

Desenvolvimento Desktop

Aula 02 - Introdu o ao Swing

Apresentação

Esta aula dará a você uma breve introdução sobre o que é e como usar a JFC (Java Foundation Classes) e o Swing. Na aula anterior, nós já vimos como instalar o NetBeans, que é o ambiente de desenvolvimento que iremos usar nessa e nas próximas aulas. Vimos, também, como ele funciona utilizando um exemplo simples (e sem componentes gráficos). Nesta aula, veremos como o NetBeans funciona, utilizando o ambiente gráfico, e como pode ser simples e fácil desenvolver um aplicativo com uma interface visualmente agradável para o usuário.



Vídeo 01 - Apresentação

Objetivos

Após esta aula, você será capaz de:

- Definir o que é JFC e Swing.
- Entender por que utilizar o Swing no desenvolvimento de aplicativos.
- Compreender o que é uma GUI (Graphical User Interface) e como utilizar alguns componentes para compor uma interface gráfica.
- Inserir comandos simples no código-fonte gerado automaticamente pelo Netbeans e Executar e visualizar o aplicativo desenvolvido.

JFC

JFC ou *Java Foundation Classes* é um conjunto de componentes e serviços que simplificam o desenvolvimento e a implantação de aplicativos para desktop e para a internet. Ela engloba um grupo de recursos para a construção de interfaces gráficas, além de adicionar interatividade e funcionalidades com qualidade profissional aos aplicativos desenvolvidos em Java. A JFC é composta pelos seguintes componentes:

- Componentes Gráficos Swing — incluem os botões, caixas de texto, painéis, rótulos e todos os outros componentes gráficos que compõem um aplicativo de Desktop.
- Suporte a *Look-and-Feel* — é um termo utilizado na apresentação da interface gráfica do usuário e compreende os aspectos da sua concepção, incluindo elementos como cores, formas, disposição e tipos de caracteres (o "Look"), bem como o comportamento de elementos dinâmicos tais como botões, caixas e menus (o "Feel").
- API (Interface de Programação de Aplicativos) de Acessibilidade — aplicativos com interfaces que disponibilizam tecnologia de auxílio que facilitam o uso do sistema por pessoas deficientes.
- API Java 2D — interfaces de programação para aplicativos com gráficos em duas dimensões.
- Internacionalização — facilidades na adaptação de determinado software à língua e cultura de determinado país.

Esta disciplina se concentrará principalmente nos componentes gráficos Swing e em outras funcionalidades Java que não estão entre os componentes JFC, como manipulação de arquivos e empacotamento de programas para distribuição.



Vídeo 02 - Java Foundation Classes

Aprendendo Swing com o NetBeans IDE

Conforme visto na aula anterior, o NetBeans IDE oferece muitas vantagens quando comparado a um editor de textos comum. Dessa forma, recomendamos o seu uso sempre que possível. Ainda na aula anterior, você pôde obter informações sobre como consegui-lo e como instalá-lo no seu computador.

Nesta aula, introduziremos o Swing através da construção, passo a passo, de uma aplicação que faz conversão entre unidades de velocidade, no nosso caso, de milhas por hora (mph) para quilômetros por hora (km/h). Essa aplicação utiliza um pequeno subconjunto dos componentes do Swing e, à medida que você for lendo as próximas aulas, vai se deparar com mais componentes.

Para a construção de aplicativos gráficos, iremos utilizar o construtor de interfaces gráficas do NetBeans IDE, que torna a construção de interfaces com o usuário uma simples tarefa de arrastar e soltar. Sua geração automática de código simplifica bastante o processo de desenvolvimento das GUIs (Graphical User Interface ou Interface Gráfica com o Usuário), permitindo que o foco do programador esteja na lógica da aplicação, e não na infraestrutura necessária ao aplicativo.

Como esta aula é um passo a passo, nós recomendamos que você abra o NetBeans IDE e execute cada etapa na medida em que lê o material. Essa será a maneira mais rápida e fácil de aprender a programar com Swing. Cada passo está ilustrado com a tela que você vai encontrar durante a construção do aplicativo, facilitando a leitura, caso não esteja em frente a um computador com o NetBeans IDE instalado.

A interface Gráfica final deste aplicativo vai se parecer com a mostrada na **Figura 1**, a seguir.

Figura 01 - Janela final do aplicativo a ser desenvolvido



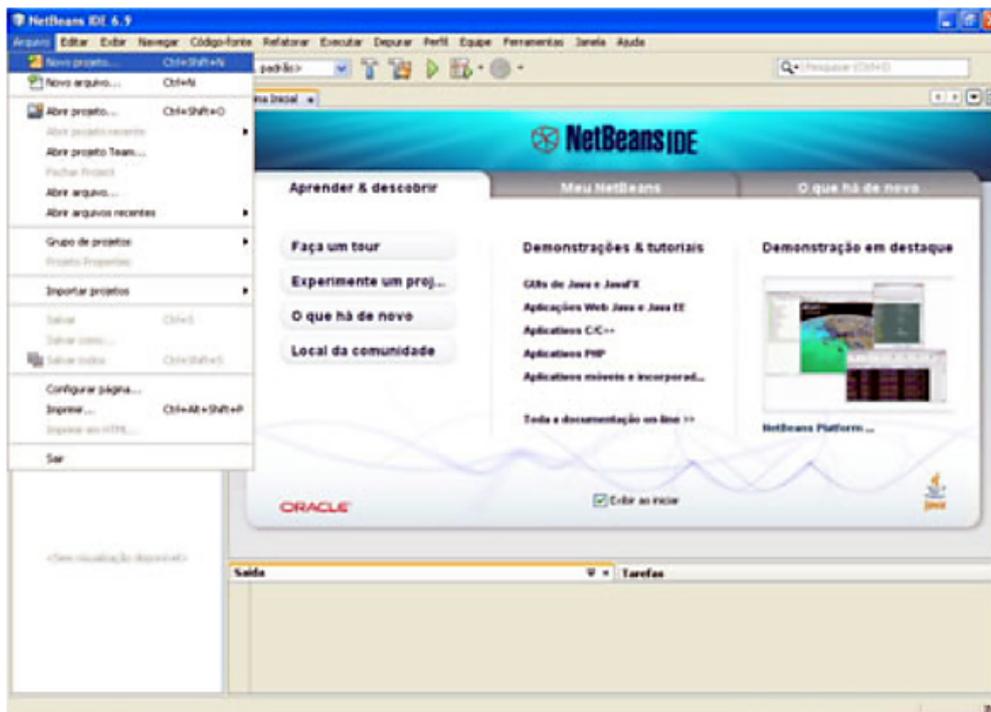
Fonte: NetBeans

Vamos aos passos para a construção do nosso aplicativo:

Passo 1:

1. Para iniciar, abra o NetBeans IDE e crie um novo projeto a partir do menu **Arquivo**. Veja a **Figura 2**. Perceba que os atalhos do teclado para cada comando aparecem à direita de cada opção do menu.

