**JFrame e JPanel: como criar uma aplicação gráfica em Java**

No artigo passado, de [Introdução ao estudo de GUI](http://www.javaprogressivo.net/2012/11/JFrame-e-JPanel-Introducao-ao-estudo-de-GUI-em-Java.html), demos uma explicação teórica e bem simples de entender sobre JFrame e JPanel. Agora vamos, de fato, usar JFrame e JPanel.  
  
A partir deste ponto de nossa **apostila de Java**, iremos criar nossa própria 'janela', ou seja, não vamos usar as janelas já feitas, como nas caixas de diálogo.

* [Apostila de Java em PDF](https://www.javaprogressivo.net/p/apostila-java-download-gratis-baixar.html)

**JFrame: Como criar Janelas em Java**

JFrame nada mais é que uma classe, como outra qualquer que estudamos e criamos ao longo de nosso curso.  
Porém, essa classe é que será responsável por criar a tela em que iremos desenhar, colocar botões, menus, caixas de texto e tudo mais que existem nas janelas de aplicativos.  
  
Para fazer uso do JFrame, temos que importar essa classe do pacote swing, que contém diversas funcionalidades para programação gráfica:  
import javax.swing.JFrame;  
  
Cada objeto que criamos é um frame, ou janela, diferente. Uma aplicação normal tem dezenas ou centenas de frames.  
  
Nessa seção vamos criar alguns objetos da classe JFrame passando uma String argumento para o construtor padrão, essa string será o título de nossa aplicação.  
  
Esses JFrames, porém, possuem dezenas de funcionalidades e opções, usaremos as funcionalidades: o que ocorre quando clicamos no 'x' de close, adicionar panels, definir o tamanho do frame e se ele será visível ou não.  
  
Como dissemos, o JFrame é uma Classe. E para criar uma classe, criamos um objeto.  
Assim, nossa janela será um objeto chamado "janela".  
Vamos passar uma string para o construtor dessa classe, que servirá como título de nosso *frame*/janela.  
  
Porém, para vermos nosso frame, precisamos definir uma funcionalidade: que ele seja visível!  
Pode parecer óbvio, mas um programa pode ter muitos, mas muitos frames, e obviamente não podemos exibir todos, senão nossa tela ficaria uma bagunça.  
  
Assim, para exibir esse frame, usamos o método ***setVisible***, que recebe *true* ou *false*.  
Portanto, um simples programa em Java que cria e exibe um frame será:

**import** javax.swing.JFrame;

**public** **class** framesPanels {

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

JFrame janela = **new** JFrame("Meu primeiro frame em Java");

janela.setVisible(**true**);

}}

Note que vai ser criado uma janela, mas provavelmente você nem vai notar. Isso porque não fornecemos o tamanho da janela, então ela foi criado com 0 pixel de largura e 0 de altura.  
Vamos usar o método ***setVisible***, que recebe um [tipo *boolean*](http://www.javaprogressivo.net/2012/08/java-o-tipo-boolean-base-da-logica_5695.html). Como queremos a janela visível, colocamos true.  
Caso queira definir um tamanho pré-definido, use o método **setSize()**, que recebe dois parâmetros com os pixels da janela (horizontal e vertical):  
**import** javax.swing.JFrame;

**public** **class** framesPanels {

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

JFrame janela = **new** JFrame("Meu primeiro frame em Java");

janela.setSize(300,200);

janela.setVisible(**true**);

}}

Para saber mais funcionalidades da classe JFrame, veja a documentação em:  
<http://docs.oracle.com/javase/6/docs/api/javax/swing/JFrame.html>

|  |  |
| --- | --- |
|  |  |

**JPanel: Inserido elementos em um JFrame**

Como vimos, nosso frame está vazio. Só definimos o tamanho da 'moldura' e o título.  
Vamos adicionar alguns elementos ao nosso frame, e faremos isso inserindo um *JPanel*, onde nele podemos colocar uma infinidade de elementos, chamados JComponents:

|  |
| --- |
| [Como criar JFrame e JPanel](https://1.bp.blogspot.com/-igl9p9NBE6g/UKRqWFG4XXI/AAAAAAAABA8/UJ9pzF93rKQ/s1600/JFrame+e+JPanel+em+Java.gif) |
| <http://www.apl.jhu.edu/~hall/java/Swing-Tutorial/Swing-Tutorial-JPanel.html> |

