



Curso: T-ADS

Turma: 3º B - NOTURNO

Matéria: ESTRUTURA DE DADOS

Aluno: ARTHUR SILVA BERDUSCO DE SOUZA

ADO - 1: Desenvolver uma aplicação utilizando a linguagem JAVA e os conceitos da Aula 02 e Aula 03, sobre Array e manipulação de dados.

São Paulo, 31 de Agosto de 2023

► **Desenvolver uma aplicação utilizando a linguagem JAVA e os conceitos da Aula 02 e Aula 03, sobre Array e manipulação de dados.**

► **Descrever a situação e/ou problema (tema livre);**

Nesta ADO, desenvolvi um sistema de gestão de tarefas. O sistema será responsável por realizar o agendamento de atividades de acordo com data, hora e descrição.

Cada atividade que for inserida, terá um controle de verificação de status automático, podendo ser:

- Atrasada
- Pendente
- Em_Andamento
- Concluída (Poderá ser selecionado pelo usuário quando for concluída)

Além de realizar o agendamento, poderá também ser feita a consulta das tarefas agendadas, edição e exclusão se for necessário.

► **Criar uma Classe Vetor com todos os métodos para manipular os dados;**

► **Criar uma Classe para testar a aplicação, possibilitando a interação com o usuário, para manipular os dados;**

O sistema foi desenvolvido utilizando a biblioteca Swing do Java, onde foi criado o layout de interação do usuário.

A interface busca ser simples, objetiva e satisfatório para o usuário.

Obs: No projeto há um ícone do Senac em jpg e uma dependência externa "JCalendar" que foi implementada para o uso do DataChooser "Seletor de data".

O projeto foi desenvolvido em 5 classes:

1. Main //Teste do sistema
2. Tarefa //Modelagem dos métodos e atributos de uma tarefa
3. ListaTarefas //Responsável pelo gerenciamento das tarefas
4. TelaPrincipal //Classe Swing responsável pela interface principal
5. PainelTarefa // Classe Swing responsável pelo formulário de tarefa

Classes:

Main.java

```
package com.mycompany.tarefasagendadas;

import javax.swing.*;

public class Main {

    public static void main(String[] args) {

        try {

            // Define o Look and Feel para Nimbus

            UIManager.setLookAndFeel("javax.swing.plaf.nimbus.NimbusLookAndFeel");

        } catch (Exception e) {

            e.printStackTrace();

        }

        TelaPrincipal sistema = new TelaPrincipal();

        sistema.setVisible(true);

    }

}
```

Tarefa.java

```
package com.mycompany.tarefasagendadas;

import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;

public class Tarefa {

    private int id = 0;

    private String titulo;

    private String data;

    private String horainicio;

    private String horaFim;

    private String descricao;

    private Status status;

    public void atualizarStatus() {

        Calendar agora = Calendar.getInstance();

        Date dataHoraAtual = agora.getTime();

        // Formata a data e hora atual para comparação

        SimpleDateFormat formatoDataHora = new SimpleDateFormat("dd/MM/yyyy HH:mm");

        String dataHoraAtualStr = formatoDataHora.format(dataHoraAtual);

        // Verifica o status com base nas comparações

        if(status == Status.CONCLUIDO){

            return;

        }

        if (dataHoraAtualStr.compareTo(data + " " + horaFim) > 0) {

            status = Status.ATRASADA;
```

```

    } else if (dataHoraAtualStr.compareTo(data + " " + horaInicio) >= 0
        && dataHoraAtualStr.compareTo(data + " " + horaFim) <= 0) {

        status = Status.EM_ANDAMENTO;

    } else {

        status = Status.PENDENTE; // Talvez você queira adicionar um status "pendente" para tarefas futuras

    }

}

```

```

public void concluirStatus(){

    status = Status.CONCLUIDO;

}

```

```

public Status getStatus() {

    return status;

}

```

```

public Tarefa() {

}

```

```

public Tarefa(int id, String titulo, String data, String horaInicio, String horaFim, String descricao) {

    this.id = id;

    this.titulo = titulo;

    this.data = data;

    this.horaInicio = horaInicio;

    this.horaFim = horaFim;

    this.descricao = descricao;

}

