JMenu("Arquivo");
        mnuNovo = new JMenuItem("Novo");
        mnuAbrir = new JMenuItem("Abrir...");
        mnuSalvar = new JMenuItem("Salvar");
        mnuSalvarComo = new JMenuItem("Salvar como...");
        mnuSair = new JMenuItem("Sair");

        this.setJMenuBar(menuPrincipal);
        menuPrincipal.add(mnuArquivo);
        mnuArquivo.add(mnuNovo);
        mnuArquivo.add(mnuAbrir);
        mnuArquivo.add(mnuSalvar);
        mnuArquivo.add(mnuSalvarComo);
        mnuArquivo.add(new JSeparator());
        mnuArquivo.add(mnuSair);

        mnuSalvar.setEnabled(false);

        mnuNovo.addActionListener(ml);
        mnuAbrir.addActionListener(ml);
        mnuSalvar.addActionListener(ml);
        mnuSalvarComo.addActionListener(ml);
        mnuSair.addActionListener(ml);

        JPanel painel = new JPanel(new BorderLayout());
        txt = new JTextArea();
        scroll = new JScrollPane(txt);
        txt.setBorder(BorderFactory.createEmptyBorder(5, 5, 5, 5));
        txt.setFont(new Font("Dialog", Font.PLAIN, 14));

        painel.add(scroll, BorderLayout.CENTER);
        stb = new StatusBar();
        painel.add(stb, java.awt.BorderLayout.SOUTH);
        this.getContentPane().add(painel);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        this.setTitle("Editor de Arquivos");
        this.setSize(500, 500);
        this.setVisible(true);
    }
}

```

```

    public void saiPrograma() {
        if (!textoSalvo.equals(txt.getText())) {
            int retorno = JOptionPane.showConfirmDialog(this, "Arquivo '" +
                ((file != null) ? file.getName() : "Sem titulo") + "' não foi salvo!
Deseja sair assim mesmo?");
            if (retorno != JOptionPane.YES_OPTION) {
                return;
            }
        }

        System.exit(0);
    }

    public static void setNativeLookAndFeel() {
        try {
            UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
        } catch (Exception e) {
            System.out.println("Error setting native LAF: " + e);
        }
    }

    public static void main(String args[]) {
        setNativeLookAndFeel();
        Editor edit = new Editor();
    }

    class MenuListener implements ActionListener {
        Editor ed;
        final JFileChooser fc = new JFileChooser();

        public MenuListener(Editor editor) {
            ed = editor;
            fc.setFileFilter(new FiltroTXT());
        }

        public String leArquivo(File f) {
            try {
                FileReader fr = new FileReader(f);
                BufferedReader br = new BufferedReader(fr);
                String linha, retorno = "";
                while ((linha = br.readLine()) != null) {
                    retorno += linha + "\n";
                }
                br.close();
                fr.close();
                return retorno;
            } catch (IOException e) {
                JOptionPane.showMessageDialog(ed, "Erro de E/S");
                return null;
            }
        }

        public void escreveArquivo(File f, String texto) {
            try {
                FileWriter fw = new FileWriter(f);
                BufferedWriter bw = new BufferedWriter(fw);
            }
        }
    }

```

```

        texto = texto.replaceAll("\n", "\r\n");
        bw.write(texto);
        bw.close();
        fw.close();
    } catch (IOException e) {
        JOptionPane.showMessageDialog(ed, "Erro de E/S");
    }
}

public void actionPerformed(ActionEvent e) {

    int retorno=0;
    JMenuItem mi = (JMenuItem)e.getSource();

    if (mi.getText().equals("Novo")) {
        ed.txt.setText("");
        ed.mnuSalvar.setEnabled(false);
        ed.stb.setText("Arquivo: Sem Nome");
    }

    else if (mi.getText().equals("Abrir...")) {
        retorno = fc.showOpenDialog(ed);

        if (retorno == JFileChooser.APPROVE_OPTION) {
            ed.file = fc.getSelectedFile();
            ed.txt.setText(leArquivo(ed.file));
            ed.mnuSalvar.setEnabled(true);
            ed.stb.setText("Arquivo: " + ed.file.getName());
            ed.txt.setSelectionStart(0);
            ed.txt.setSelectionEnd(0);
            ed.textoSalvo = new String(ed.txt.getText());
        }
    }

    else if (mi.getText().equals("Salvar")) {
        escreveArquivo(ed.file, ed.txt.getText());
        ed.textoSalvo = new String(ed.txt.getText());
    }

    else if (mi.getText().equals("Salvar como...")) {
        retorno = fc.showSaveDialog(ed);

        if (retorno == JFileChooser.APPROVE_OPTION) {
            ed.file = fc.getSelectedFile();
            if (ed.file.exists()) {
                retorno = JOptionPane.showConfirmDialog(ed, "Arquivo '" +
ed.file.getName() + "' já existe! Deseja sobrescrever?");
                if (retorno != JOptionPane.YES_OPTION) {
                    return;
                }
            }
        }

        escreveArquivo(ed.file, ed.txt.getText());
        ed.mnuSalvar.setEnabled(true);
        ed.textoSalvo = new String(ed.txt.getText());
        ed.stb.setText("Arquivo: " + ed.file.getName());
    }
}

```

```

        }

        else if (mi.getText().equals("Sair")) {
            ed.saiPrograma();
        }
    }
}

class FiltroTXT extends javax.swing.filechooser.FileFilter {
    public boolean accept(File f) {
        return (f.getName().toLowerCase().endsWith(".txt") ||
f.getName().toLowerCase().endsWith(".html"));
    }

    public String getDescription() {
        return "Arquivos .txt ou .html";
    }
}

class StatusBar extends javax.swing.JLabel {

    public StatusBar() {
        super();
        super.setPreferredSize(new Dimension(100, 16));
        setMessage("Pronto");
    }

    public void setMessage(String message) {
        setText(" "+message);
    }
}

```