



**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
RIO GRANDE DO SUL
Câmpus Feliz

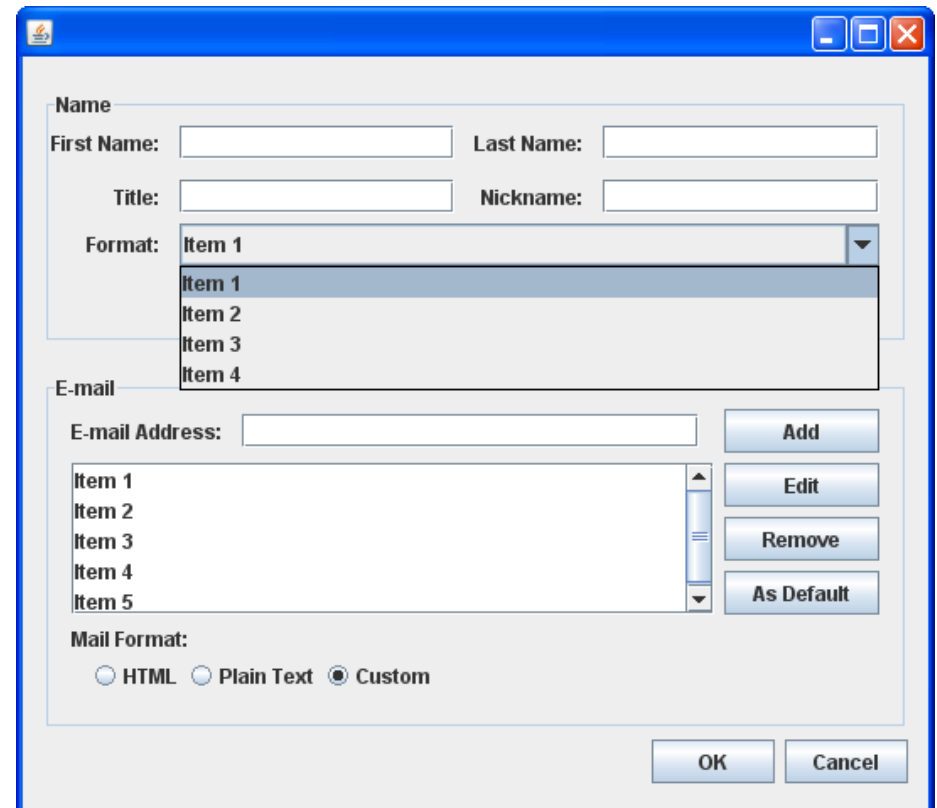
GUI

Prof. Moser Fagundes

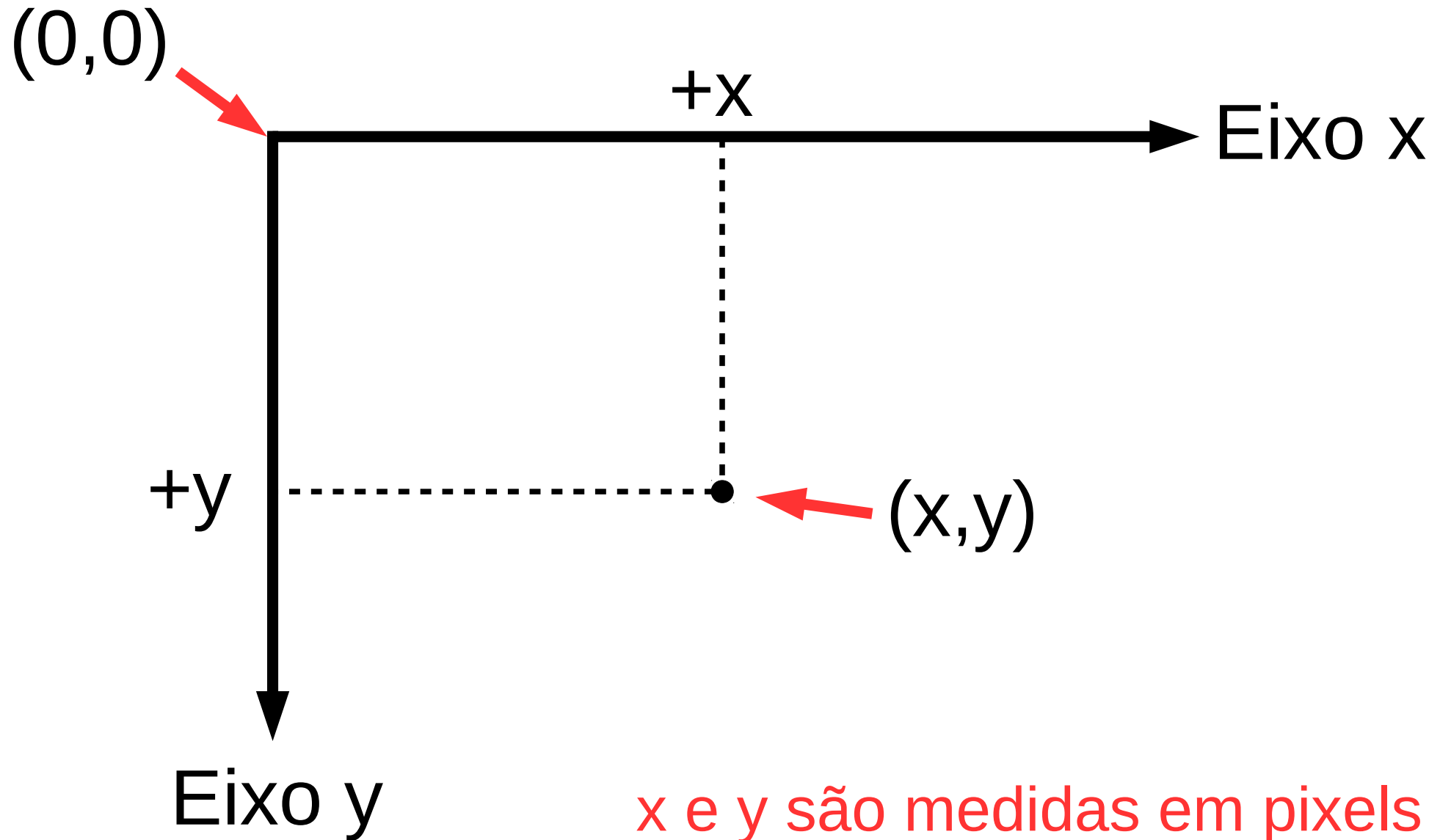
Programação II

GUI

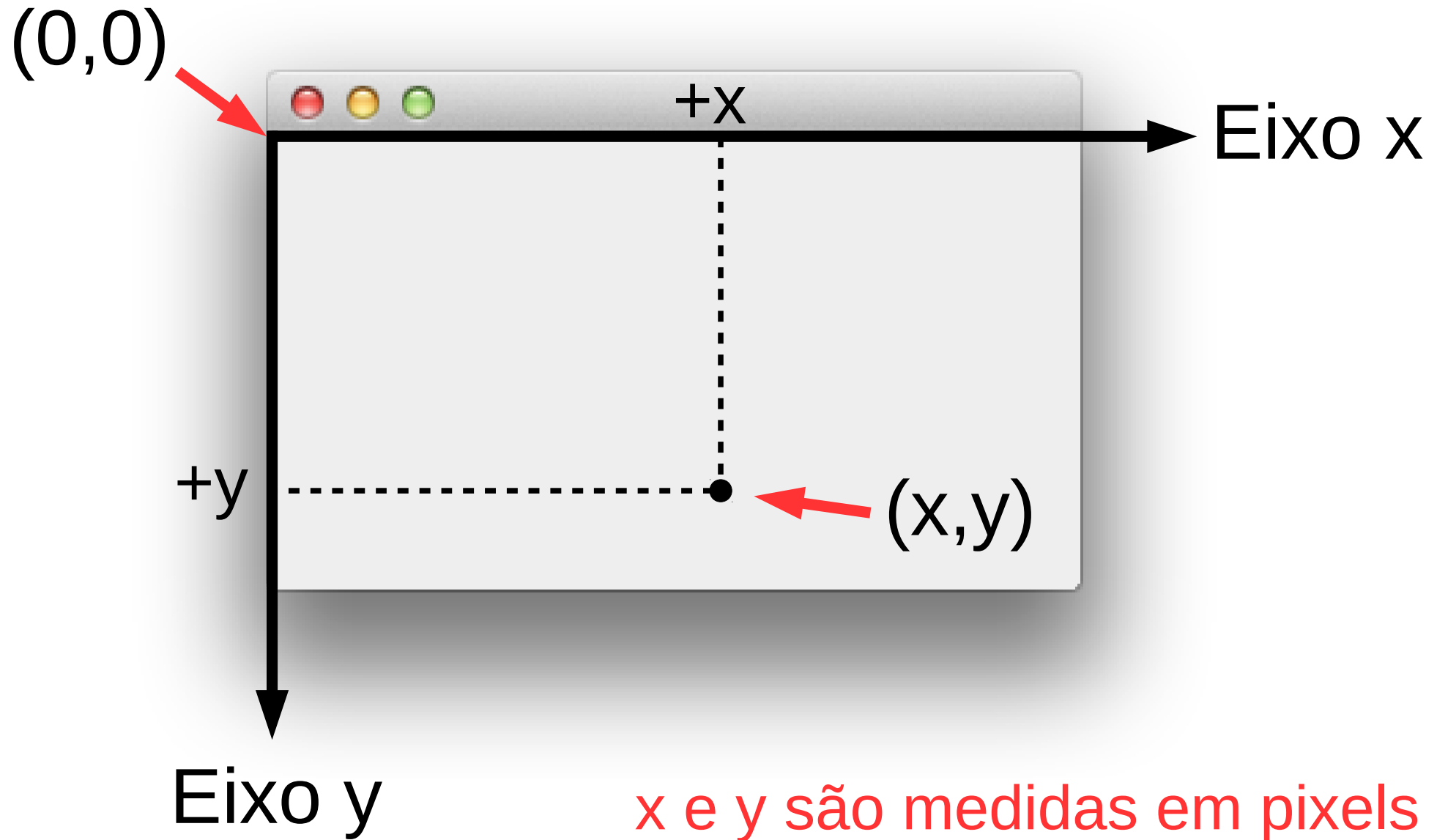
- **GUI** = **G**raphical **U**ser **I**nterface
- Ou *Interface Gráfica do Usuário* em português.
- **Exemplos:**



Sistema de coordenadas do Java



Sistema de coordenadas do Java



Elementos básicos da GUI

- Para criar uma GUI, precisamos de:
 - um **frame** (quadro, contorno) – classe JFrame
 - um **panel** (painel, interior do frame) – classe JPanel