```

```

public int getId() {

    return id;

}

```

```

public String getTitulo() {

    return titulo;

}

```

```
}
```

```
public String getData() {
```

```
    return data;
```

```
}
```

```
public String getHoralInicio() {
```

```
    return horalnicio;
```

```
}
```

```
public String getHoraFim() {
```

```
    return horaFim;
```

```
}
```

```
public String getDescricao() {
```

```
    return descricao;
```

```
}
```

```
public void setId(int id) {
```

```
    this.id = id;
```

```
}
```

```
public void setTitulo(String titulo) {
```

```
    this.titulo = titulo;
```

```
}
```

```
public void setData(String data) {
```

```
    this.data = data;
```

```
}
```

```
public void setHoralInicio(String horalnicio) {
```

```
    this.horalnicio = horalnicio;
```

```
}
```

```
public void setHoraFim(String horaFim) {
```

```
        this.horaFim = horaFim;
    }

    public void setDescricao(String descricao) {

        this.descricao = descricao;
    }

    @Override

    public String toString() {

        return "Tarefa{" + "id=" + id + ", titulo=" + titulo + ", data=" + data + ", horainicio=" + horainicio + ", horaFim=" + horaFim + ", descricao=" + descricao + '}';
    }

    public enum Status {

        EM_ANDAMENTO,

        CONCLUIDO,

        ATRASADA,

        PENDENTE
    }
}
```

ListaTarefas.java

```
package com.mycompany.tarefasagendadas;

import java.util.Scanner;

import java.util.Arrays;

public class ListaTarefas {

    private Tarefa[] tarefas;

    private int id = 1;

    public int getId() {

        return id;

    }

    public Tarefa[] getTarefas() {

        return tarefas;

    }

    public void setTarefas(Tarefa[] tarefas) {

        this.tarefas = tarefas;

    }

    private int tamanho = 0;

    public ListaTarefas() {

        this.tarefas = new Tarefa[5];

    }

    public void editarTarefa(Tarefa tarefaAtualizada){

        for(Tarefa tarefa: getTarefas()){

            if(tarefa.getId() == tarefaAtualizada.getId()){

                tarefa = tarefaAtualizada;

                return;

            }

        }

    }

}
```



```
    }  
}
```

```
public void adicionar(Tarefa tarefa) throws Exception {  
  
    this.aumentaCapacidade();  
  
    if (this.tamanho < tarefas.length) {  
  
        tarefa.setId(this.getId());  
  
        this.tarefas[this.tamanho] = tarefa;  
  
        this.tamanho++;  
  
        this.id++;  
  
    } else {  
  
        throw new Exception("O vetor está cheio, não é possível adicionar novos tarefas");  
  
    }  
}
```

```
public boolean adicionaNaPosicao(int posicao, Tarefa elemento) throws Exception {  
  
    this.aumentaCapacidade();  
  
    if (posicao >= 0 && posicao < this.tamanho) {  
  
        for (int i = this.tamanho; i > posicao; i--) {  
  
            this.tarefas[i + 1] = this.tarefas[i];  
  
        }  
  
        this.tarefas[posicao] = elemento;  
  
        this.tamanho++;  
  
    } else {  
  
        throw new Exception("Posição Inválida!");  
  
    }  
  
    return true;  
}
```

```
public Tarefa buscarTarefa(int posicao) throws Exception {  
  
    if (posicao >= 0 && posicao < tamanho) {  
  
        return tarefas[posicao];  
  
    } else {  
  
        throw new Exception("Posição de busca não encontrada");  
  
    }  
}
```

```
}
```

```
public int buscarPosicao(Tarefa elemento) {
```

```
    for (int i = 0; i < this.tamanho; i++) {
```

```
        if (tarefas[i].equals(elemento)) {
```

```
            return i;
```

```
        }
```

```
    }
```

```
    return -1;
```

```
}
```

```
public void remover(int posicao) throws Exception {
```

```
    if (posicao >= 0 && posicao < this.tamanho) {
```

```
        for (int i = posicao; i < this.tamanho - 1; i++) {
```

```
            this.tarefas[i] = this.tarefas[i + 1];
```

```
        }
```

```
        this.tamanho--;
```

```
    } else {
```

```
        throw new Exception("Posição do array inválida");
```

```
    }
```

```
}
```

```
public int getTamanho() {
```

```
    return this.tamanho;
```

```
}
```

```
@Override
```

```
public String toString() {
```

```
    StringBuilder s = new StringBuilder();
```

```
    s.append("[");
```

```
    for (int i = 0; i < this.tamanho - 1; i++) {
```

```
        s.append(this.tarefas[i]);
```

```
        s.append(", ");
```

```
    }
```

```
        if (this.tamanho > 0) {

            s.append(this.tarefas[this.tamanho - 1]);

        }

        s.append("]");

        return s.toString();

    }

    public void aumentaCapacidade() {

        if (this.tamanho == this.tarefas.length) {

            Tarefa[] tarefasNovos = new Tarefa[this.tarefas.length * 2];

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

                tarefasNovos[i] = this.tarefas[i];

            }

            this.tarefas = tarefasNovos;

        }

    }

}
```