Figura 02 - JCriando um novo projeto



Fonte: NetBeans

Passo 2:

1. Uma nova janela vai apresentar as opções disponíveis. Selecione **Java** na coluna **Categoria** e **Aplicativo Java** na coluna **Projetos**, conforme mostra a **Figura 3**:

Figura 03 - Especificando o novo projeto



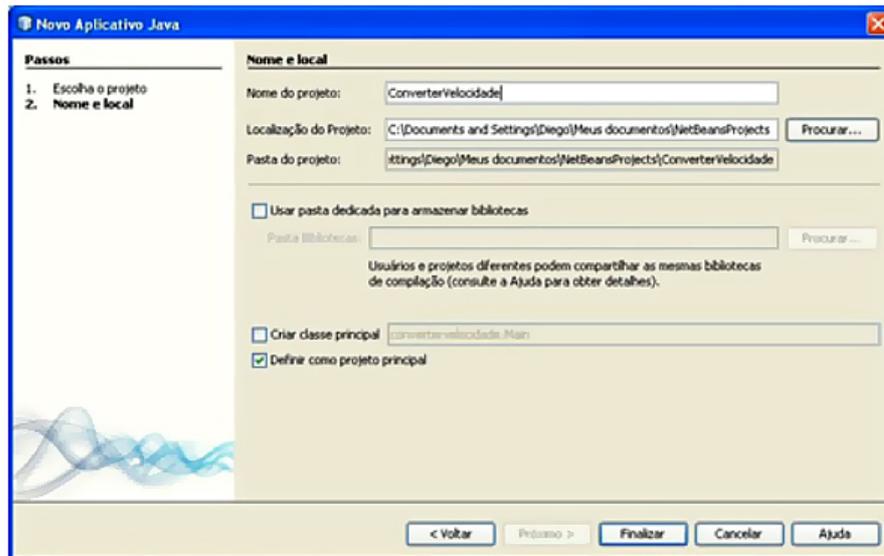
Fonte: NetBeans

2. A caixa de texto inferior descreve cada opção selecionada. Nesse primeiro exemplo, utilizaremos a Categoria Java e o Projeto Aplicativo Java. Com essas opções selecionadas, clique no botão **Próximo >** para continuar.

Passo 3:

1. Defina o nome do projeto: "ConverterVelocidade". Você também pode alterar a localização do projeto e sua pasta, conforme mostra a **Figura 4** a seguir.

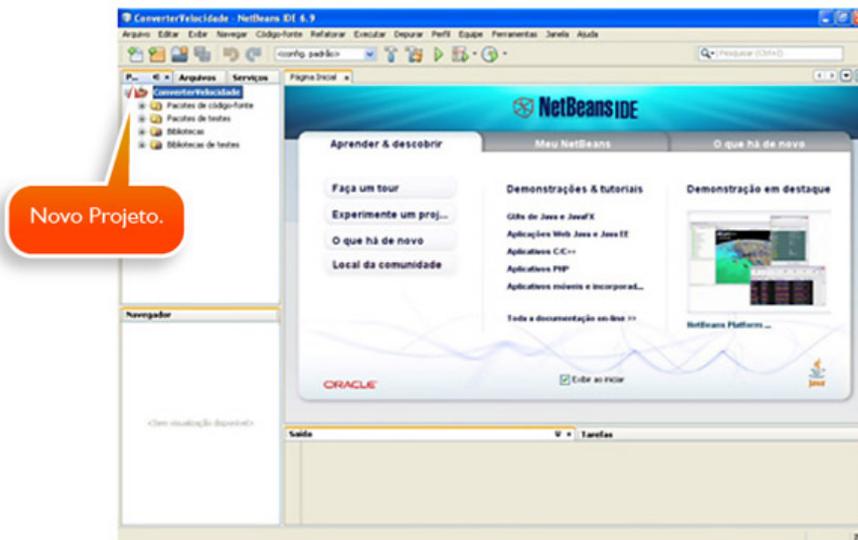
Figura 04 - Finalizando o novo projeto



Fonte: NetBeans

2. Certifique-se de que a opção (*check box*) **Criar classe principal** esteja desmarcada.
3. Clique no botão **Finalizar** para concluir a criação do novo projeto. Quando o carregamento for finalizado, aparecerá uma tela similar à da **Figura 5**.

Figura 05 - Ambiente inicial, após a criação do novo projeto



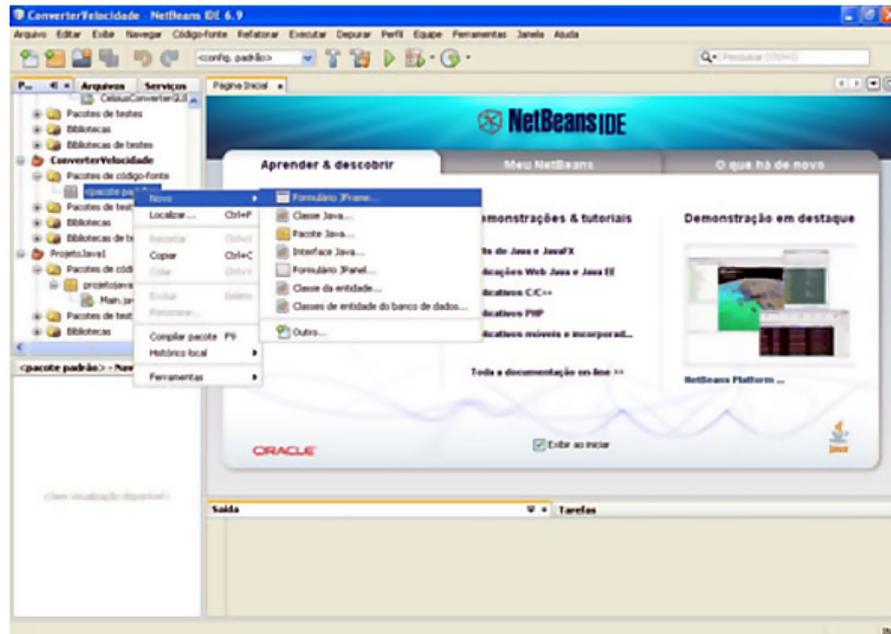
Fonte: NetBeans

4. A guia **Projetos**, no canto superior esquerdo, mostra o projeto recentemente criado.

Passo 4:

1. Adicione um novo formulário **JFrame** clicando com o botão direito do mouse sobre o nome do projeto (**ConverterVelocidade**) e seguindo o caminho, como apresentado na **Figura 6**.

