

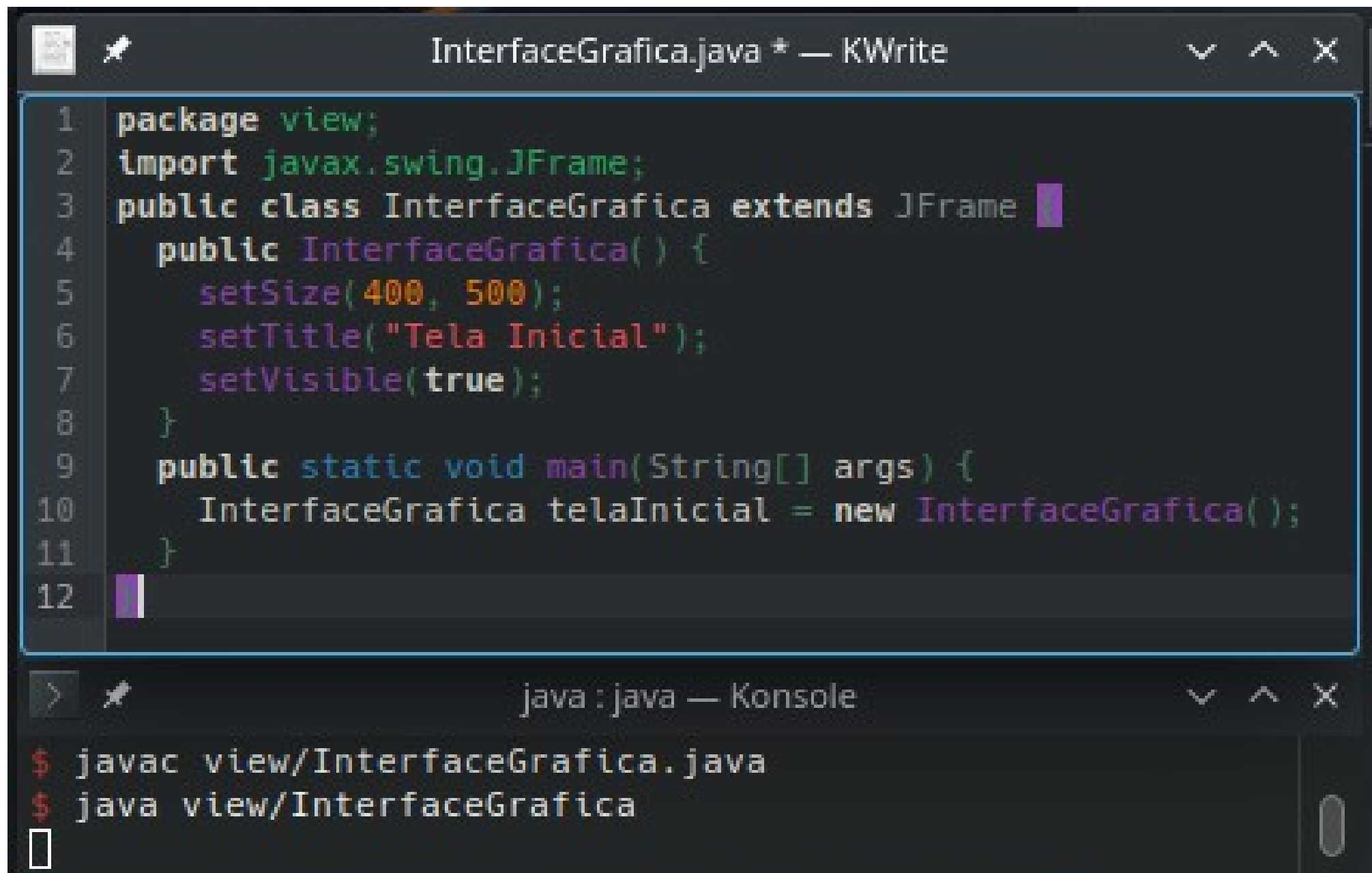
# Desenvolvimento de interfaces gráficas na linguagem Java

Programação Orientada a Objetos II

- Interface gráfica (GUI) de um sistema
  - É um dos elementos mais impactantes para
    - Aceitação do sistema pelo cliente
    - Validação pelos usuários
- Necessita de grande dedicação
  - Por meio delas o usuário percebe e interage com o sistema
- Criação inicial de uma interface gráfica utilizando a biblioteca Swing do Java

