

Criando uma GUI Swing no NetBeans IDE

Baixe [quickstart-gui.zip](#) como um arquivo zip de leitura offline


Esse tutorial o conduzirá no processo de criação da interface gráfica do usuário (GUI) de uma aplicação denominada ContactEditor por meio do Construtor de GUIs do NetBeans IDE. No processo, você fará o layout de uma GUI inicial que possibilita exibir e editar informações de contato dos indivíduos incluídos em um banco de dados de funcionários.



Nesse tutorial, você aprenderá a:

- Usar a interface do Construtor de GUI
- Criar um contêiner de GUIs
- Adicionar componentes
- Redimensionar componentes
- Alinhar componentes
- Ajustar a ancoragem dos componentes
- Definir o comportamento de redimensionamento automático dos componentes
- Editar as propriedades dos componentes

Esse tutorial demora aproximadamente 30 minutos para ser concluído.

Observação: sempre que o  ícone for mostrado, você poderá clicar para visualizar uma demonstração interativa detalhada da funcionalidade do Construtor de GUIs.

Para concluir este tutorial, você precisa dos seguintes recursos e softwares.

Software ou recurso	Versão necessária
NetBeans IDE	versão 6.5, 6.1 ou 6.0
Java Development Kit (JDK)	versão 6 ou versão 5

Guia de introdução

O Construtor de GUIs do IDE torna possível construir GUIs de aparência profissional sem um conhecimento profundo dos gerenciadores de layout. Você pode criar formulários, simplesmente colocando componentes onde desejar.

Para obter descrições sobre os comentários visuais do Construtor de GUIs, você pode usar a [Legenda de comentários visuais do Construtor de GUIs](#).

Criando um projeto

Como todo o desenvolvimento em Java no IDE ocorre dentro dos projetos, precisamos primeiro criar um projeto ContactEditor para armazenar as fontes e outros arquivos do projeto. Um projeto do IDE é um grupo de arquivos de Códigos-fonte Java e metadados associados que inclui arquivos de propriedades específicos do projeto, um script de construção Ant que controla as configurações de construção e execução e um arquivo project.xml que mapeia destinos Ant para comandos do IDE. Embora as aplicações Java geralmente consistam em vários projetos do IDE, para o objetivo desse tutorial, construiremos uma aplicação simples, inteiramente armazenada em um único projeto.

Para criar um novo projeto de aplicação ContactEditor:

1. Escolha Arquivo > Novo projeto. Como alternativa, você pode clicar no ícone Novo projeto na barra de ferramentas do IDE.
2. No painel Categorias, selecione o nó Java e no painel Projetos, escolha Aplicação Java. Clique em Próximo.
3. Insira ContactEditor no campo Nome do projeto e especifique o local do projeto.
4. Deixe desmarcada a caixa de verificação Usar pasta dedicada para armazenar bibliotecas. (Se você estiver usando o NetBeans IDE 6.0, essa opção não está disponível.)
5. Verifique se a caixa de verificação Definir como projeto principal está marcada e limpe o campo Criar classe principal.
6. Clique em Terminar.
O IDE criará a pasta `ContactEditor` no sistema no local designado. Essa pasta contém todos os arquivos associados ao projeto, como o script Ant, pastas para armazenamento de fontes e testes e uma pasta para os metadados específicos do projeto. Para exibir a estrutura do projeto, use a janela Arquivos do IDE.

Criar um contêiner de GUIs

Após criar a nova aplicação, você pode ter percebido que a pasta Pacotes de códigos-fonte na janela Projetos contém um nó `<pacote padrão>` vazio. Para prosseguir com a construção da nossa interface, precisamos criar um contêiner Java no qual colocaremos os outros componentes requeridos da GUI. Nessa etapa criaremos um contêiner usando o componente

JFrame e o colocaremos em um novo pacote.

