

Usando a classe **JOptionPane**

(<http://www.tiexpert.net/programacao/java/joptionpane.php>)

1. Uma Introdução à Programação Visual

Após apresentarmos os conceitos estruturais principais de linguagem, a idéia de orientação a objeto e as classes que mais utilizaremos como [String](#) e [Numbers](#), passaremos agora a focar numa programação mais voltada a práticas possíveis, demonstrando classes com funções mais avançadas e importantes, além de abandonarmos um pouco o visual de console e partirmos para a programação visual de Java.

Para introduzir esta nova fase, iniciaremos descrevendo um pouco sobre **JOptionPane** do pacote visual **Swing**.

A classe **JOptionPane** nos proporciona uma série de métodos estáticos que ao serem invocados criam caixas de diálogos simples e objetivas.

Para usar **JOptionPane** temos sempre que importar o pacote **javax.swing.JOptionPane** primeiro.

2. Caixas de Diálogo Input Text

As caixas de diálogo de entrada de texto ou Input Text Dialog servem para fazer uma requisição de algum dado ao usuário de forma bem simples e direta.

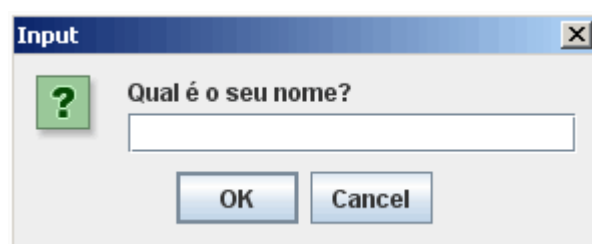
O que é digitado pelo usuário é retornado pelo método em forma de string.

Existem mais de 10 métodos sobrecarregados para invocar uma caixa de diálogo Input Text, mas, a princípio, usaremos a mais simples. O método `showInputDialog` recebe um argumento que é a string contendo a informação desejada, o que na maioria das vezes é uma pergunta ou pedido.

O exemplo abaixo demonstra um programa pedindo para que digite seu nome.

```
1. import javax.swing.JOptionPane;
2.
3. public class CaixasDeDialogo {
4.     public static void main(String[] args) {
5.         JOptionPane.showInputDialog("Qual é o seu nome?");
6.     }
7. }
```

Será apresentada uma janela a seguir.



3. Caixas de Diálogo Confirm

Outra caixa de diálogo simples e objetiva do JOptionPane é a caixa de diálogo de confirmação ou Confirm Dialog.

A Confirm Dialog (caixa de confirmação) consiste de uma caixa contendo uma mensagem, um ícone e três botões: sim, não e cancelar.

Apesar deste ser o aspecto padrão, esta caixa, como qualquer outra de JOptionPane, pode ser facilmente configurada (assunto que será tratado com mais detalhes nas próximas páginas).

No método padrão chamado `showConfirmDialog` usamos dois argumentos:

1. O primeiro é a dependência ou frame pai, ou seja, de qual janela esta confirmação está sendo gerada. Como nossos exemplos iniciais não possuem nenhuma dependência, então, sempre usaremos *null* neste argumento.
2. O segundo argumento é a mensagem que desejamos mostrar ao usuário.

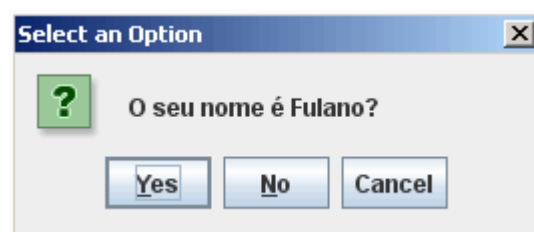
O método `showConfirmDialog` sempre retorna uma constante que é a resposta clicada pelo usuário, que são:

Valor	Nome da Constante	Equivale
0	YES_OPTION	ao clicar no botão Yes (sim)
1	NO_OPTION	ao clicar no botão No (não)
2	CANCEL_OPTION	ao clicar no botão Cancel (cancelar)

Melhorando o exemplo anterior ficaria assim.

```
01.import javax.swing.JOptionPane;
02.
03.public class CaixasDeDialogo {
04.    public static void main(String[] args) {
05.        String nome = null;
06.        nome = JOptionPane.showInputDialog("Qual é o seu nome?");
07.        JOptionPane.showConfirmDialog(null, "O seu nome é " + nome + "?");
08.    }
09.}
```

A caixa de confirmação pareceria da seguinte forma:



4. Caixa de Diálogo de Mensagem

A caixa de diálogo de mensagem é uma caixa que serve apenas para emitir uma mensagem. Esta caixa também é muito configurável e versátil, pois serve para muitas situações distintas como uma mensagem de erro, um alerta, ou simplesmente uma informação.

O método `showMessageDialog` é responsável em trazer a caixa de mensagem, o qual pode ter muitos argumentos, porém, vamos nos ater ao mais simples.

Assim como o método `showConfirmDialog`, usaremos *null* como valor de nosso primeiro argumento, pois, por enquanto, não há dependência de outras janelas em nosso programa. O segundo argumento é a mensagem que desejamos emitir.

Para finalizar nosso exemplo, incluiremos as caixas de mensagem de acordo com as respostas.

```
01.import javax.swing.JOptionPane;
02.
03.public class CaixasDeDialogo {
04.    public static void main(String[] args) {
05.        String nome = null;
06.        int resposta;
07.        nome = JOptionPane.showInputDialog("Qual é o seu nome?");
08.        resposta = JOptionPane.showConfirmDialog(null, "O seu nome é " + nome +
            "?");
09.        if (resposta == JOptionPane.YES_OPTION) {
10.            // verifica se o usuário clicou no botão YES
11.            JOptionPane.showMessageDialog(null, "Seu nome é " + nome);
12.        } else {
13.            JOptionPane.showMessageDialog(null, "Seu nome não é " + nome);
14.        }
15.    }
16.}
```

 **Faça o Download** deste Código

Abaixo está um exemplo de como irá ficar a caixa de mensagem caso clique no botão YES.