- Inserção de componentes em uma tela
  - Botões
  - Caixas de seleção
  - A própria representação da tela
  - E outros
- Primeiro elemento a ser utilizado em uma GUI
  - Classe JFrame
  - Representação de uma área
    - Apresenta uma barra de título
      - E um espaço reservado para se adicionar componentes

# exemplo



The image shows a screenshot of a code editor and a terminal window. The code editor, titled "InterfaceGrafica.java \* — KWrite", contains the following Java code:

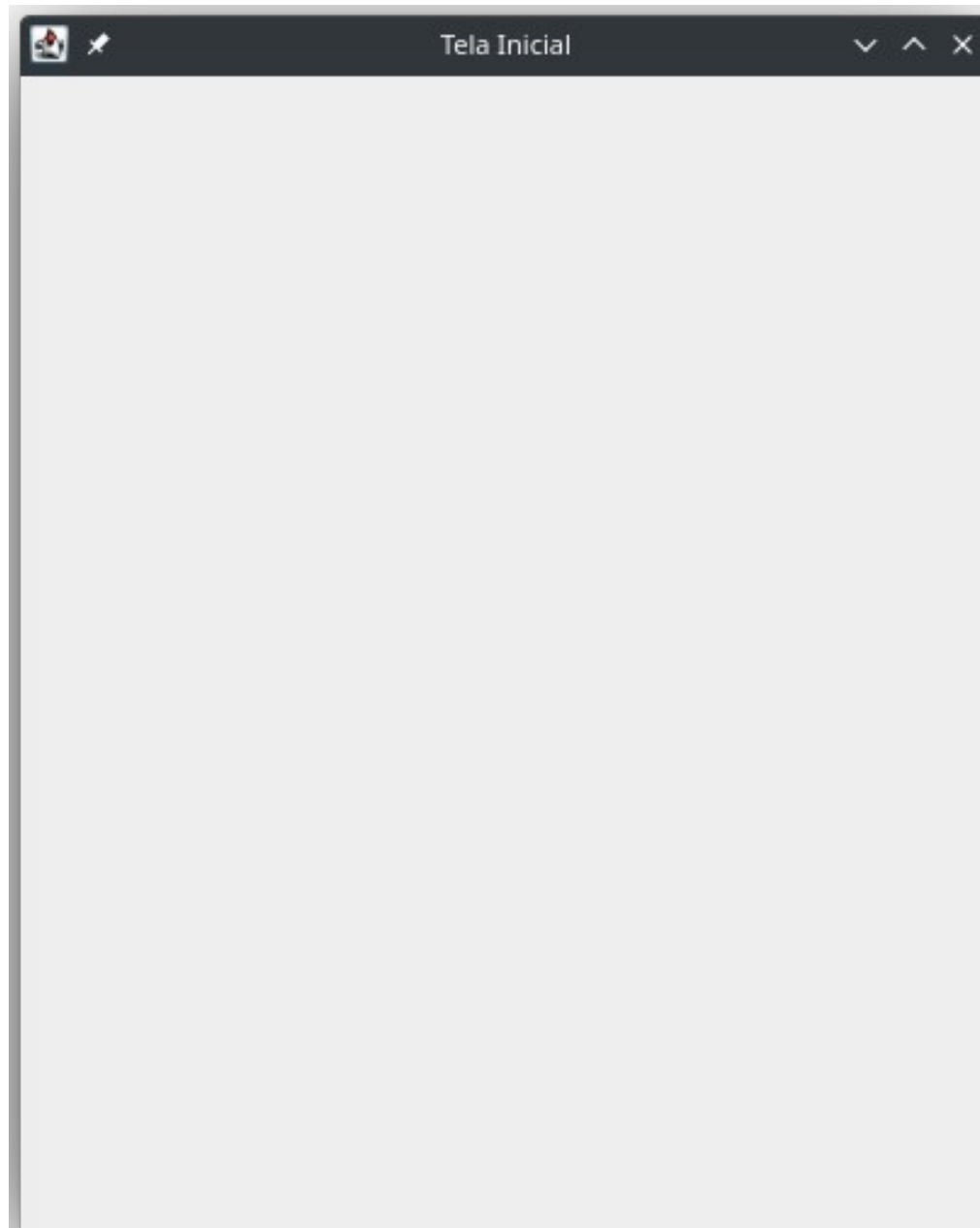
```
1 package view;
2 import javax.swing.JFrame;
3 public class InterfaceGrafica extends JFrame {
4     public InterfaceGrafica() {
5         setSize(400, 500);
6         setTitle("Tela Inicial");
7         setVisible(true);
8     }
9     public static void main(String[] args) {
10         InterfaceGrafica telaInicial = new InterfaceGrafica();
11     }
12
```