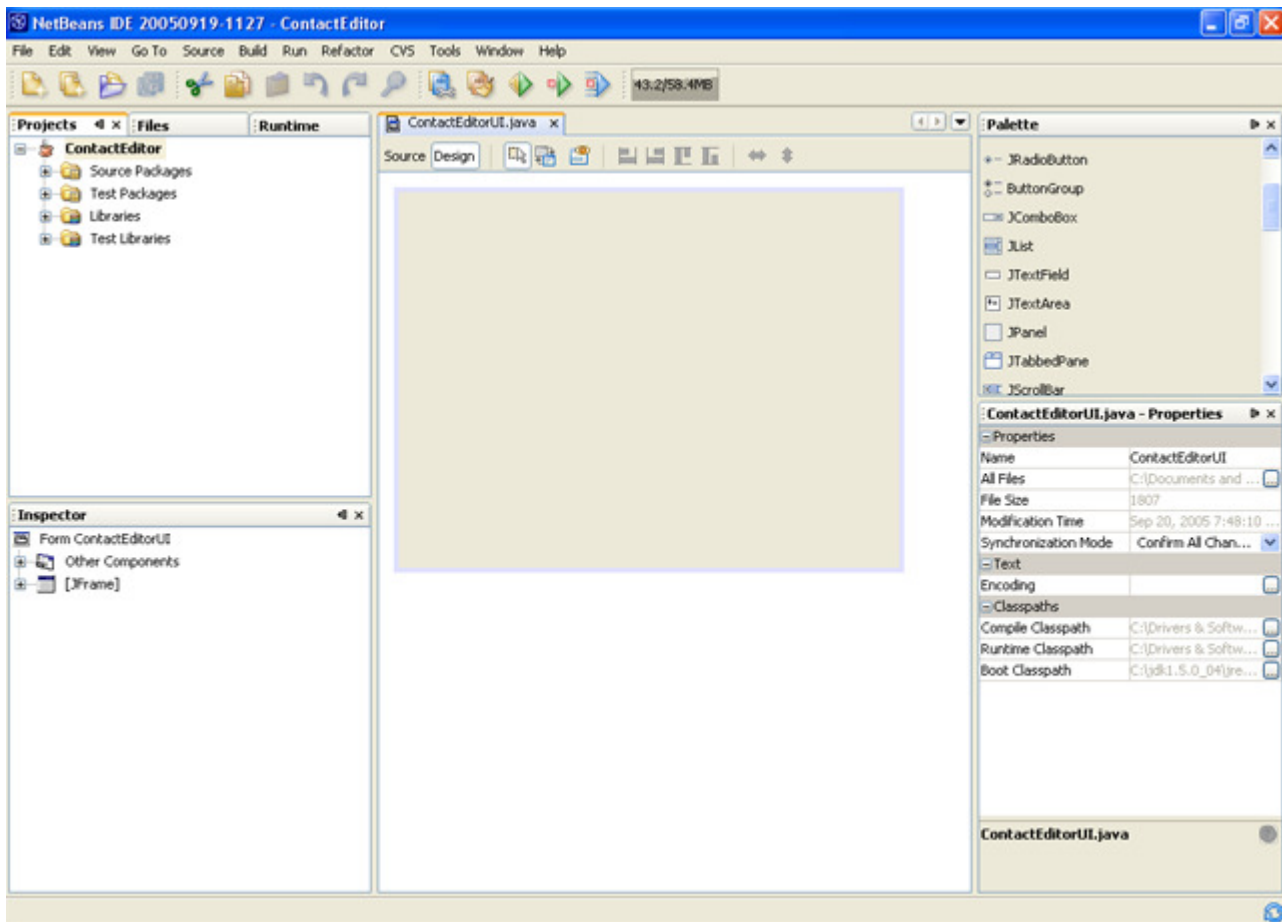
Para criar um contêiner JFrame:

1. Na janela Projetos, clique com o botão direito do mouse no nó ContactEditor e selecione Novo > Formulário JFrame.
2. Insira ContactEditorUI como Nome da classe.
3. Insira my.contacteditor como nome do pacote.
4. Clique em Terminar.
O IDE criará o formulário ContactEditorUI e a classe ContactEditorUI na aplicação ContactEditorUI.java e abrirá o formulário ContactEditorUI no Construtor de GUIs. Observe que o pacote my.contacteditor substitui o pacote padrão.