Figura 06 - Inserindo um JFrame



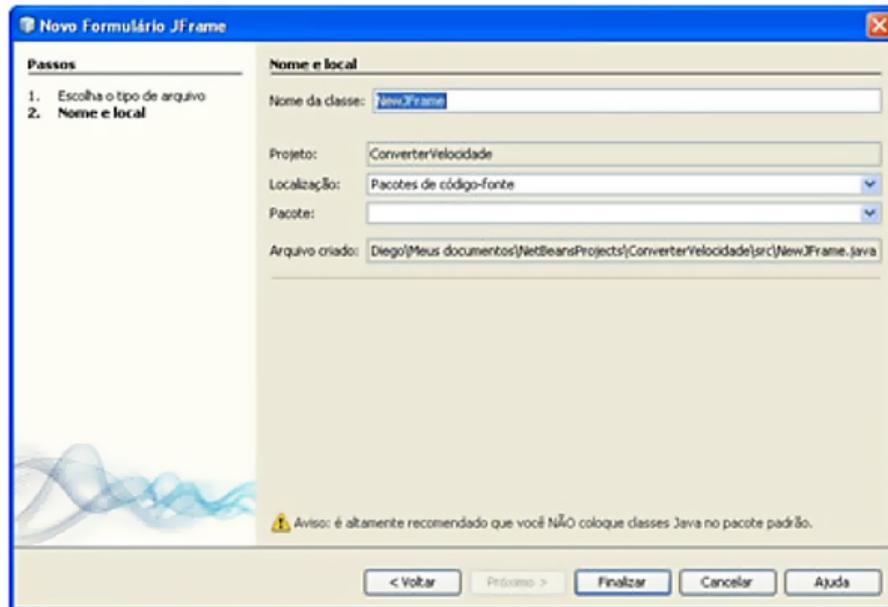
Fonte: NetBeans

2. O **JFrame** é a principal responsável pelo seu projeto, e é a partir dela que estruturaremos todo esse projeto.

Passo 5:

1. Em seguida, defina o nome da classe "**NewFrame**" e o nome do pacote "**ConverterVelocidade**". Você pode definir os nomes como quiser, mas seguiremos com a sugestão proposta pelo ambiente. Veja a **Figura 7**, a seguir.

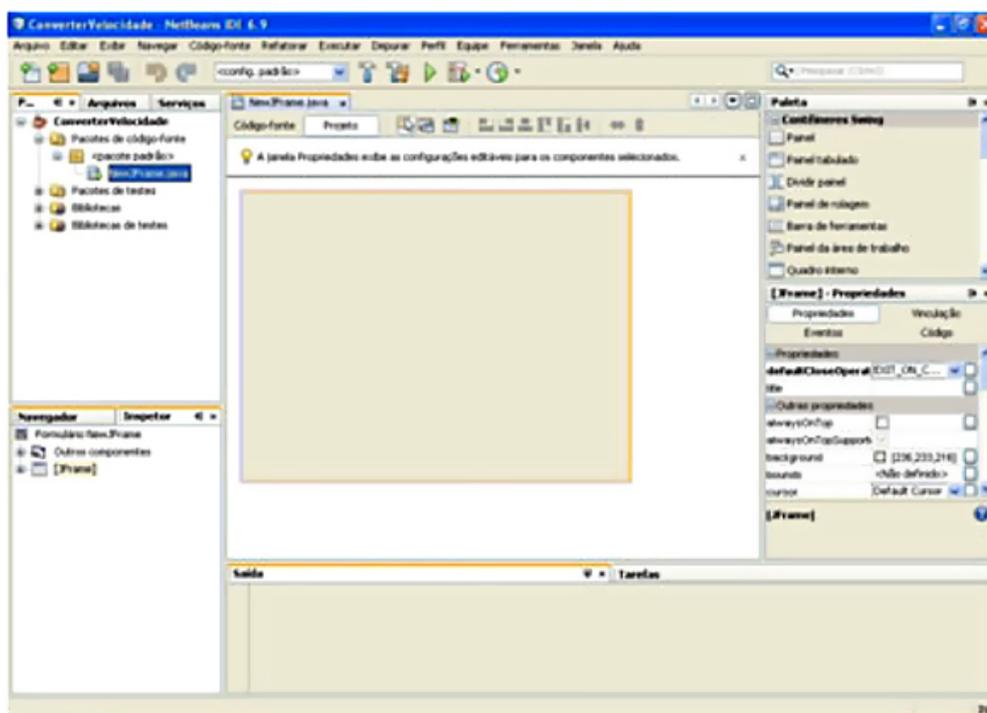
Figura 07 - Especificando detalhes do JFrame



Fonte: NetBeans

- Os campos são automaticamente sugeridos. Clique no botão **Finalizar** quando estiver pronto. Quando a IDE terminar de carregar, à direita, aparecerá a área gráfica, onde visualmente manipularemos os vários componentes do Swing.