TelaPrincipal.java

```
package com.mycompany.tarefasagendadas;

import java.awt.Point;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.table.DefaultTableModel;

public class TelaPrincipal extends javax.swing.JFrame {

    ListaTarefas tarefas = new ListaTarefas();

    public TelaPrincipal() {

        initComponents();

    }

    public void fecharTarefa(JPanel janela, Tarefa tarefa) throws Exception {

        if (tarefa.getId() == 0) {

            tarefa.setId(this.tarefas.getTamanho() + 1);

            tarefas.adicionar(tarefa);

        } else {

            tarefas.editarTarefa(tarefa);

        }

        atualizarTabelaTarefas();

        tabPainel.remove(janela);

    }

    public void fecharTarefa(JPanel janela) {

        tabPainel.remove(janela);

    }

}
```

```
}
```

```
public void atualizarTabelaTarefas() {  
  
    DefaultTableModel modelo = (DefaultTableModel) tblTarefas.getModel();  
  
    modelo.setRowCount(0);  
  
    for (int i = 0; i < tarefas.getTamanho(); i++) {  
  
        tarefas.getTarefas()[i].atualizarStatus();  
  
        modelo.addRow(new Object[]{  
  
            tarefas.getTarefas()[i].getId(),  
  
            tarefas.getTarefas()[i].getTitulo(),  
  
            tarefas.getTarefas()[i].getData(),  
  
            tarefas.getTarefas()[i].getHoralInicio(),  
  
            tarefas.getTarefas()[i].getHoraFim(),  
  
            tarefas.getTarefas()[i].getStatus(),  
  
        });  
  
    }  
}
```

```
@SuppressWarnings("unchecked")  
  
// <editor-fold defaultstate="collapsed" desc="Generated Code">  
  
private void initComponents() {  
  
    jPanel1 = new javax.swing.JPanel();  
  
    tabPainel = new javax.swing.JTabbedPane();  
  
    jScrollPane1 = new javax.swing.JScrollPane();  
  
    tblTarefas = new javax.swing.JTable();  
  
    jPanel2 = new javax.swing.JPanel();  
  
    btnVerTarefa = new javax.swing.JButton();  
  
    btnVerTarefa1 = new javax.swing.JButton();  
  
    btnAgendar = new javax.swing.JButton();  
  
    iconSenac = new javax.swing.JLabel();  
  
    jLabel1 = new javax.swing.JLabel();  
  
    jLabel2 = new javax.swing.JLabel();  

```

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
setTitle("Agenda de Tarefas");
```

```
setResizable(false);
```

```
jPanel1.setBackground(new java.awt.Color(51, 51, 51));
```

```
jPanel1.setForeground(new java.awt.Color(16, 16, 16));
```

```
tabPainel.setBorder(javax.swing.BorderFactory.createEtchedBorder());
```

```
tblTarefas.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
```

```
tblTarefas.setModel(new javax.swing.table.DefaultTableModel(  
    new Object [][] {  
  

```

```
    },  
    new String [] {  
  

```

```
        "ID", "Título", "Dia", "Hora Início", "Hora Fim", "Status"  
    }  
    ) {  
  

```

```
        boolean[] canEdit = new boolean [] {  
  

```

```
            false, false, false, false, false, false  
        };  
  

```

```
        public boolean isCellEditable(int rowIndex, int columnIndex) {  
  

```