[início](#)

Familiarizando-se com o Construtor de GUIs

Agora que já configuramos o novo projeto para a aplicação, vamos nos familiarizar com a interface do Construtor de GUIs. Para explorar a interface do Construtor de GUIs com uma demonstração interativa, clique no ícone Exibir demonstração.



 [Exibir demonstração](#)

Quando adicionamos o contêiner JFrame, o IDE abriu o formulário ContactEditorUI recém-criado em uma aba do Editor com uma barra de ferramentas que contém vários botões, como mostra a ilustração anterior. O formulário ContactEditor aberto na exibição Design do Construtor de GUIs e três janelas adicionais apareceram automaticamente ao longo das margens do IDE, permitindo navegar, organizar e editar formulários de GUI à medida que você os constrói.

As várias janelas do Construtor de GUIs incluem:

- **Área de design.** Janela principal do Construtor de GUIs para a criação e edição de formulários de GUI Java. Os botões de alternância Fonte e Design da barra de ferramentas permitem exibir o código-fonte de uma classe ou uma exibição gráfica dos seus componentes de GUI. Os botões adicionais da barra de ferramentas oferecem acesso conveniente aos comandos comuns, como escolher entre os modos Seleção e Conexão, alinhamento de componentes, definição do comportamento de dimensionamento automático e exibição de formulários.
- **Inspetor.** Fornece uma representação de todos os componentes, tanto visuais como não visuais, na aplicação como uma hierarquia em árvore. O Inspetor também oferece comentários visuais sobre qual componente da árvore está sendo editado no Construtor de GUIs, bem como permite organizar componentes nos painéis disponíveis.
- **Paleta.** Uma lista personalizável dos componentes disponíveis contendo abas para os componentes JFC/Swing, AWT e JavaBeans, bem como gerenciadores de layout. Além disso, é possível criar, remover e reorganizar as categorias exibidas na Paleta usando o

personalizador.

- **Janela Propriedades.** Exibe as propriedades do componente atualmente selecionado no Construtor de GUIs ou nas janelas Inspetor, Projetos ou Arquivos.

Se você clicar no botão Fonte, o IDE exibirá o código-fonte Java da aplicação no Editor com seções de código geradas automaticamente pelo Construtor de GUIs e indicadas por áreas azuis, denominadas Blocos protegidos. Blocos protegidos são áreas protegidas que não são editáveis na exibição Fonte. Só é possível editar o código exibido nas áreas brancas do Editor na exibição Fonte. Se for necessário fazer alterações no código em um Bloco protegido, clicar no botão Design fará com que o Editor do IDE volte ao Construtor de GUIs, onde você poderá fazer os ajustes necessários ao formulário. Quando você salvar as alterações, o IDE atualizará as fontes do arquivo.

Observação: para os desenvolvedores avançados, está disponível um Personalizador de paleta e permite adicionar componentes personalizados de JARs, bibliotecas ou projetos à Paleta.