Figura 08 - Tela, após a criação do JFrame



Fonte: NetBeans

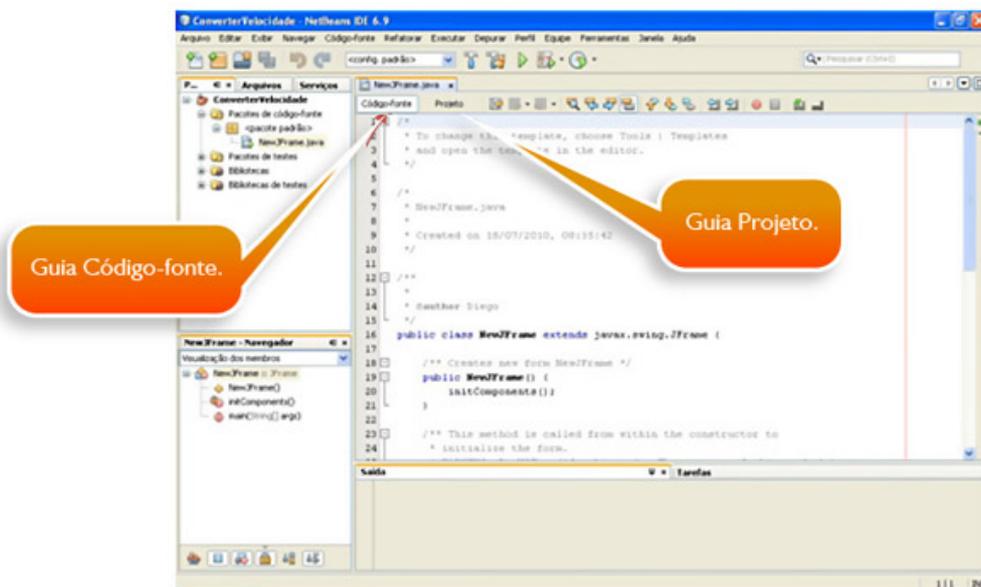
Passo 6:

Inicialmente, apenas demonstraremos como desenvolver esse pequeno projeto de conversão de velocidades. À medida que formos progredindo, entraremos em detalhes em relação aos componentes e suas propriedades.

Área do projeto:

É onde iniciaremos a construção de forma gráfica da nossa GUI. O NetBeans disponibiliza duas opções de visualização: a vista gráfica do projeto (mostrada na figura anterior) e a vista do código fonte (apresentada na **Figura 9** a seguir).

Figura 09 - Visualização do código fonte do projeto



Fonte: NetBeans

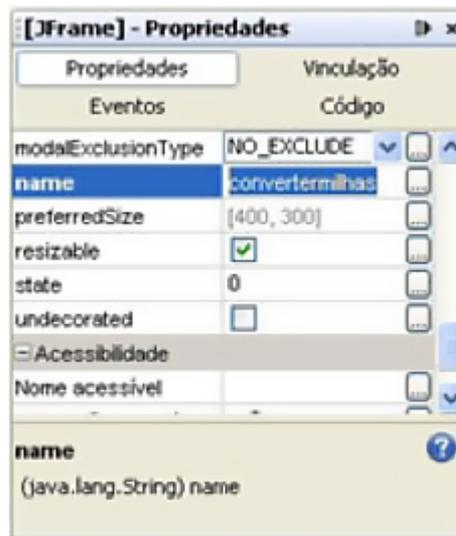
Podemos alternar a visualização facilmente, manipulando as guias em destaque na figura acima. Entretanto, vamos trabalhar com a visualização gráfica; então, clique na guia **Projeto**.

Você estará visualizando o único objeto **JFrame** representado pelo grande retângulo delimitado por uma linha laranja (**Figura 8**). À medida que formos acrescentando outros componentes e alterações, automaticamente o código gerado pela IDE aparecerá na guia código fonte. Vejamos, então, como fazer isso:

1. Clicando no componente **JFrame** aparecerá, no canto inferior direito, o painel de propriedades com várias características do componente que

poderão ser alteradas facilmente. Localize o campo **name** e altere o nome do **JFrame** para “**convertermilhas**”, conforme mostra a **Figura 10** a seguir.

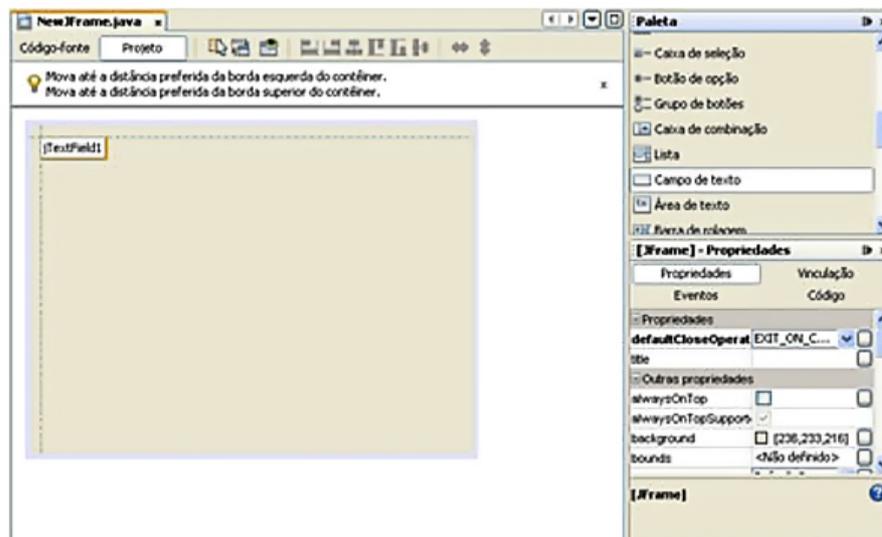
Figura 10 - Painel de propriedades



Fonte: NetBeans

2. Em seguida, na janela à direita, clique e arraste um campo de texto (**JTextField**) para dentro da área do **JFrame**. Ao localizar o **JTextField** dentro do **JFrame**, o construtor do GUI fornece as linhas tracejadas para auxiliar no ajuste de localização do objeto. Usando as linhas-guia, localize o objeto no canto superior esquerdo, como mostrado na **Figura 11** a seguir.

Figura 11 - Inserindo um campo de texto

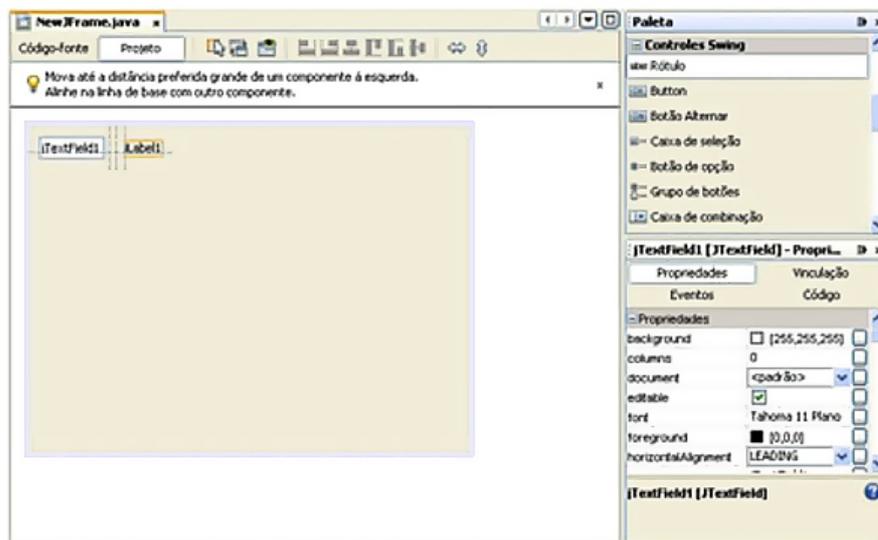


Fonte: NetBeans

3. Não se preocupe em fazer os ajustes agora. Faremos isso mais tarde, quando estivermos com todos os componentes no **JFrame**.