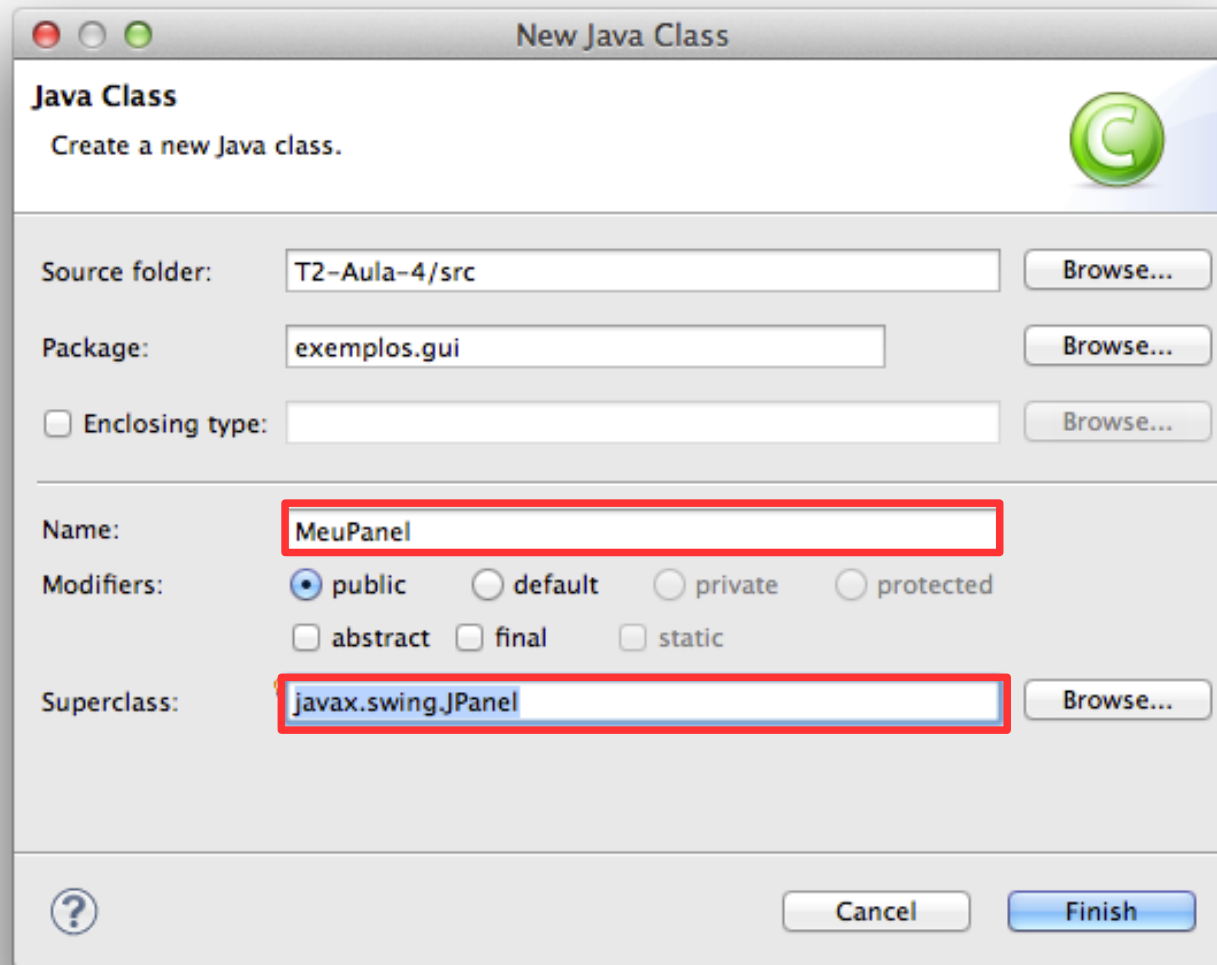
JFrame



JPanel

Exemplo

1. Criar o nossa extensão da classe JPanel.



Exemplo

Código da classe MeuPanel.

```
package exemplos.gui;  
  
import javax.swing.JPanel;  
  
public class MeuPanel extends JPanel {  
  
    // Meu código para desenhar algo será  
    // escrito nos métodos desta classe!  
  
}
```

Exemplo

2. Importar a classe Graphics.
3. Criar o método que faz o desenho.
4. Adicionar a linha **super**.paintComponent(g);

```
package exemplos.gui;
```

```
import java.awt.Graphics;  
import javax.swing.JPanel;
```

```
public class MeuPanel extends JPanel {
```

```
    public void paintComponent(Graphics g) {  
        super.paintComponent(g);
```

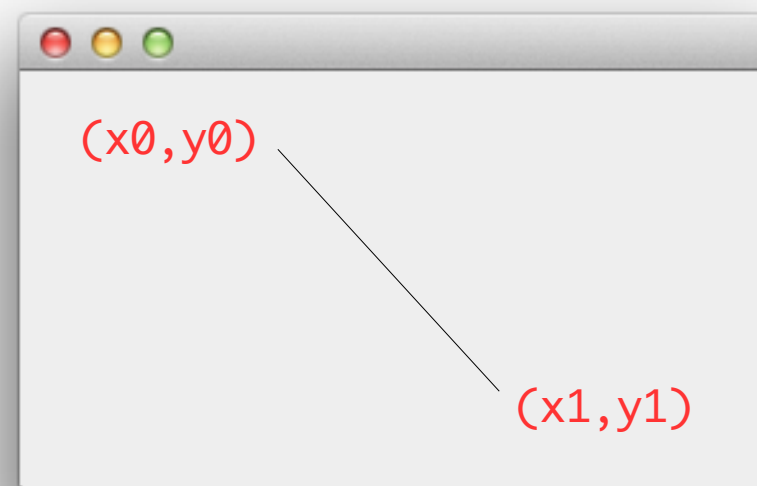
```
        // Aqui colocamos o código do desenho
```

```
    }  
}
```


Método para desenhar linhas

Para desenhar uma linha devemos saber:

- o ponto inicial (x_0, y_0)
- o ponto final (x_1, y_1)



Então, usamos o método:

```
g.drawLine(x0,y0,x1,y1);
```

Exemplo

5. Desenhar 2 linhas.

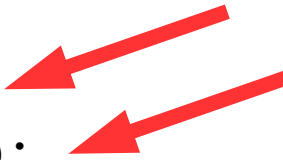
```
package exemplos.gui;

import java.awt.Graphics;
import javax.swing.JPanel;

public class MeuPanel extends JPanel {

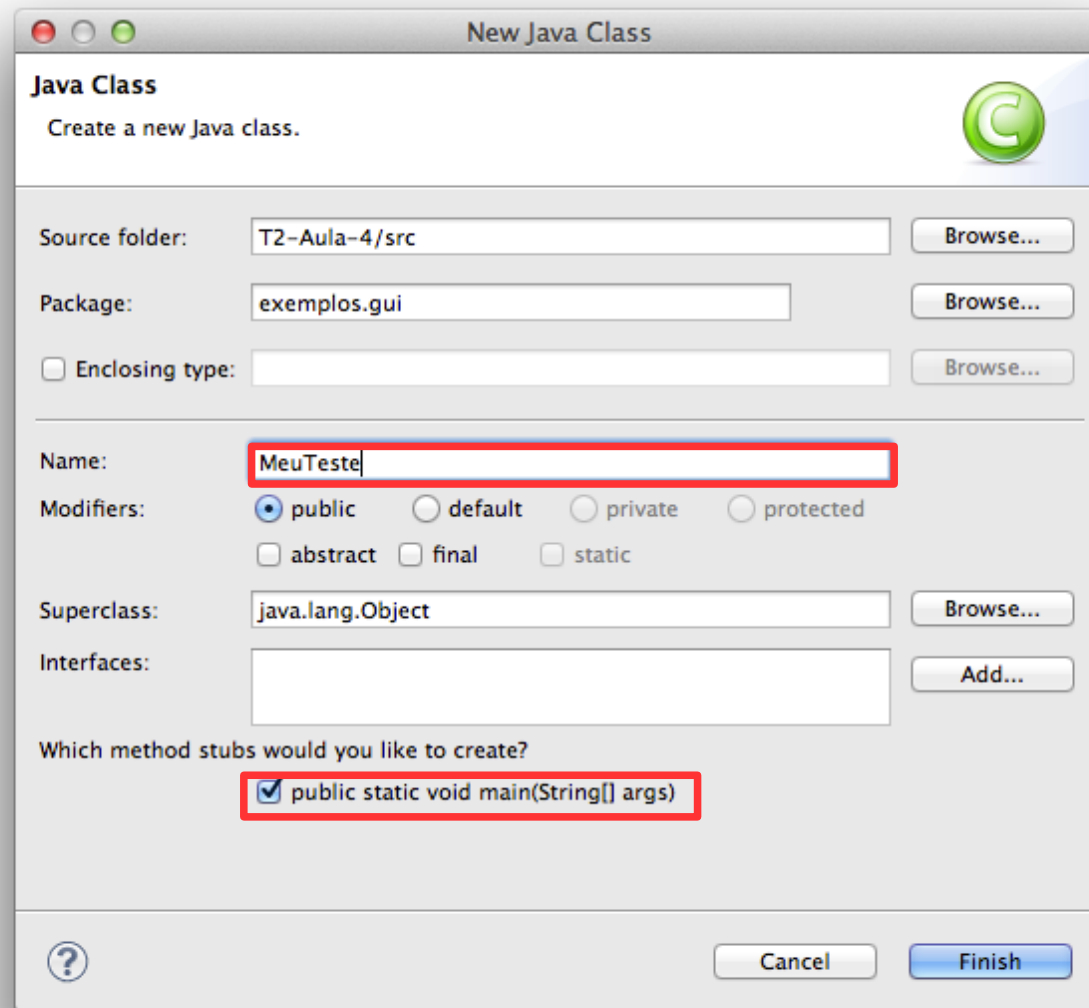
    public void paintComponent(Graphics g) {
        super.paintComponent(g);

        g.drawLine(0, 0, 300, 300);
        g.drawLine(100, 30, 400, 30);
    }
}
```