Para usar o JPanel, importamos essa classe da package swing:  
import javax.swing.JPanel;  
  
Vamos criar uma classe que será nosso painel, vamos chamar de "Painel".  
Para isso, basta fazer com que ela ***extends*** a *JPanel*, como aprendemos em [Herança](http://www.javaprogressivo.net/2012/10/heranca-de-construtores-e-override.html), para que herdemos todas as propriedades do JPanel.

**import** javax.swing.JPanel;

**public** **class** Painel **extends** JPanel{

}

Na nossa classe principal, vamos criar um objeto do tipo "Painel" e chamar de "meuPainel".  
Para adicionar esse JPanel em nosso JFrame, usamos o método ***add*** que recebe como argumento um JPanel:

Painel meuPainel = **new** Painel();

janela.add(meuPainel);

Embora tenha adicionado um Panel ao seu Frame, você não vai ver nada quando rodar, pois não adicionou nenhum elemento ao seu Panel.  
Nos próximos tutoriais desta seção de *GUI*, vamos aprender como adicionar elementos no Panel.  
  
Mas para não deixar você 'na mão', vamos mostrar um código da Panel que vai fazer um desenho bem bacana no seu JPanel/JFrame, o código completo é:  
**framesPanels.java**

**import** javax.swing.JFrame;

**public** **class** framesPanels {

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

JFrame janela = **new** JFrame("Meu primeiro frame em Java");

Painel meuPainel = **new** Painel();

janela.setDefaultCloseOperation( JFrame.EXIT\_ON\_CLOSE );

janela.add(meuPainel);

janela.setSize(600,400);

janela.setVisible(**true**);

}}

**Painel.java**

**import** java.awt.Graphics;

**import** javax.swing.JPanel;

**public** **class** Painel **extends** JPanel{

**public** void paintComponent( **Graphics** g ){

**super**.paintComponent( g );

int pixel=0;

**for**(pixel=0 ; pixel <= getHeight() ; pixel += 10){

g.drawLine(0, pixel, pixel, getHeight());

}

**for**(pixel=getHeight() ; pixel >=0 ; pixel -= 10){

g.drawLine(0, pixel, getHeight() - pixel, 0);

}

**for**(pixel=0 ; pixel <= getHeight() ; pixel +=10){

g.drawLine(getWidth(), pixel, getWidth() - pixel, getHeight());

}

**for**(pixel=getHeight() ; pixel >=0 ; pixel -= 10){

g.drawLine(getWidth(), pixel, getWidth() - (getHeight() - pixel), 0);

}

}

}

[Interface Gráfica JavaSE com NetBeans](https://www.youtube.com/playlist?list=PLWd_VnthxxLfeEUK08gB7G3316OS5xIT3) videos

https://youtu.be/6\_EWvCWlK\_8

https://youtu.be/jgiCWjCA7VM

https://youtu.be/oFXtlOCZ74c

https://youtu.be/zqYRhNbKqG0

https://youtu.be/F5I80fpz3\_M

https://youtu.be/3ZVxYgaK3Yc

https://youtu.be/JyYTaDJkl7A

https://youtu.be/1For7kaUz0Q

https://youtu.be/JqbeKkoAYq4

https://youtu.be/QgRLUMLMEyE

https://youtu.be/QtPHLBbV1Uw

https://youtu.be/NO24OAdJ2Wc

https://youtu.be/W5LsoFlwAWE

https://youtu.be/W6nzDgC9vYY

https://youtu.be/BjwcJME\_r2w

https://youtu.be/ZVZU-AJkJxA

https://youtu.be/Z5zcXRa\_v34

https://youtu.be/zqI7i1FiG0g

https://youtu.be/RjXQAlAYa1k