The terminal window, titled "java : java — Konsole", shows the following commands and their output:

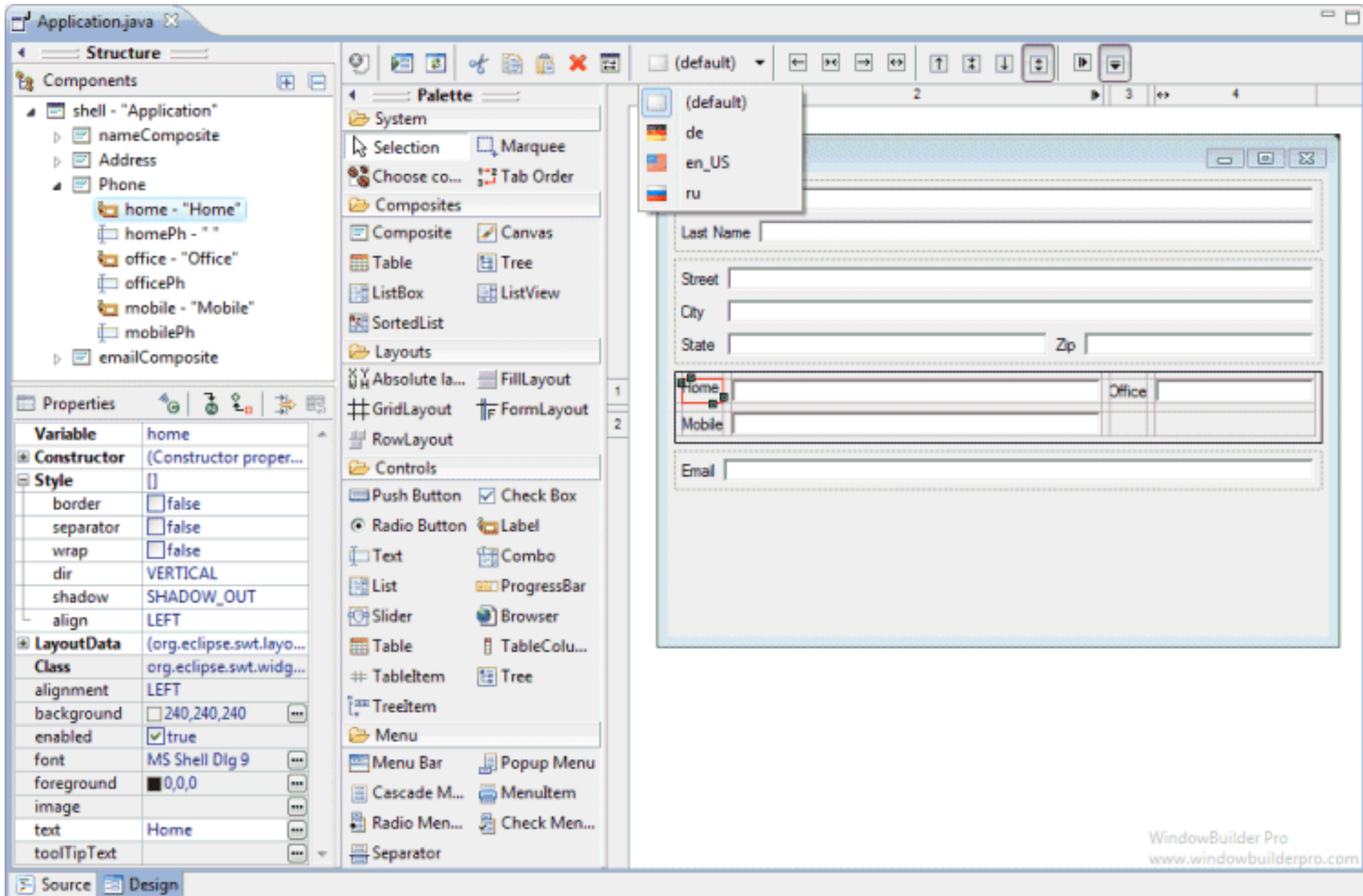
```
$ javac view/InterfaceGrafica.java
$ java view/InterfaceGrafica
```

The terminal window also features a vertical scrollbar on the right side.