Exemplo

6. Criar uma classe de teste contendo o main.



Exemplo

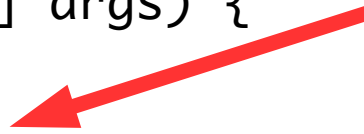
7. Criar um objeto da classe JFrame.

```
package exemplos.gui;
```

```
public class MeuTeste {
```

```
    public static void main(String[] args) {
```

```
        MeuPanel p = new MeuPanel();
```



```
    }  
}
```

Exemplo

7. Criar um objeto da classe JFrame.

8. Configurar o frame.

```
package exemplos.gui;
```

```
public class MeuTeste {
```

```
    public static void main(String[] args) {
```

```
        MeuPanel p = new MeuPanel();
```

```
        JFrame frame = new JFrame();
```

```
        frame.add(p);
```

```
        frame.setSize(500,500);
```

```
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
        frame.setVisible(true);
```

```
    }
```

```
}
```




Exemplo

Configuração do frame passo-a-passo.

...

```
JFrame frame = new JFrame();  
frame.add(p);  
frame.setSize(500,500);  
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
frame.setVisible(true);
```

Cria o frame
vazio, sem
conteúdo.




Exemplo

Configuração do frame passo-a-passo.

...

```
JFrame frame = new JFrame();  
frame.add(p);  
frame.setSize(500,500);  
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
frame.setVisible(true);
```

Adiciona o
painel dentro
do frame.



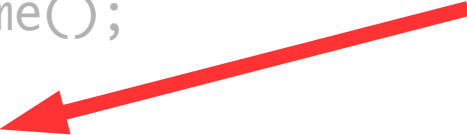
Exemplo

Configuração do frame passo-a-passo.

...

```
JFrame frame = new JFrame();  
frame.add(p);  
frame.setSize(500,500);  
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
frame.setVisible(true);
```

Determina o
tamanho do
frame em pixels.



Exemplo

Configuração do frame passo-a-passo.

...

```
JFrame frame = new JFrame();  
frame.add(p);  
frame.setSize(500,500);  
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
frame.setVisible(true);
```



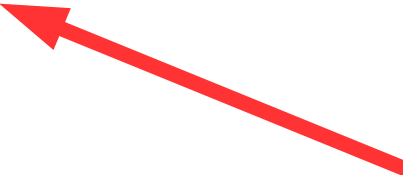
Determina que se fecharmos a janela, o programa se encerra.

Exemplo

Configuração do frame passo-a-passo.

...

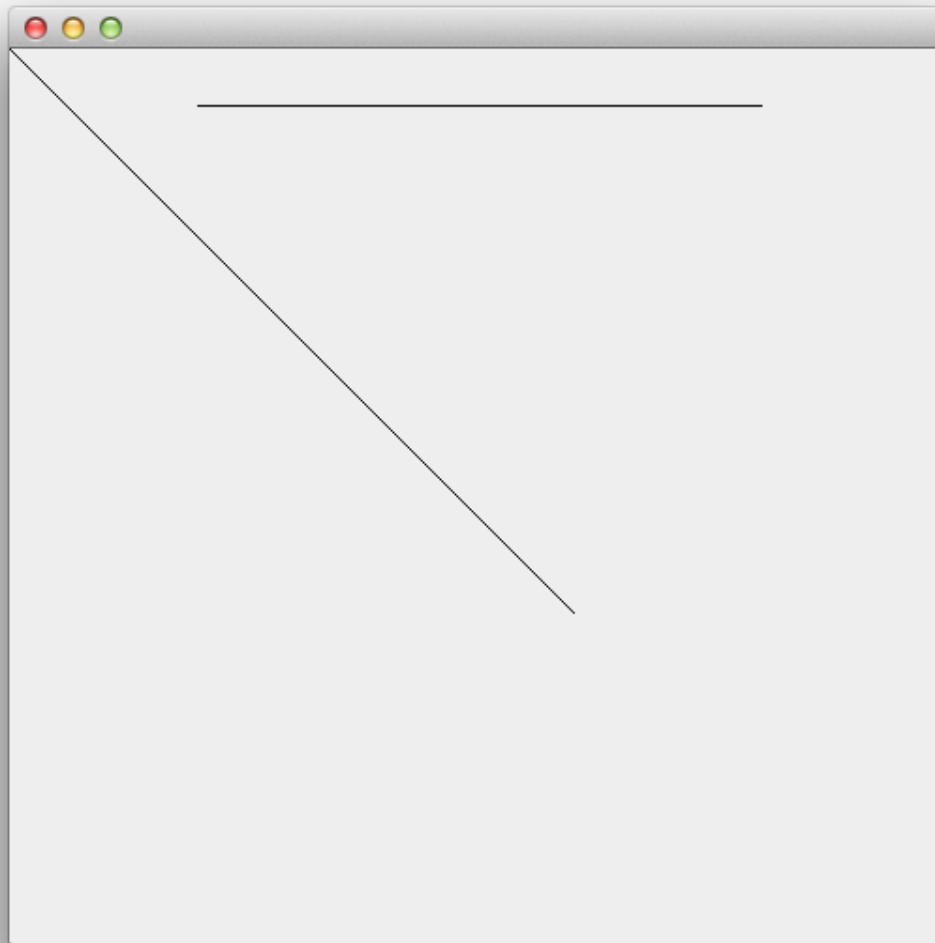
```
JFrame frame = new JFrame();  
frame.add(p);  
frame.setSize(500,500);  
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
frame.setVisible(true);
```