[início](#)

Conceitos-chave

O Construtor de GUIs do IDE resolve o problema central da criação de GUIs Java otimizando o fluxo de dados da criação de interfaces gráficas, liberando os desenvolvedores das complexidades dos gerenciadores de layout do Swing. Ele faz isso estendendo o Construtor de GUIs atual do NetBeans para dar suporte ao paradigma claro "Design livre" com regras simples de layout, fáceis de compreender e usar. À medida que você faz o layout do formulário, o Construtor de GUIs fornece linhas de guia visuais sugerindo espaçamento e alinhamento ótimo dos componentes. Em segundo plano, o Construtor de GUIs converte as decisões de design em uma IU funcional implementada com o uso de um gerenciador de layouts GroupLayout e outras construções do Swing. Como ele usa um modelo de layout dinâmico, as GUIs construídas com o Construtor de GUIs se comportam como o esperado em tempo de execução, ajustando para acomodar as alterações feitas sem alterar os relacionamentos definidos entre os componentes. Quando você redimensionar o formulário, alterar locais ou especificar aparência diferente, a GUI se ajustará automaticamente para respeitar a aparência das inserções e dos deslocamentos.

Desenho livre

No Construtor de GUIs do IDE, você pode construir formulários simplesmente colocando os componentes onde quiser, como se estivesse usando posicionamento absoluto. O Construtor de GUIs descobre quais gerenciadores de layout são obrigatórios e gera o código para você automaticamente. Você não precisa se preocupar com inserções, âncoras, preenchimentos e etc.

Posicionamento automático dos componentes (Encaixe)

À medida que você adiciona componentes a um formulário, o Construtor de GUIs fornece comentários visuais que auxiliam no posicionamento dos componentes de acordo com a aparência do seu sistema operacional. O Construtor de GUIs oferece dicas in-line úteis e outros comentários visuais em relação ao local no qual os componentes devem ser colocados no formulário, encaixando automaticamente os componentes na posição ao longo das linhas de guia. Ele faz essas sugestões de acordo com a posição dos componentes já colocados no formulário, enquanto permite que o enchimento permaneça flexível, de maneira que a aparência diferente do destino seja processada corretamente em tempo de execução.

Comentário visual

O Construtor de GUIs também oferece comentários visuais sobre as relações de ancoragem e encadeamento de componentes. Esses indicadores permitem identificar rapidamente as várias relações de posicionamento e o comportamento fixo dos componentes que afetam a maneira como a GUI aparecerá e se comportará em tempo de execução. Isso acelera o processo de design de GUIs, permitindo que você crie rapidamente interfaces visuais com aspecto profissional e que funcionam.

[início](#)

O principal primeiro

Agora que você está familiarizado com a interface do Construtor de GUIs, esse é o momento para começar a desenvolver a IU da nossa aplicação ContactEditor. Nesta seção veremos a utilização da Paleta do IDE para adicionar os vários componentes da GUI necessários ao nosso formulário.

Graças ao paradigma Design livre do IDE, não é necessário brigar com os gerenciadores de layout para controlar o tamanho e posição dos componentes nos contêineres. Tudo o que você precisa fazer é arrastar e soltar os componentes necessários no formulário da GUI conforme exibido nas ilustrações abaixo.

Adicionando componentes: os Fundamentos

Embora o Construtor de GUIs simplifique o processo de criação de GUIs Java, é sempre útil esboçar a maneira como quer que a interface se pareça antes de começar a fazer seu layout. Muitos designers de interfaces consideram isso uma técnica de "melhor prática", entretanto, para esse tutorial, você pode simplesmente dar uma olhada como nosso formulário preenchido deve se parecer indo para a seção [Visualização da GUI](#).