- Agora adicione o rótulo (**JLabel**), seguindo as linhas-guia, posicionando-o ao lado do **JTextField**, como mostra a **Figura 12**.

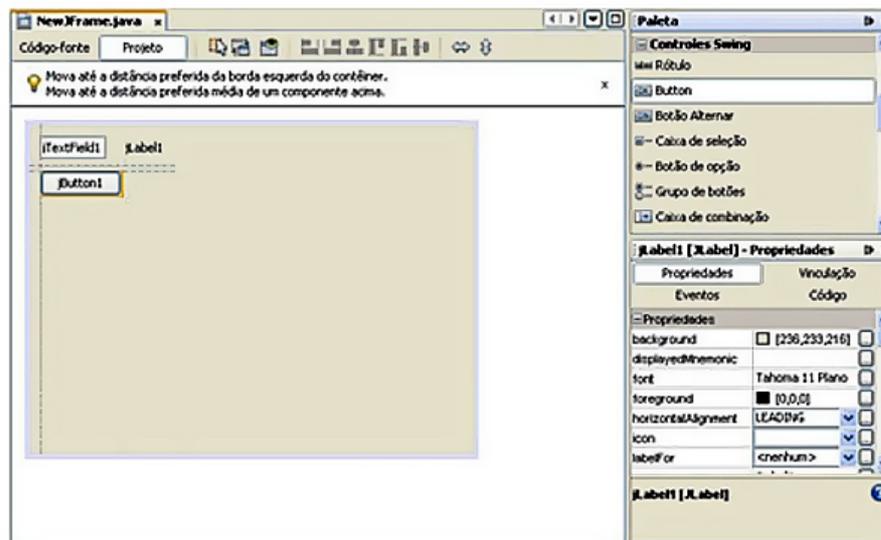
Figura 12 - Inserindo um rótulo



Fonte: NetBeans

- As linhas de ajuste auxiliam no alinhamento, facilitando a melhor distribuição dos componentes.
- Agora, adicione um botão (**JButton**) logo abaixo do **JTextField**. Lembre-se de utilizar as linhas guia para alinhar os componentes. Veja a **Figura 13**.

Figura 13 - Inserindo um botão

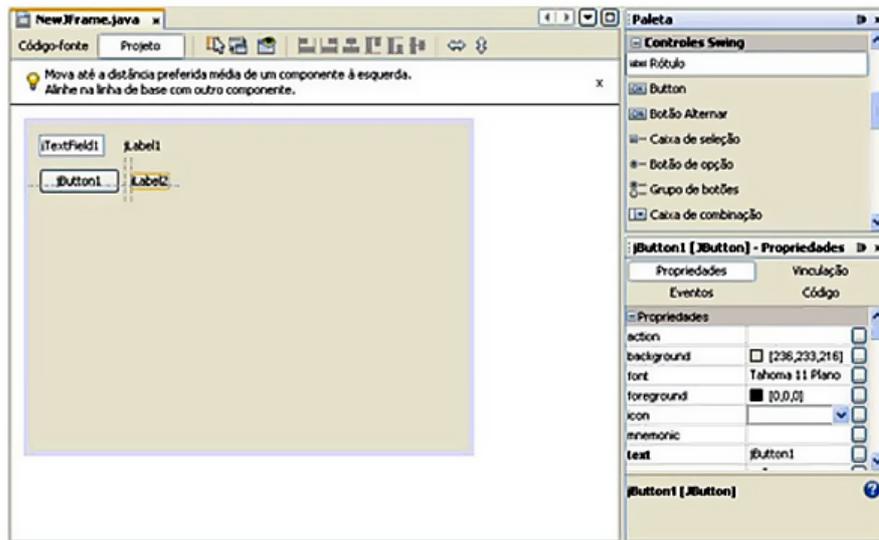


Fonte: NetBeans

- Não tente alterar qualquer propriedade ainda. Faremos as alterações de nomes e dimensões mais tarde.

8. Agora adicione um segundo **JLabel** ao lado do botão, conforme mostra a **Figura 14** a seguir.

Figura 14 - Inserindo um segundo rótulo



Fonte: NetBeans

9. Pronto. Agora temos todos os componentes que precisaremos para concluir o exemplo proposto: converter velocidade de milhas/hora em quilômetros/hora. Só faltam os ajustes.

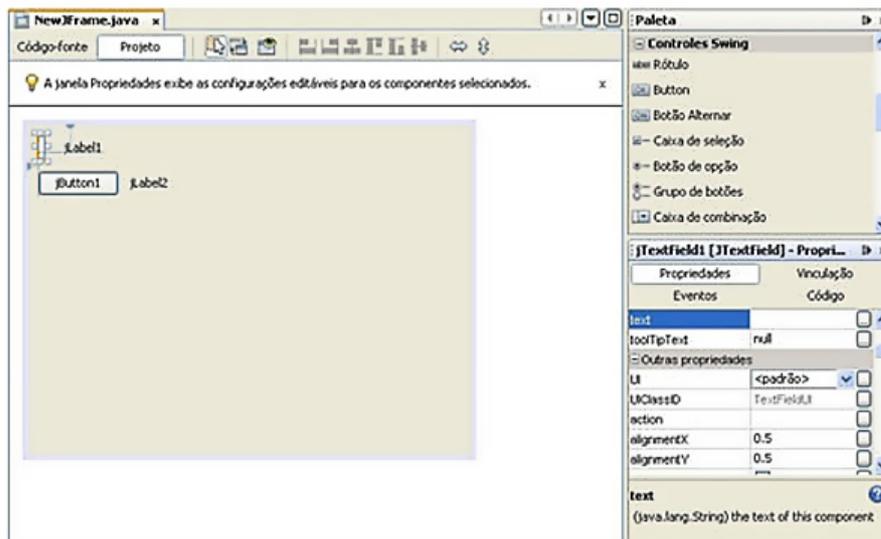


Vídeo 03 - Aplicação Swing e Construção da Interface Gráfica

Ajustando os Componentes

1. Repare que logo acima do **JFrame** aparece uma pequena janela com algumas dicas que podem lhe auxiliar. Caso queira fechar essas dicas, basta clicar no pequeno quadrado (com um "X") à direita.
2. Para alterar o texto do **JTextField**, clique no componente e procure no painel de propriedades o campo **text**. Ao apagar o texto padrão JTextField1, que constava anteriormente, as dimensões são alteradas automaticamente, como mostra a **Figura 15** a seguir.