Faz com que a janela se torne visível ao usuário.

Exemplo

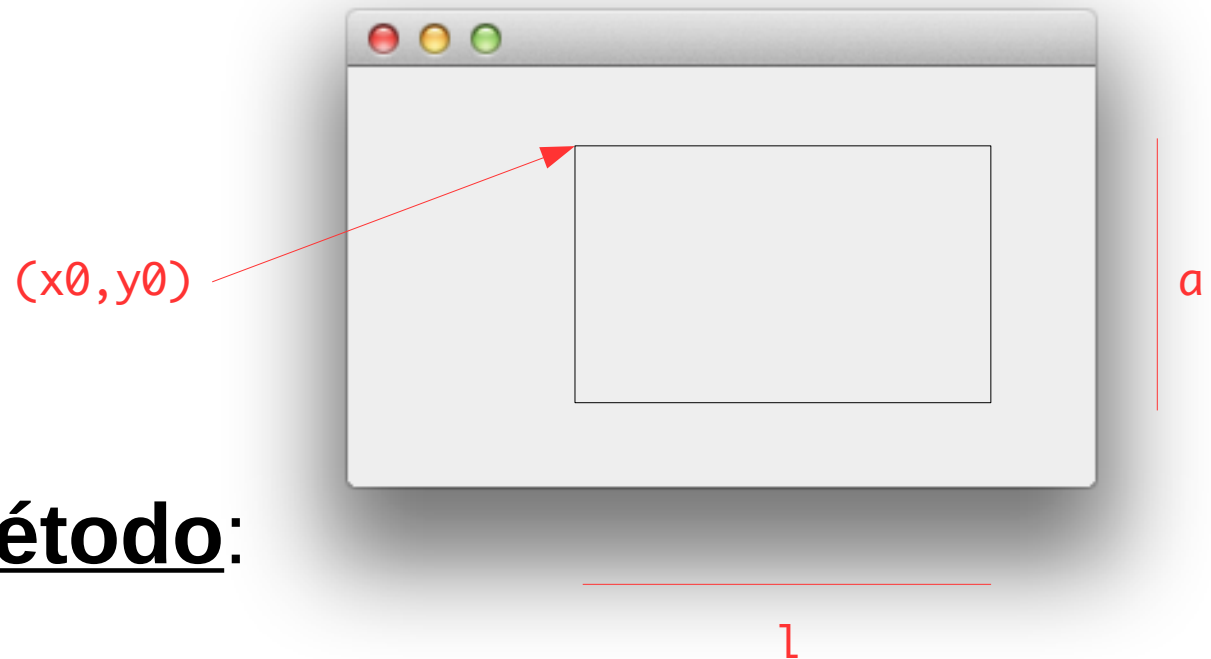
Resultado:



Método para desenhar retângulos

Para desenhar um retângulo devemos saber:

- o ponto esquerdo superior $(x0, y0)$
- largura l
- altura a



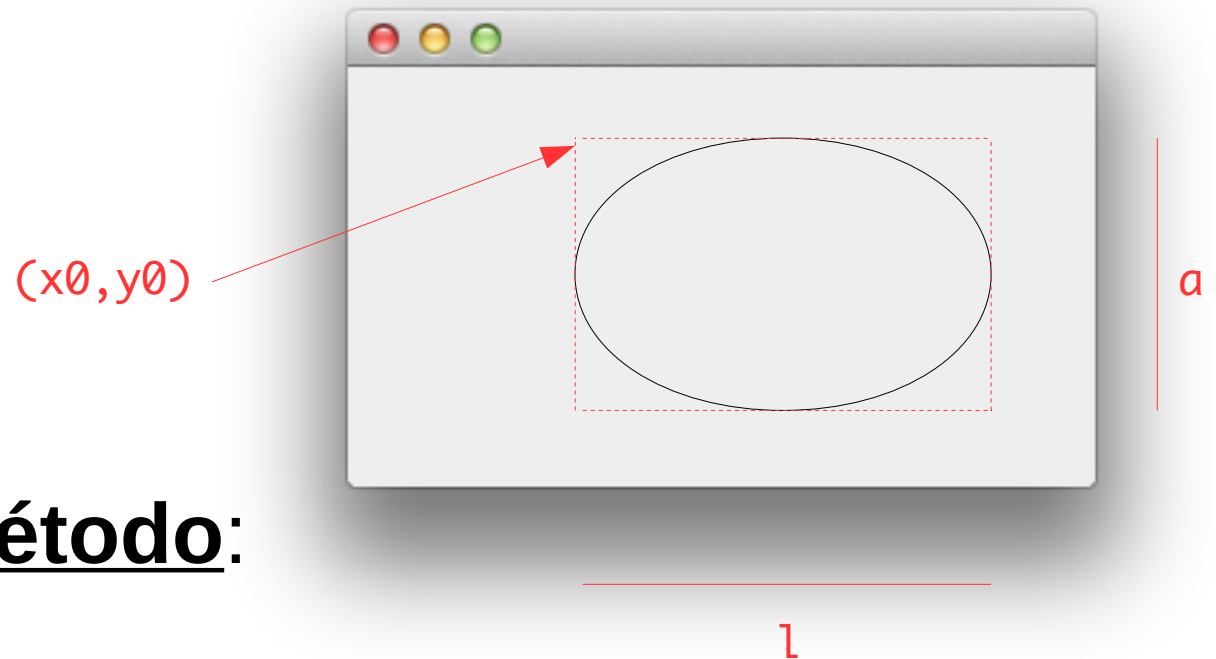
Então, usamos o método:

```
g.drawRect(x0, y0, l, a);
```