# resultado



<https://www.eclipse.org/windowbuilder/>



# Ferramentas GUI

- Ajudam, porém existem casos onde é necessário codificar diretamente. Exemplos:
  - entender como os componentes são alocados na tela
  - como são feitos os processos de posicionamento
  - controle de eventos
  - solução de eventuais problemas
  - recursos adicionais
  - componentes criados em tempo de execução
  - ajustes finos e otimização
- Além disso é preciso aprender a usar a ferramenta, o que pode demandar tempo adicional

- package view;
  - mantém classes relacionadas no mesmo pacote
  - evita conflitos de nomes
  - ajuda a manutenibilidade de código
- coesão
  - delimita classes relacionadas a um propósito
  - manutenibilidade
  - reuso
- acoplamento
  - dependência entre módulos



- `import javax.swing.JFrame;`
  - pacotes a incluir no código (neste caso, Swing)
- `public class InterfaceGrafica extends JFrame {`
  - classe para representar a tela
  - `extends` a classe `JFRAME`
- `public InterfaceGrafica() {`
  - construtor da classe `InterfaceGrafica`

- setSize()
  - tamanho da tela
- setTitle()
  - título da tela
- setVisible()
  - define se a tela é visível
- ```
public static void main(String[] args) {  
    InterfaceGrafica telaInicial = new InterfaceGrafica();
```

  - a execução do programa começa em main
  - new cria o objeto

# 5 passos para criar uma interface gráfica

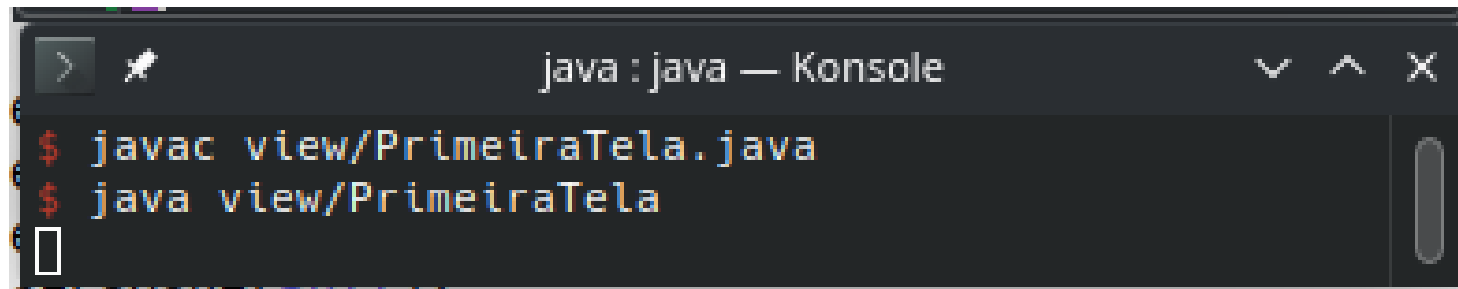
- Criar uma relação de especialização com a classe que representa a tela
- Declarar como atributos os elementos que serão adicionados à tela
- Definir a forma de alocação dos elementos gráficos na tela
- No construtor, instanciar, configurar e posicionar os itens na tela
- Tratar os eventos dos componentes para tratar as ações do usuário com a interface gráfica

```
1 package view;
2 import javax.swing.JFrame;
3 import javax.swing.JLabel;
4 import javax.swing.JTextField;
5
6 public class PrimeiraTela extends JFrame {
7     private JLabel lblNome;
8     private JTextField txtNome;
9
10    public PrimeiraTela() {
11        lblNome = new JLabel("Nome");
12        txtNome = new JTextField();
13
14        setSize(400, 200);
15        setTitle("Tela Inicial");
16        setVisible(true);
17        setLayout(null);
18
19        lblNome.setBounds(10, 10, 100, 25);
20        txtNome.setBounds(50, 10, 200, 25);
21        getContentPane().add(lblNome);
22        getContentPane().add(txtNome);
23    }
24    public static void main(String[] args) {
25        PrimeiraTela t1 = new PrimeiraTela();
26    }
```

texto sem  
formatação

- sem gerenciador de layout  
- usa coordenadas x, y

largura,  
altura



```
java : java — Konsole
$ javac view/PrimeiraTela.java
$ java view/PrimeiraTela

```

A terminal window titled "java : java — Konsole" with standard window controls. It contains two lines of text: a prompt followed by "javac view/PrimeiraTela.java" and another prompt followed by "java view/PrimeiraTela". A cursor is visible on the line following the second command.