```
            return canEdit [columnIndex];  
  

```

```
    });  
  

```

```
tblTarefas.setRowHeight(30);
```

```
tblTarefas.getColumnModel().getColumn(0).setPreferredWidth(10);
```

```
tblTarefas.getColumnModel().getColumn(1).setPreferredWidth(120);
```

```
tblTarefas.getColumnModel().getColumn(2).setPreferredWidth(50);
```

```
tblTarefas.getColumnModel().getColumn(3).setPreferredWidth(40);
```

```
tblTarefas.getColumnModel().getColumn(4).setPreferredWidth(40);
```

```
jScrollPane1.setViewportViewView(tblTarefas);
```

```
tabPainel.addTab("Agenda", jScrollPane1);
```

```
jPanel2.setBorder(javax.swing.BorderFactory.createEtchedBorder());
```

```
btnVerTarefa.setBackground(new java.awt.Color(51, 102, 255));
```

```
btnVerTarefa.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
```

```
btnVerTarefa.setText("Consultar");
```

```
btnVerTarefa.addActionListener(new java.awt.event.ActionListener() {
```

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

```
        btnVerTarefaActionPerformed(evt);
```

```
    }
```

```
});
```

```
btnVerTarefa1.setBackground(new java.awt.Color(51, 102, 255));
```

```
btnVerTarefa1.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
```

```
btnVerTarefa1.setText("Deletar");
```

```
btnVerTarefa1.addActionListener(new java.awt.event.ActionListener() {
```

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

```
        btnVerTarefa1ActionPerformed(evt);
```

```
    }
```

```
});
```

```
btnAgendar.setBackground(new java.awt.Color(51, 102, 255));
```

```
btnAgendar.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
```

```
btnAgendar.setText("Adicionar");
```

```
btnAgendar.addActionListener(new java.awt.event.ActionListener() {
```

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

```
        btnAgendarActionPerformed(evt);
```

```
    }
```

```
});
```

```
iconSenac.setIcon(new javax.swing.ImageIcon(getClass().getResource("/senac2.png"))); // NOI18N
```

```
jLabel1.setText("Desenvolvido por:");
```

```

jLabel2.setFont(new java.awt.Font("Arial", 2, 12)); // NOI18N

jLabel2.setText("Arthur Silva Berdusco");


javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

    jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel2Layout.createSequentialGroup()

            .addGap(12, 12, 12)

            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addGroup(jPanel2Layout.createSequentialGroup()

                    .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                    .addContainerGap())

                .addGroup(jPanel2Layout.createSequentialGroup()

                    .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.CENTER)

                        .addComponent(iconSenac)

                        .addComponent(btnAgendar, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

                        .addComponent(btnVerTarefa, javax.swing.GroupLayout.DEFAULT_SIZE, 129, Short.MAX_VALUE)

                        .addComponent(btnVerTarefa1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

                    .addGroup(jPanel2Layout.createSequentialGroup()

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

                        .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))

                    .addContainerGap(12, Short.MAX_VALUE)))

            .addContainerGap());

jPanel2Layout.setVerticalGroup(

    jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel2Layout.createSequentialGroup()

            .addGap(16, 16, 16)

            .addComponent(iconSenac)

            .addGap(40, 40, 40)

            .addComponent(btnAgendar, javax.swing.GroupLayout.PREFERRED_SIZE, 35, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGap(30, 30, 30)

            .addComponent(btnVerTarefa, javax.swing.GroupLayout.PREFERRED_SIZE, 35, javax.swing.GroupLayout.PREFERRED_SIZE)

```



```

        .addGap(30, 30, 30)

        .addComponent(btnVerTarefa1,                javax.swing.GroupLayout.PREFERRED_SIZE,                35,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,    javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

        .addComponent(jLabel1)

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

        .addComponent(jLabel2)

        .addGap(15, 15, 15))

    );

    javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(
        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel2,                javax.swing.GroupLayout.PREFERRED_SIZE,                javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(tabPainel, javax.swing.GroupLayout.PREFERRED_SIZE, 624, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                    .addComponent(tabPainel, javax.swing.GroupLayout.DEFAULT_SIZE, 488, Short.MAX_VALUE)
                    .addComponent(jPanel2,                javax.swing.GroupLayout.DEFAULT_SIZE,                javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
    );

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

```

```

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

        .addComponent(jPanel1,          javax.swing.GroupLayout.DEFAULT_SIZE,          javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

    );

    layout.setVerticalGroup(

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

        .addComponent(jPanel1,          javax.swing.GroupLayout.DEFAULT_SIZE,          javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