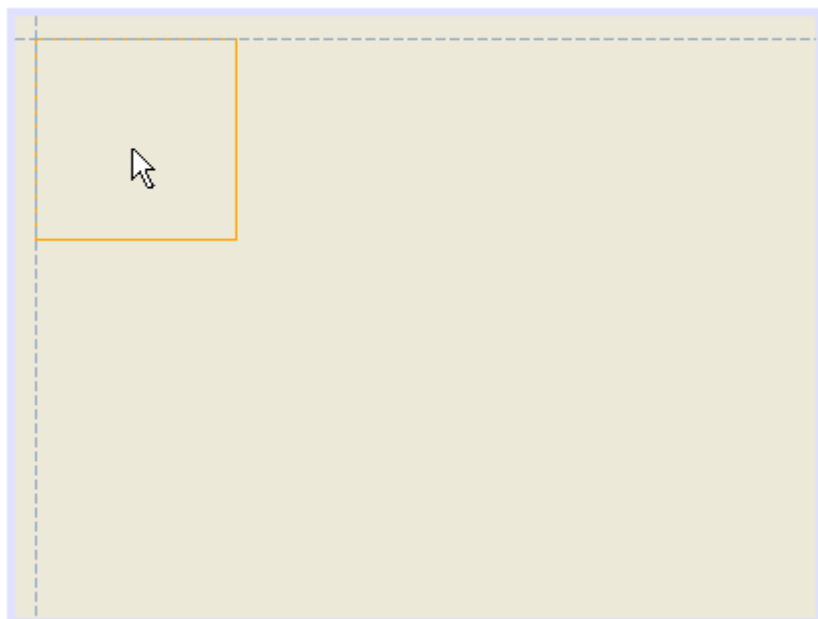
Como já adicionamos um JFrame como contêiner de nível superior do formulário, a próxima etapa é adicionar dois JPanels que permitirão unir os componentes da nossa IU usando bordas com títulos. Veja as ilustrações abaixo e observe o comportamento "arrastar e soltar" do IDE para fazer isso.

Para adicionar um JPanel:

1. Na janela Paleta, selecione o componente Painel na categoria Swing, clicando e soltando o botão do mouse.
2. Mova o cursor até o canto esquerdo superior do formulário no Construtor de GUIs. Quando o

componente estiver localizado ao lado das margens superior e esquerda do contêiner, linhas de guia de alinhamento horizontal e vertical serão exibidas indicando as margens preferidas. Clique no formulário para colocar o JPanel nesse local.

O componente `JPanel` será exibido no formulário `ContactEditorUI` com realce na cor laranja, que significa que está selecionado, como mostra a ilustração abaixo. Após soltar o botão do mouse, pequenos indicadores serão exibidos para mostrar as relações de ancoragem dos componentes e um nó `JPanel` correspondente será exibido na janela Inspecor.