Método para desenhar elipses

Para desenhar uma elipse devemos saber:

- o ponto esquerdo superior (x_0, y_0)
- largura l
- altura a



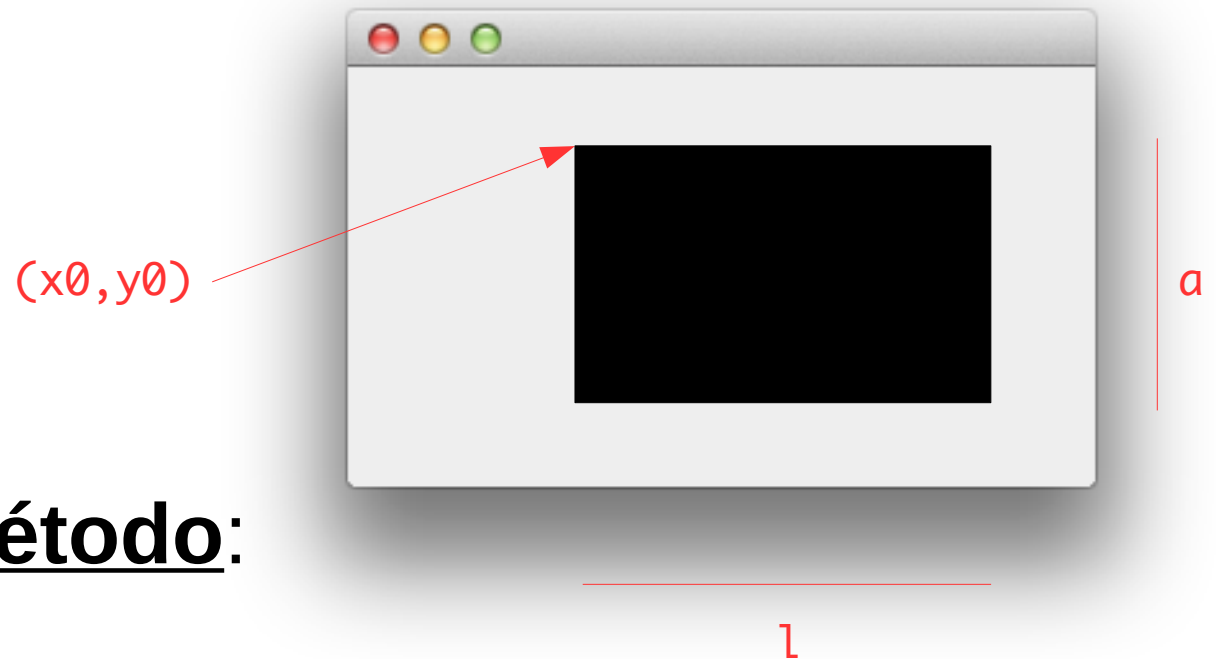
Então, usamos o método:

```
g.drawOval(x0, y0, l, a);
```

Retângulos preenchidos

Para desenhar um retângulo devemos saber:

- o ponto esquerdo superior (x_0, y_0)
- largura l
- altura a



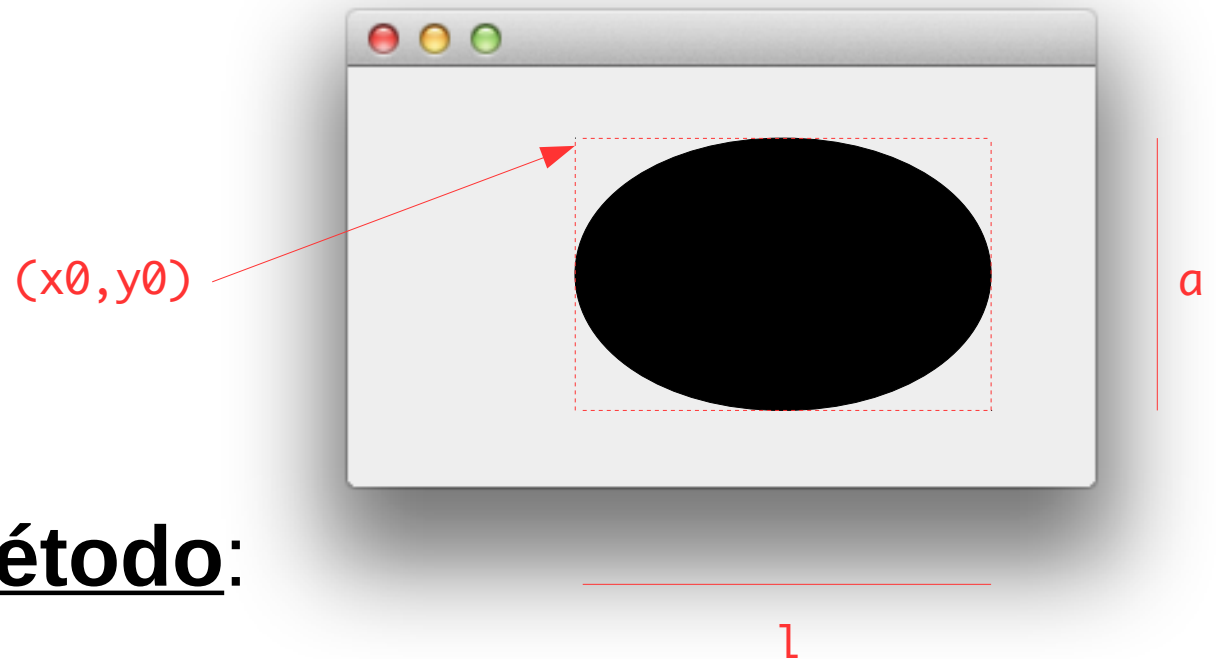
Então, usamos o método:

```
g.fillRect(x0,y0,l,a);
```

Elipses preenchidas

Para desenhar uma elipse devemos saber:

- o ponto esquerdo superior (x_0, y_0)
- largura l
- altura a



Então, usamos o método:

```
g.fillOval(x0,y0,l,a);
```

Exemplo

Desenhar 2 formas preenchidas.