| Componente       | Descrição                                                                                                                                                                                        |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>JButton</b>   | Objeto usado para criar botões.                                                                                                                                                                  |
| <b>JCheckBox</b> | Objeto usado para oferecer uma opção para o usuário. Normalmente é representado por uma caixa de seleção, que quando está com "check" representa "sim" e quando está com "não" representa "não". |
| <b>JComboBox</b> | Objeto usado para oferecer mais de uma opção para o usuário em forma de lista <i>drop-down</i> . Para cada lista somente um item pode ser selecionado.                                           |
| <b>JList</b>     | Objeto usado para oferecer mais de uma opção para o usuário, mas, diferentemente do JComboBox, esse componente permite a seleção de mais que uma opção.                                          |
| <b>JPanel</b>    | Objeto usado para organizar diversos componentes.                                                                                                                                                |

- `JFormattedTextField`: entrada com formatação
- Os componentes na interface podem ser posicionados de forma estática ou dinâmica
- construtor: quando um objeto é criado, o construtor é executado

```
java : java — Konsole
$ javac PrimeiraTela2.java
Note: PrimeiraTela2.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
$ java PrimeiraTela2

```

The image shows a Java Swing window titled "Tela Inicial". It contains a login form with the following components:

- A text field labeled "Nome" (Name).
- A text field labeled "CPF" (CPF) with a mask showing three dots and a dash.
- A dropdown menu labeled "Tipo de usuário" (User type) with "Administrador" (Administrator) selected.
- A blue button labeled "Enviar" (Send).





```
32     }
33     lblTipo = new JLabel("Tipo de usuário");
34     cmbTipo = new JComboBox(tiposUsuarios);
35     btnOK = new JButton("Enviar");
36     ctn.setLayout(null);
37     lblNome.setBounds(0, 0, 100, 25);
38     txtNome.setBounds(150, 0, 200, 25);
39     lblCPF.setBounds(0, 50, 100, 25);
40     txtCPF.setBounds(150, 50, 200, 25);
41     lblTipo.setBounds(0, 100, 200, 25);
42     cmbTipo.setBounds(150, 100, 200, 25);
43     btnOK.setBounds(150, 150, 100, 100);
44     ctn.add(lblNome);
45     ctn.add(txtNome);
46     ctn.add(lblCPF);
47     ctn.add(txtCPF);
48     ctn.add(lblTipo);
49     ctn.add(cmbTipo);
50     ctn.add(btnOK);
51     setVisible(true);
52     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
53 }
54
55 public static void main(String[] args) {
56     PrimeiraTela2 t1 = new PrimeiraTela2();
57 }
58
```

botão

posição absoluta

insere no container do jframe

cria a instância

# Exemplo

- [https://www.youtube.com/watch?v=yL\\_G0uCVSmI](https://www.youtube.com/watch?v=yL_G0uCVSmI)
  - usa o Netbeans

Java Swing componentes básicos

Juan Jesús Martínez

2015

(sem som)

# Exemplo Andrijauskas Fig. 1.5, p. 24.

```
import java.awt.*;
import java.text.ParseException;
import javax.swing.*;
import javax.swing.text.*;

public class InterfaceGraficaCampos extends JFrame {
    private JLabel lblNome;
    private JTextField txtNome;

    private JLabel lblEndereco;
    private JTextField txtEndereco;

    private JLabel lblTelefone;
    private JFormattedTextField txtTelefone;

    private JLabel lblCPF;
    private JFormattedTextField txtCPF;

    private final String[] tipoSanguineo = {
        "A", "B", "AB", "O"
    };

    private JLabel lblTipoSanguineo;
    private JComboBox cmbTipoSanguineo;
```