    );

    pack();

    setLocationRelativeTo(null);

} // </editor-fold>

```

```

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

```

```

    if (tabPainel.getTabCount() > 5) {

        JOptionPane.showMessageDialog(this, "Você está com muitas abas em andamento, finalize uma para abrir outra.");

        return;

    }

```

```

    PainelTarefa painelTarefa = new PainelTarefa(this);

    tabPainel.addTab("Nova Tarefa", painelTarefa);

    tabPainel.setSelectedComponent(painelTarefa);

}

```

```

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

```

```

    if (tblTarefas.getSelectedRow() != -1) {

        if (tabPainel.getTabCount() > 5) {

            JOptionPane.showMessageDialog(this, "Você está com muitas abas em andamento, finalize uma para abrir outra.");

            return;

        }


```

```

        String idTarefa = String.valueOf(tblTarefas.getValueAt(tblTarefas.getSelectedRow(), 0));

```

```

int idSelecionado = Integer.parseInt(idTarefa);

for (int i = 0; i < tarefas.getTamanho(); i++) {

    if (tarefas.getTarefas()[i].getId() == idSelecionado) {

        PaineiTarefa paineITarefa = new PaineiTarefa(this, tarefas.getTarefas()[i]);

        tabPainel.addTab("Tarefa: " + tarefas.getTarefas()[i].getId(), paineITarefa);

        tabPainel.setSelectedComponent(paineITarefa);

        tblTarefas.clearSelection();

    }

} else {

    JOptionPane.showMessageDialog(this, "Selecione uma tarefa.");

}

}

private void btnVerTarefa1ActionPerformed(java.awt.event.ActionEvent evt) {

    int id = Integer.parseInt(tblTarefas.getValueAt(tblTarefas.getSelectedRow(), 0).toString());

    for(int i = 0; i < tarefas.getTamanho(); i++){

        if(tarefas.getTarefas()[i].getId() == id){

            try {

                tarefas.remover(i);

            } catch (Exception ex) {

                JOptionPane.showMessageDialog(this, "Não foi possível remover tarefa");

            }

        }

    }

    atualizarTabelaTarefas();

}

/**
 * @param args the command line arguments
 */

```

```

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(TelaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

        java.util.logging.Logger.getLogger(TelaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (IllegalAccessException ex) {

        java.util.logging.Logger.getLogger(TelaPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

        java.util.logging.Logger.getLogger(TelaPrincipal.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 TelaPrincipal().setVisible(true);

        }

    });

}

// Variables declaration - do not modify

private javax.swing.JButton btnAgendar;

private javax.swing.JButton btnVerTarefa;

```

```
private javax.swing.JButton btnVerTarefa1;  
  
private javax.swing.JLabel iconSenac;  
  
private javax.swing.JLabel jLabel1;  
  
private javax.swing.JLabel jLabel2;  
  
private javax.swing.JPanel jPanel1;  
  
private javax.swing.JPanel jPanel2;  
  
private javax.swing.JScrollPane jScrollPane1;  
  
private javax.swing.JTabbedPane tabPainel;  
  
private javax.swing.JTable tblTarefas;  
  
// End of variables declaration  
  
}
```

PainelTarefa.java

```
package com.mycompany.tarefasagendadas;

import java.util.Date;

import javax.swing.JSpinner;

import javax.swing.SpinnerDateModel;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.logging.Level;

import java.util.logging.Logger;

import com.mycompany.tarefasagendadas.Tarefa;

import java.util.Locale;

import javax.swing.JOptionPane;

public class PainelTarefa extends javax.swing.JPanel {

    TelaPrincipal jf;

    Tarefa tarefa;

    public PainelTarefa() {

        initComponents();

    }

    public PainelTarefa(TelaPrincipal jf) {

        initComponents();

        this.jf = jf;

    }

    public boolean camposValidos() throws Exception {

        if (txtTitulo.getText().trim().isEmpty()) {

            new Exception("Campo título não pode ser nulo");

            return false;

        }

    }

}
```

```

    }

    if (dtcData.getDate() == null) {

        new Exception("Campo data não pode ser nulo");

        return false;

    }

    if (spnInicio.getValue() == null) {

        new Exception("Campo hora início não pode ser nulo");

        return false;

    }

    if (spnFim.getValue() == null) {

        new Exception("Campo hora fim não pode ser nulo");

        return false;