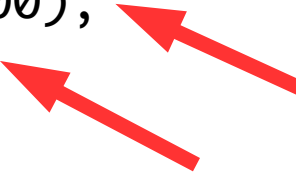
```
package exemplos.gui;

import java.awt.Graphics;
import javax.swing.JPanel;

public class MeuPanel extends JPanel {

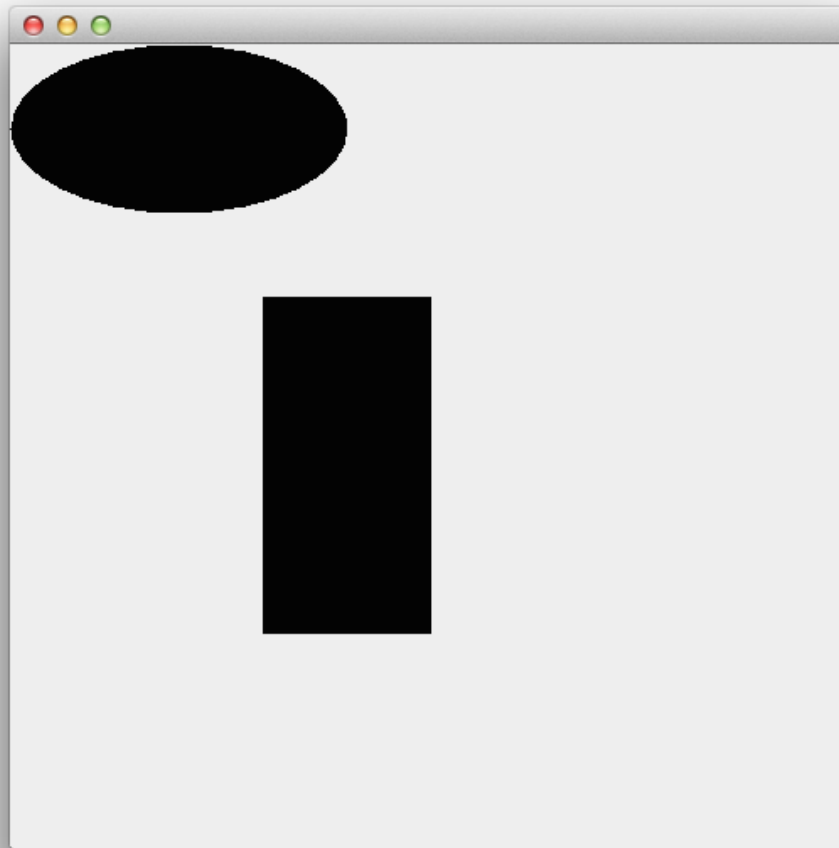
    public void paintComponent(Graphics g) {
        super.paintComponent(g);

        g.fillRect(150, 150, 100, 200);
        g.fillOval(0, 0, 200, 100);
    }
}
```

Two red arrows are drawn on the right side of the code block. One arrow points from the right towards the parameters of the `g.fillRect` method call. The other arrow points from the right towards the parameters of the `g.fillOval` method call.

Exemplo

Resultado:



Adicionando cor

Mudamos a cor e depois desenhamos.

```
package exemplos.gui;
```

```
import java.awt.Graphics;
```

```
import javax.swing.JPanel;
```

```
public class MeuPanel extends JPanel {
```

```
    public void paintComponent(Graphics g) {  
        super.paintComponent(g);
```

```
        g.setColor(Color.BLUE);
```

```
        g.fillRect(150, 150, 100, 200);
```

```
        g.setColor(Color.RED);
```

```
        g.fillRect(0, 0, 200, 100);
```

```
    }
```