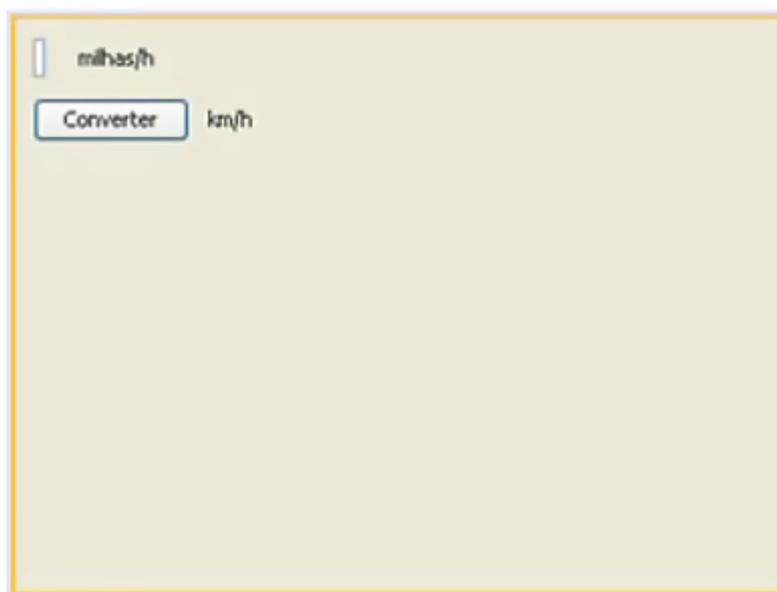
Figura 15 - Alterando a propriedade text



Fonte: NetBeans

3. Como não queremos nenhum texto inicial no **JTextField**, manteremos esse campo vazio. Agora, de forma similar, altere o campo **text** dos outros componentes, adicionando os nomes sugeridos, como na **Figura 16** (**JLabel1** "milhas/h", **JBUTTON** "Converter" e **JLabel2** "Km/h").

Figura 16 - Componentes com a propriedade text alterada

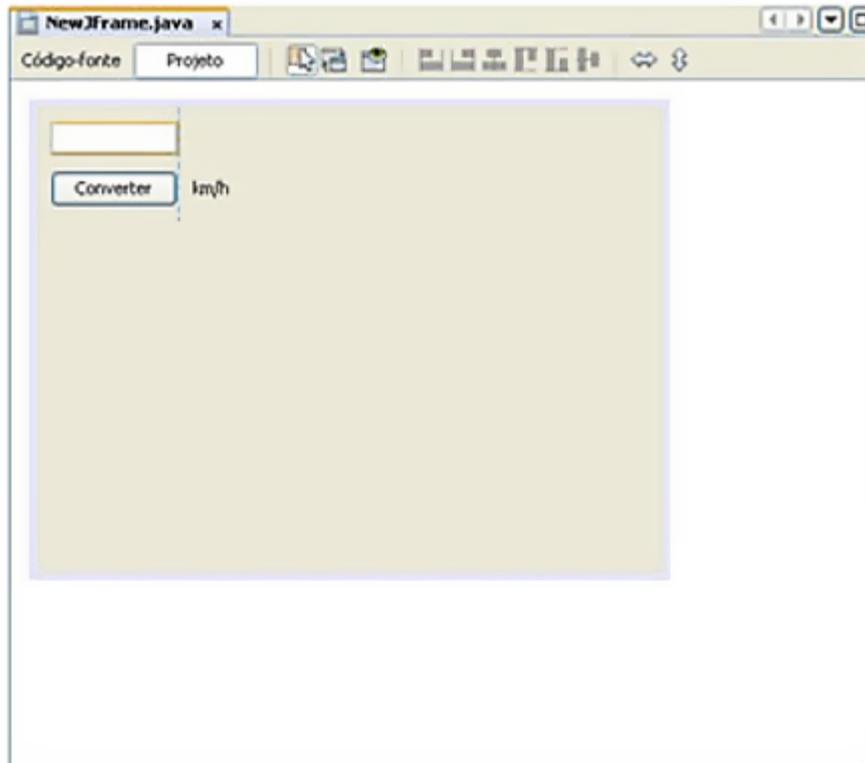


Fonte: NetBeans

4. Agora sim, vamos ajustar as dimensões dos componentes. Clique no **JTextField**. Repare nos pequenos quadrados que aparecem no contorno do componente.

5. Clique no pequeno quadrado do lado direito do componente. Segure o botão do mouse e arraste o cursor para ajustar no tamanho desejado, sem se preocupar em ocultar outros componentes que estejam próximos. Veja a **Figura 17** a seguir:

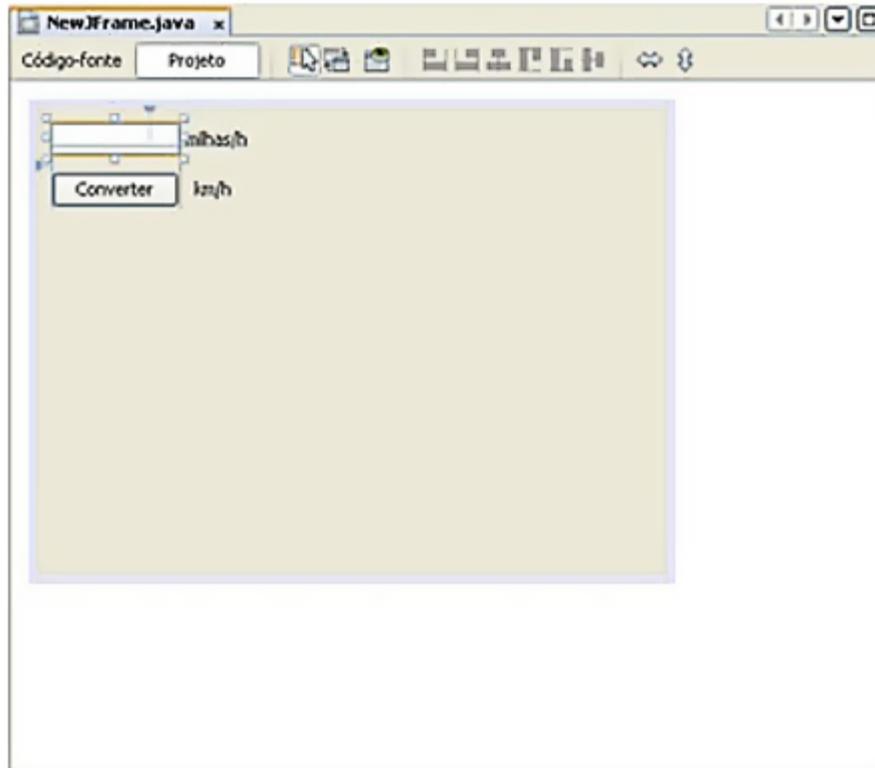
Figura 17 - Alterando as dimensões do JTextFiled



Fonte: NetBeans

6. Ao soltar o botão do mouse, os outros componentes se ajustam automaticamente às novas posições. Veja a **Figura 18** a seguir:

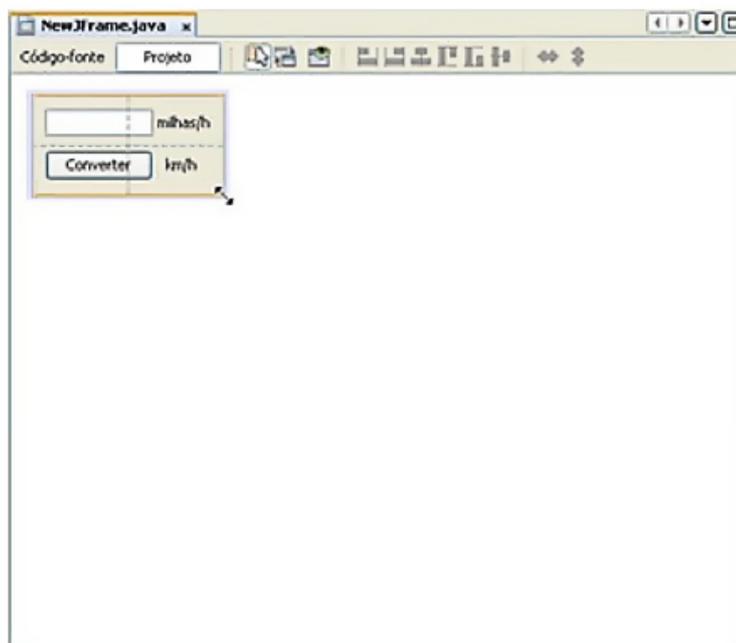
Figura 18 - Componentes com posicionamento ajustado



Fonte: NetBeans

7. Por fim, ajuste o tamanho do **JFrame**. Posicione o cursor no canto extremo inferior direito e arraste até obter as dimensões desejadas, eliminando todo o espaço extra. Veja a **Figura 19** a seguir:

Figura 19 - Ajustando as dimensões do JFrame

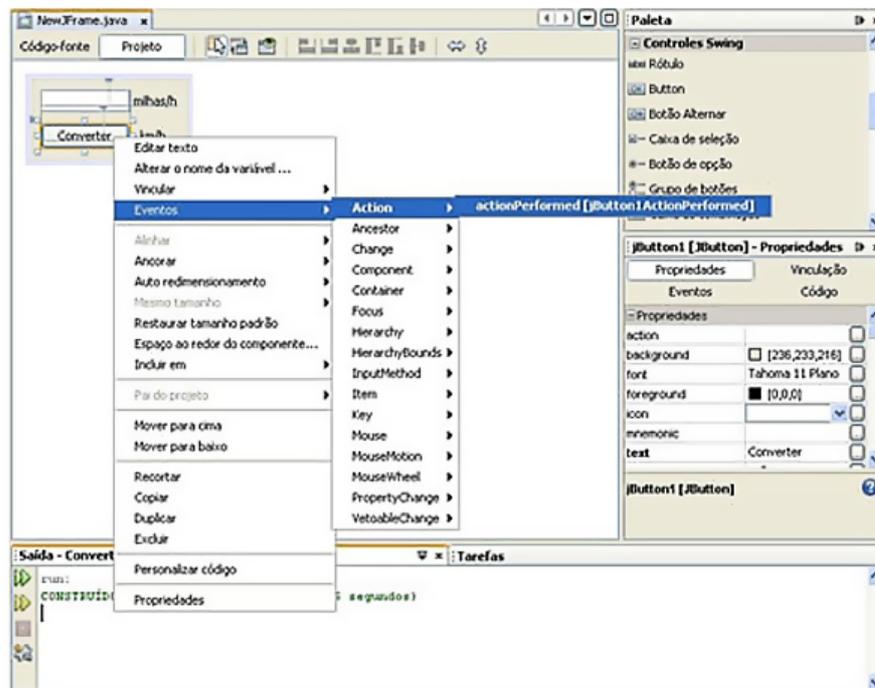


Fonte: NetBeans

8. A parte realizada na GUI está completa. Vamos, agora, concluir o aplicativo inserindo os comandos de conversão ao botão.

9. Clique no botão **Converter** com o botão direito do mouse. Siga a sequência de opções conforme mostra a **Figura 20** a seguir:

Figura 20 - Inserindo um evento no botão



Fonte: NetBeans

10. Aparecerá na tela a área onde deverá ser incluído o código que será executado quando o botão for apertado.

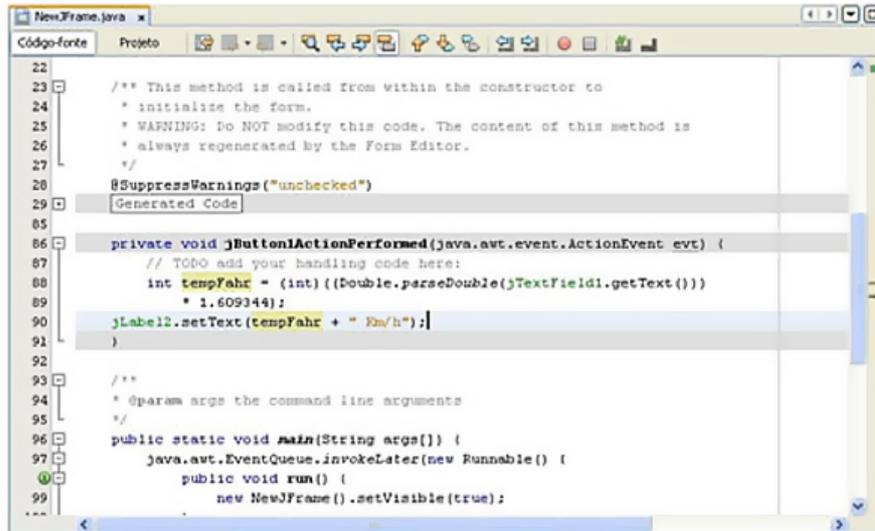
11. Insira a linha de comando especificada abaixo

```
1 int tempFahr = (int)((Double.parseDouble(jTextField1.getText())) * 1.609344); <br>
2 jLabel2.setText(tempFahr + " Km/h");
```

Perceba que o `jTextField1` faz referência ao nome do atributo do campo de texto que foi criado e o `jLabel2` ao nome do segundo rótulo (Km/h).

12. Não se preocupe, agora, em entender o código. Posteriormente, vamos detalhar exemplos similares. O código-fonte deve ficar conforme mostrado na **Figura 21** a seguir:

Figura 21 - Inserindo a linha de código no botão

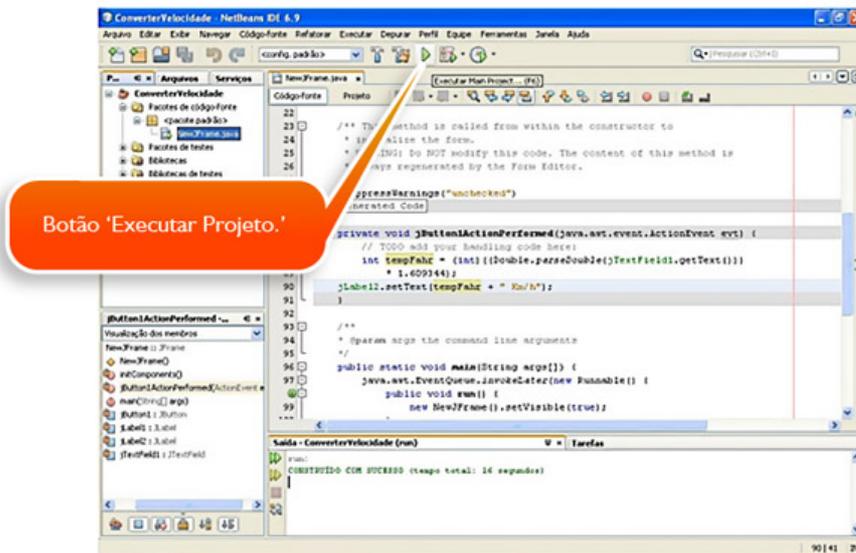


```
22
23 /** This method is called from within the constructor to
24  * initialize the form.
25  * WARNING: Do NOT modify this code. The content of this method is
26  * always regenerated by the Form Editor.
27  */
28 @SuppressWarnings("unchecked")
29 Generated Code
30
31
32
33
34
35
36 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
37     // TODO add your handling code here:
38     int tempFahr = (int) (Double.parseDouble(jTextField1.getText())
39         * 1.609344);
40     jLabel2.setText(tempFahr + " Km/h");
41 }
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Fonte: NetBeans

13. Pronto, finalizamos o aplicativo. Agora, vamos executar o que construímos. Na barra de tarefas superior, clique no ícone indicado na **Figura 22** ou utilize a tecla de atalho **F6**.

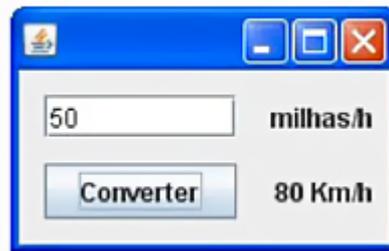
Figura 22 - Visualização final do código fonte



Fonte: NetBeans

14. A janela do aplicativo aparecerá no canto superior esquerdo, como apresentado lá no início, antes de começarmos a construir o aplicativo (veja a **Figura 23**).

Figura 23 - Janela final do aplicativo



15. Teste o aplicativo e confira se o resultado está de acordo com o esperado, conforme mostra a **Figura 23**.

Veja que o resultado da conversão aparece no **JLabel2**, e que o resultado aparece juntamente com o texto "**km/h**", substituindo o texto anterior. Agora, exercite o que aprendemos nesta aula fazendo a atividade proposta.



Vídeo 04 - Adicionando Funcionalidades

Atividade 01

1. Desenvolva um aplicativo que converta um valor qualquer em real para dólar. Além dos componentes utilizados no exemplo desta aula, use mais um **JTextField** para a informação da taxa do dólar.

Conclusão

Bem, chegamos ao final de nossa segunda aula. Esperamos que você tenha aprendido as facilidades que o uso de uma ferramenta gráfica pode proporcionar, entendendo o quanto o NetBeans e o Swing podem contribuir no desenvolvimento de quaisquer aplicativos, dos mais simples aos mais complexos. Nas próximas aulas, exploraremos outros componentes, apresentando com mais detalhes as funcionalidades que o Swing disponibiliza.

Resumo

Nesta aula, vimos como explorar o ambiente gráfico do NetBeans IDE 6.9, utilizando alguns componentes de controle do Swing. Você aprendeu a construir um aplicativo simples de conversão de unidades, usando a facilidade do ambiente gráfico e linhas de código para definir a ação dos componentes utilizados. A partir das próximas aulas, veremos outros componentes e como podemos utilizá-los para criar aplicativos mais complexos.

Autoavaliação

1. Explique o que é JFC.
2. Que componentes do JFC foram utilizados no exemplo desenvolvido nesta aula?
3. Quais as facilidades que o Swing pode proporcionar no desenvolvimento de aplicativos?
4. Qual o resultado da execução de um projeto desenvolvido no ambiente gráfico?

Referências

THE JAVA TUTORIALS. **Lesson:** Learning swing with the NetBeans IDE. Disponível em:
<http://download.oracle.com/docs/cd/E17409_01/javase/tutorial/uiswing/learn/index.html>. Acesso em: 25 abr. 2012.