    }

    return true;

}

public PaineiTarefa(TelaPrincipal jf, Tarefa tarefa) {

    this.jf = jf;

    this.tarefa = tarefa;

    initComponents();

    SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");

    try {

        Date data = df.parse(tarefa.getData());

        txtTitulo.setText(tarefa.getTitulo());

        dtcData.setDate(data);

        SimpleDateFormat timeFormat = new SimpleDateFormat("HH:mm");

        Date horainicio = timeFormat.parse(tarefa.getHoralInicio());

        spnInicio.setValue(horainicio);

        Date horaFim = timeFormat.parse(tarefa.getHoraFim());

        spnFim.setValue(horaFim);

        txtDescricao.setText(tarefa.getDescricao());

```

```

        cboStatus.setSelectedItem(tarefa.getStatus());

    } catch (Exception ex) {

        ex.printStackTrace();

    }

}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel4 = new javax.swing.JLabel();

    txtTitulo = new javax.swing.JTextField();

    jLabel5 = new javax.swing.JLabel();

    Date date = new Date();

    SpinnerDateModel sm = new SpinnerDateModel(date, null, null, Calendar.HOUR_OF_DAY);

    spnInicio = new javax.swing.JSpinner(sm);

    dtcData = new com.toedter.calendar.JDateChooser();

    jLabel1 = new javax.swing.JLabel();

    SpinnerDateModel sm2 = new SpinnerDateModel(date, null, null, Calendar.HOUR_OF_DAY);

    spnFim = new javax.swing.JSpinner(sm2);

    jLabel2 = new javax.swing.JLabel();

    jLabel3 = new javax.swing.JLabel();

    btnSalvar = new javax.swing.JButton();

    jScrollPane2 = new javax.swing.JScrollPane();

    txtDescricao = new javax.swing.JTextArea();

    btnSalvar1 = new javax.swing.JButton();

    cboStatus = new javax.swing.JComboBox<>();

    jLabel4.setFont(new java.awt.Font("Serif", 0, 14)); // NOI18N

    jLabel4.setText("Título:");

    txtTitulo.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N

```



```

jLabel5.setFont(new java.awt.Font("Serif", 0, 14)); // NOI18N

jLabel5.setText("Descrição:");


JSpinner.DateEditor de = new JSpinner.DateEditor(spnInicio, "HH:mm");

spnInicio.setEditor(de);

spnInicio.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N


dtcData.setLocale(new Locale("pt", "BR"));

dtcData.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N

dtcData.setDate(Calendar.getInstance().getTime());


jLabel1.setFont(new java.awt.Font("Serif", 0, 14)); // NOI18N

jLabel1.setText("Data da tarefa:");


JSpinner.DateEditor de2 = new JSpinner.DateEditor(spnFim, "HH:mm");

spnFim.setEditor(de2);

spnFim.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N


jLabel2.setFont(new java.awt.Font("Serif", 0, 14)); // NOI18N

jLabel2.setText("Horário de início:");


jLabel3.setFont(new java.awt.Font("Serif", 0, 14)); // NOI18N

jLabel3.setText("Horário de fim:");


btnSalvar.setBackground(new java.awt.Color(51, 102, 255));

btnSalvar.setFont(new java.awt.Font("Arial", 0, 12)); // NOI18N

btnSalvar.setText("Salvar e fechar");

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

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

        btnSalvarActionPerformed(evt);

    }

});


txtDescricao.setColumns(20);

txtDescricao.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N

```

[illegible]

```

        .addGroup(layout.createSequentialGroup())

        .addComponent(jLabel1)

        .addGap(0, 0, Short.MAX_VALUE))

        .addComponent(dtcData, javax.swing.GroupLayout.DEFAULT_SIZE, 202, Short.MAX_VALUE))

    .addGap(18, 18, 18)

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

        .addComponent(jLabel2)

        .addComponent(spnInicio,                javax.swing.GroupLayout.PREFERRED_SIZE,                150,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(18, 18, 18)

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

            .addComponent(jLabel3)

            .addComponent(spnFim,                javax.swing.GroupLayout.PREFERRED_SIZE,                150,
javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addGroup(layout.createSequentialGroup())