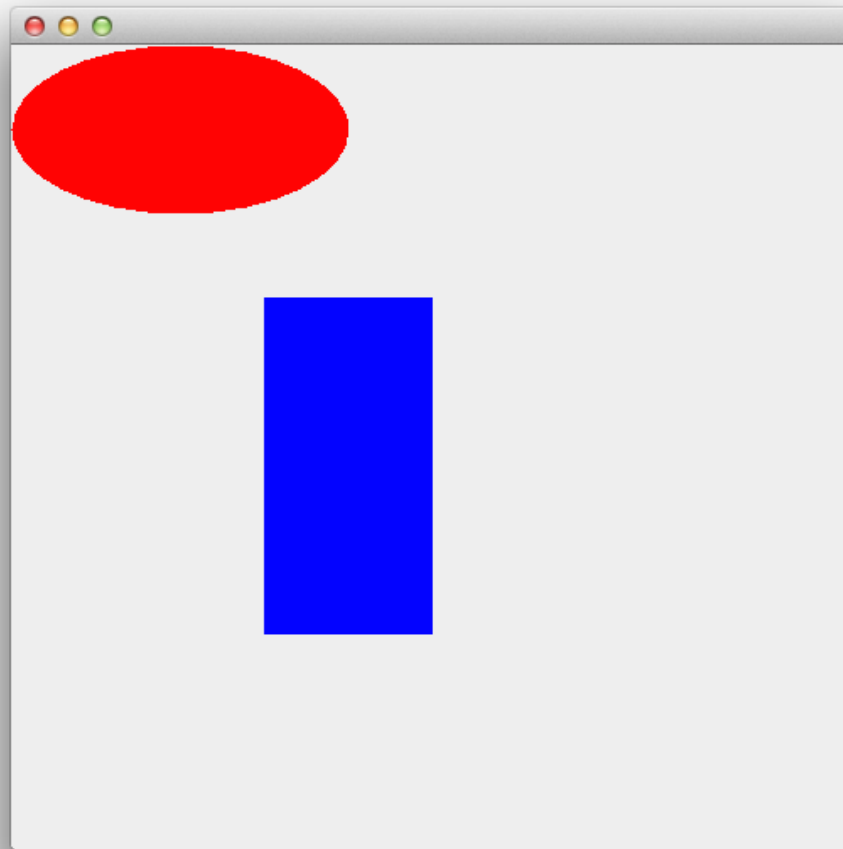
```
}
```

Troca a cor
para azul

Troca a cor
para vermelho

Exemplo

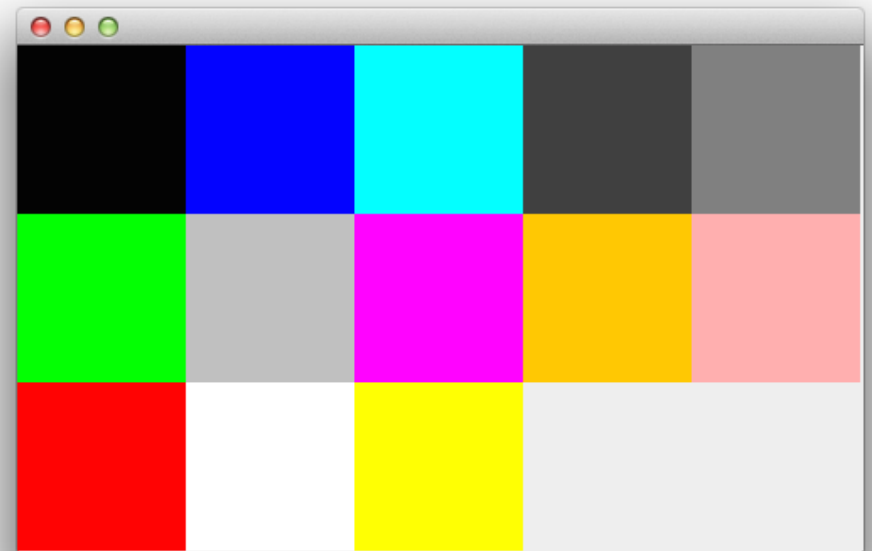
Resultado:



Cores

As seguintes cores estão disponíveis como constantes da classe `Color`:

```
g.setColor(Color.BLACK);  
g.setColor(Color.BLUE);  
g.setColor(Color.CYAN);  
g.setColor(Color.DARK_GRAY);  
g.setColor(Color.GRAY);  
g.setColor(Color.GREEN);  
g.setColor(Color.LIGHT_GRAY);  
g.setColor(Color.MAGENTA);  
g.setColor(Color.ORANGE);  
g.setColor(Color.PINK);  
g.setColor(Color.RED);  
g.setColor(Color.WHITE);  
g.setColor(Color.YELLOW);
```



Cores Personalizadas

Para criar uma cor personalizada, devemos criar um objeto da classe **Color**. Quando criamos este objeto, passamos 3 números que correspondem a números do sistema de cores RGB (**Red**, **Green**, **Blue**).

http://www.w3schools.com/colors/colors_picker.asp

No exemplo abaixo, criamos duas cores personalizadas.

```
Color minhaCor1 = new Color(30,50,200);  
g.setColor(minhaCor1);  
g.fillRect(0, 0, 100, 100);
```

```
Color minhaCor2 = new Color(200,50,50);  
g.setColor(minhaCor2);  
g.fillRect(100, 0, 100, 100);
```

Escrevendo texto

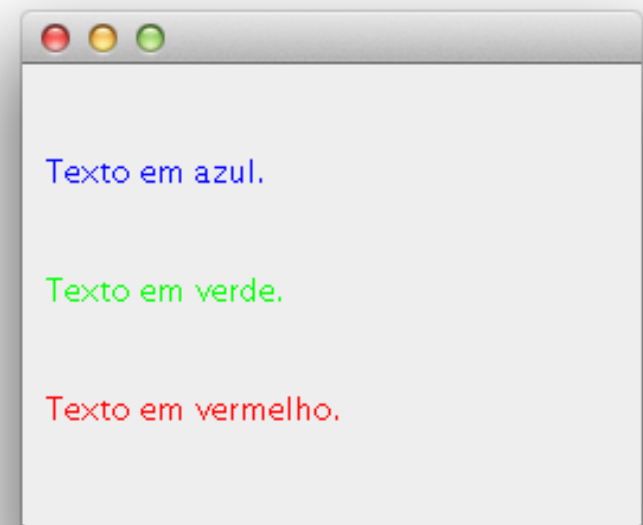
Para escrever um texto na tela, usamos o método `drawString` da classe `Graphics`. Para tal, devemos informar a `String` a ser escrita, a sua posição (x,y).

No exemplo abaixo, escrevemos três frases em diferentes cores.

```
g.setColor(Color.BLUE);  
g.drawString("Texto em azul.",10,50);  
  
g.setColor(Color.GREEN);  
g.drawString("Texto em verde.",10,100);  
  
g.setColor(Color.RED);  
g.drawString("Texto em vermelho.",10,150);
```

↑
Texto

↑ ↑
X Y



Fontes

Para modificar a fonte de um texto, criamos um objeto da classe Font que conterá as seguintes configurações:

- Nome da fonte – As fontes default do Java são **Monospaced**, **Serif** e **SansSerif**.
- Estilo da fonte
 - Plano: **Font.PLAIN**
 - Itálico: **Font.ITALIC**
 - Negrito: **Font.BOLD**
- Tamanho da fonte (valor numérico)

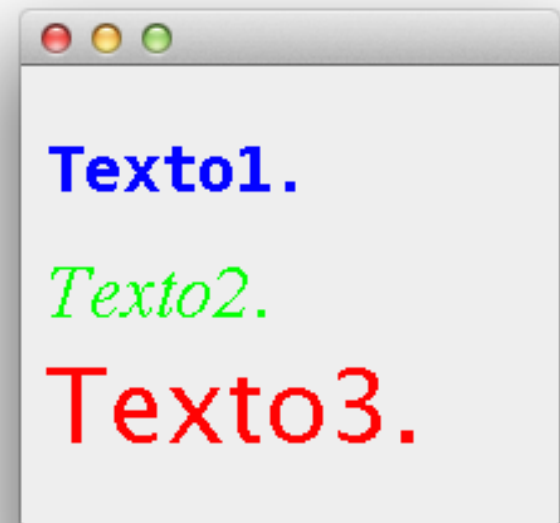
Exemplo com fontes

No exemplo abaixo, escrevemos diversas frases em diferentes cores, com fontes de diferentes estilos e tamanhos.

```
Font f1 = new Font("Monospaced",Font.BOLD,25);  
g.setFont(f1);  
g.setColor(Color.BLUE);  
g.drawString("Texto1.", 10, 50);
```

```
Font f2 = new Font("Serif",Font.ITALIC,30);  
g.setFont(f2);  
g.setColor(Color.GREEN);  
g.drawString("Texto2.", 10, 100);
```

```
Font f3 = new Font("SansSerif",Font.PLAIN,40);  
g.setFont(f3);  
g.setColor(Color.RED);  
g.drawString("Texto3.", 10, 150);
```



Exercício

Ver lista de exercícios no Moodle.