Fonte: <https://cm-cls-content.s3.amazonaws.com/ebook/embed/qrcode/2018-2/programacao-orientada-a-objetos-II/u1/s1/codigo.pdf>

```
private final String[] fatorRH = {  
    "+", "-"  
};  
private JLabel lblRH;  
private JComboBox cmbRH;  
  
private final String[] cursos = {  
    "Ciência da Computação",  
    "Análise e desenvolvimento de sistemas"  
};  
private JLabel lblCurso;  
private JComboBox cmbCurso;  
  
private JLabel lblNomeEmergencia;  
private JTextField txtNomeEmergencia;  
  
private JLabel lblTelefoneEmergencia;  
private JFormattedTextField txtTelefoneEmergencia;  
  
private JButton btnInserir;  
private JButton btnCancelar;  
  
private Container ctn;
```

```
public InterfaceGraficaCampos() {  
    setSize(400, 500);  
    setTitle("Sistema de cadastro");  
    ctn = getContentPane();  
  
    lblNome = new JLabel("Nome");  
    txtNome = new JTextField();  
  
    lblEndereco = new JLabel("Endereço");  
    txtEndereco = new JTextField();  
  
    lblTelefone = new JLabel("Telefone");  
    try {  
        txtTelefone = new JFormattedTextField(new  
MaskFormatter("(##)#####"));  
    } catch (ParseException e) {  
        e.printStackTrace();  
    }  
}
```

```
lblTipoSanguineo = new JLabel("Tipo de sanguineo");
cmbTipoSanguineo = new JComboBox(tipoSanguineo);

lblRH = new JLabel("Fator RH");
cmbRH = new JComboBox(fatorRH);

lblCurso = new JLabel("Curso");
cmbCurso = new JComboBox(cursos);

lblNomeEmergencia = new JLabel("Contato de emergência");
txtNomeEmergencia = new JTextField();

lblTelefoneEmergencia = new JLabel("Telefone");
try {
    txtTelefoneEmergencia = new
        JFormattedTextField(new MaskFormatter("(##)#####"));
} catch (ParseException e) {
    e.printStackTrace();
}

btnInserir = new JButton("Inserir");
btnCancelar = new JButton("Cancelar");

ctn.setLayout(null);
```

```
lblNome.setBounds(0, 0, 100, 25);  
txtNome.setBounds(150, 0, 200, 25);  
  
lblEndereco.setBounds(0, 50, 100, 25);  
txtEndereco.setBounds(150, 50, 200, 25);  
  
lblTelefone.setBounds(0, 100, 100, 25);  
txtTelefone.setBounds(150, 100, 200, 25);  
  
lblCPF.setBounds(0, 150, 100, 25);  
txtCPF.setBounds(150, 150, 200, 25);  
  
lblTipoSanguineo.setBounds(0, 200, 200, 25);  
cmbTipoSanguineo.setBounds(150, 200, 50, 25);  
lblRH.setBounds(220, 200, 50, 25);  
cmbRH.setBounds(280, 200, 50, 25);  
  
lblCurso.setBounds(0, 250, 200, 25);  
cmbCurso.setBounds(150, 250, 200, 25);  
  
lblNomeEmergencia.setBounds(0, 300, 200, 25);  
txtNomeEmergencia.setBounds(150, 300, 200, 25);  
  
lblTelefoneEmergencia.setBounds(0, 350, 200, 25);  
txtTelefoneEmergencia.setBounds(150, 350, 200, 25);  
  
btnInserir.setBounds(100, 400, 100, 50);  
btnCancelar.setBounds(200, 400, 100, 50);
```





```
ctn.add(lblNome);
ctn.add(txtNome);
ctn.add(lblEndereco);
ctn.add(txtEndereco);
ctn.add(lblTelefone);
ctn.add(txtTelefone);
ctn.add(lblCPF);
ctn.add(txtCPF);
ctn.add(lblTipoSanguineo);
ctn.add(cmbTipoSanguineo);
ctn.add(lblRH);
ctn.add(cmbRH);
ctn.add(lblCurso);
ctn.add(cmbCurso);
ctn.add(lblNomeEmergencia);
ctn.add(txtNomeEmergencia);
ctn.add(lblTelefoneEmergencia);
ctn.add(txtTelefoneEmergencia);
ctn.add(btnInserir);
ctn.add(btnCancelar);

setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); }

public static void main(String[] args) {
    new InterfaceGraficaCampos(); } }
```

---



  Sistema de cadastro


Nome

Endereço

Telefone

CPF

Tipo de sanguineo   Fator ...  

Curso  

Contato de emergência

Telefone

# Saída

```
> aula01_11 :...a — Konsole
$ javac InterfaceGraficaCampos.java
Note: InterfaceGraficaCampos.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
$ java InterfaceGraficaCampos

```

# Formatação

- CEP: 12912-010
  - `JFormattedTextFielddtxtCEP = new  
JFormattedTextField(new MaskFormatter("#####-##"));`
- CNPJ: 50.364.671/0001-00
  - `JFormattedTextFielddtxtCNPJ = new  
JFormattedTextField(new  
MaskFormatter("##.###.####/####-##"));`
- Código de fornecedor
  - `JFormattedTextFielddtxtCodFornecedor = new  
JFormattedTextField(new MaskFormatter("UU-#####L"));`

# = número, U = caixa alta, L = caixa baixa