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

            .addComponent(jLabel4,                javax.swing.GroupLayout.PREFERRED_SIZE,                88,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(jLabel5,                javax.swing.GroupLayout.PREFERRED_SIZE,                65,
javax.swing.GroupLayout.PREFERRED_SIZE))

            .addGap(0, 0, Short.MAX_VALUE)))

    .addContainerGap())

);

layout.setVerticalGroup(

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

    .addGroup(layout.createSequentialGroup())

        .addContainerGap()

        .addComponent(jLabel4)

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

        .addComponent(txtTitulo, javax.swing.GroupLayout.PREFERRED_SIZE, 30, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(34, 34, 34)

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

            .addGroup(layout.createSequentialGroup())

                .addComponent(jLabel1)

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

                .addComponent(dtcData, javax.swing.GroupLayout.PREFERRED_SIZE, 30, javax.swing.GroupLayout.PREFERRED_SIZE))

            .addGroup(layout.createSequentialGroup())

```

```

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

            .addComponent(jLabel2)

            .addComponent(jLabel3))

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

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

            .addComponent(spnInicio,                javax.swing.GroupLayout.PREFERRED_SIZE,        30,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(spnFim,                javax.swing.GroupLayout.PREFERRED_SIZE,        30,
javax.swing.GroupLayout.PREFERRED_SIZE))))

        .addGap(42, 42, 42)

        .addComponent(jLabel5)

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

        .addComponent(jScrollPane2, javax.swing.GroupLayout.DEFAULT_SIZE, 119, Short.MAX_VALUE)

        .addGap(18, 18, 18)

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

            .addComponent(btnSalvar, javax.swing.GroupLayout.PREFERRED_SIZE, 41, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(btnSalvar1, javax.swing.GroupLayout.PREFERRED_SIZE, 41, javax.swing.GroupLayout.PREFERRED_SIZE)

            .addComponent(cboStatus, javax.swing.GroupLayout.PREFERRED_SIZE, 41, javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGap(20, 20, 20))

    );
} // </editor-fold>

```

```

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

```

```

    try {

```

```

        if (camposValidos()) {

```

```

            if (this.tarefa == null) {

```

```

                this.tarefa = new Tarefa();

```

```

            }

```

```

            this.tarefa.setTitulo(txtTitulo.getText()); //OK

```

```

            this.tarefa.setDescricao(txtDescricao.getText()); // OK

```

```

            //DATA TAREFA: OK

```

```

SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");

String data = df.format(dtcData.getDate());

this.tarefa.setData(data);


SimpleDateFormat inputFormat = new SimpleDateFormat("EEE MMM dd HH:mm:ss zzz yyyy");

SimpleDateFormat outputFormat = new SimpleDateFormat("HH:mm");

Date dataInicio = inputFormat.parse(spnInicio.getValue().toString());

String horaInicioFormatada = outputFormat.format(dataInicio);

this.tarefa.setHoraInicio(horaInicioFormatada);


Date dataFim = inputFormat.parse(spnFim.getValue().toString());

String horaFimFormatada = outputFormat.format(dataFim);

this.tarefa.setHoraFim(horaFimFormatada);


if(cboStatus.getSelectedIndex() == 2){

    tarefa.concluirStatus();

}


this.jf.fecharTarefa(this, tarefa);

} else {

    JOptionPane.showMessageDialog(this, "Preencha todos os campos.");

}


} catch (Exception ex) {

    JOptionPane.showMessageDialog(this, "Revise os campos e tente novamente");

}

}

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

    this.jf.fecharTarefa(this);

}


// Variables declaration - do not modify

private javax.swing.JButton btnSalvar;

```

```
private javax.swing.JButton btnSalvar1;  
  
private javax.swing.JComboBox<String> cboStatus;  
  
private com.toedter.calendar.JDateChooser dtcData;  
  
private javax.swing.JLabel jLabel1;  
  
private javax.swing.JLabel jLabel2;  
  
private javax.swing.JLabel jLabel3;  
  
private javax.swing.JLabel jLabel4;  
  
private javax.swing.JLabel jLabel5;  
  
private javax.swing.JScrollPane jScrollPane2;  
  
private javax.swing.JSpinner spnFim;  
  
private javax.swing.JSpinner spnInicio;  
  
private javax.swing.JTextArea txtDescricao;  
  
private javax.swing.JTextField txtTitulo;  
  
// End of variables declaration  
}
```