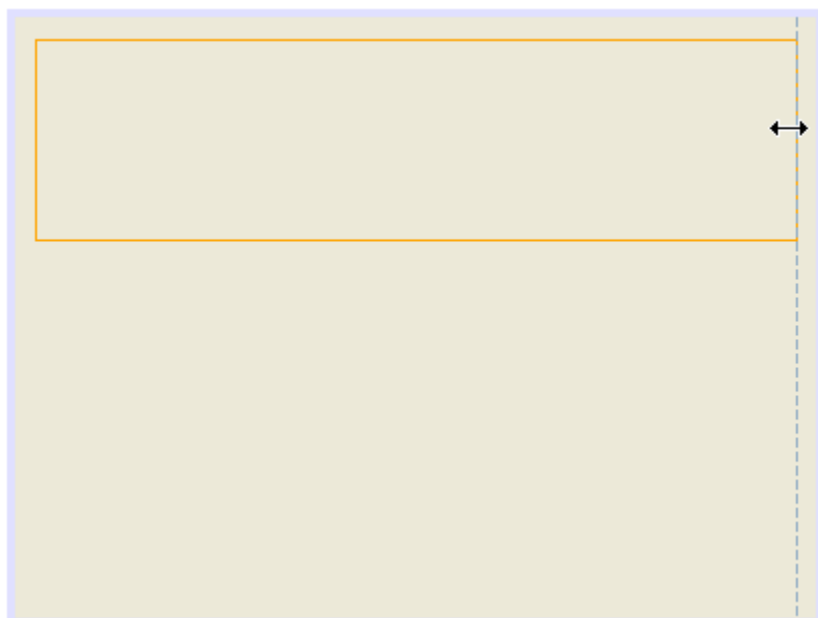


Depois, precisamos redimensionar o JPanel para dar espaço aos componentes que serão inseridos posteriormente, mas tomemos um minuto para indicar antes outros recursos de visualização do Construtor de GUIs. Para fazer isso, precisamos cancelar a seleção do JPanel que acabamos de adicionar. Como nós ainda não adicionamos uma borda com título, o painel desaparecerá. Observe, entretanto, que quando você passar o cursor sobre o JPanel, as margens mudarão para cinza claro para que sua posição possa ser vista claramente. Você precisa apenas clicar em qualquer lugar no componente para selecioná-lo novamente e fazer com que as alças de redimensionamento e os indicadores de ancoragem reapareçam.

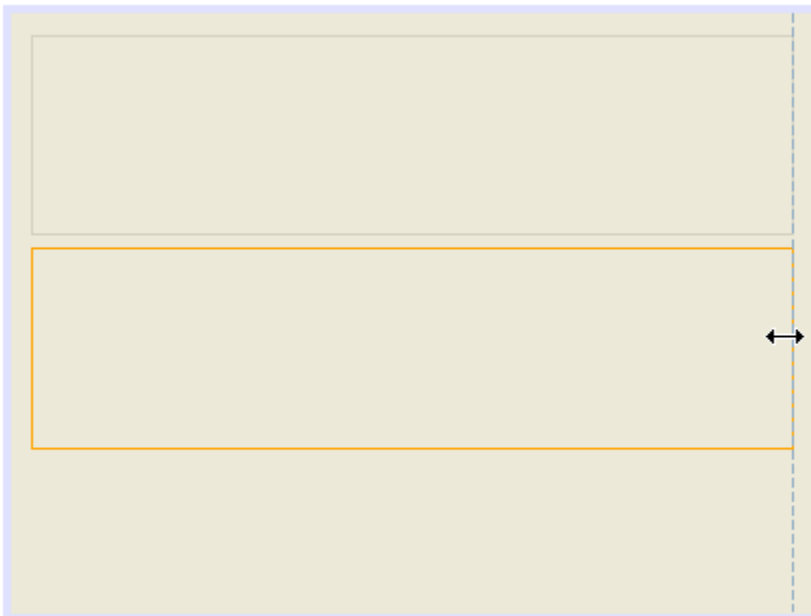
Para redimensionar o JPanel:

1. Selecione o JPanel recém-adicionado. As pequenas alças de redimensionamento quadradas reaparecerão em torno do perímetro do componente.
2. Clique e segure a alça de redimensionamento na margem direita do JPanel e arraste-a até que a linha de guia de alinhamento pontilhada seja exibida ao lado da margem do formulário.
3. Solte a alça de redimensionamento para redimensionar o componente.

O componente `JPanel` é estendido para ficar entre as margens esquerda e direita do contêiner de acordo com o deslocamento recomendado, como mostra a ilustração abaixo.




Agora que adicionamos um painel para conter a informação Nome da nossa IU, precisamos repetir o processo para adicionar outro diretamente abaixo da primeira para a informação E-mail. Consultando as ilustrações abaixo, repita as duas tarefas anteriores, prestando atenção ao posicionamento sugerido pelo Construtor de GUIs. Observe que o espaçamento vertical sugerido entre os dois JPanels é muito mais estreito que entre as margens. Quando você tiver adicionado o segundo JPanel, redimensione-o para que preencha o espaço vertical restante do formulário.



Como queremos distinguir visualmente as funções nas seções superior e inferior da nossa GUI, precisamos adicionar uma borda e título a cada JPanel. Primeiro faremos isso usando a janela Propriedades e depois a experimentaremos usando o menu pop-up.

Para adicionar bordas de título aos JPanels:

1. Selecione o JPanel superior no Construtor de GUIs.
2. Na janela Propriedades, clique no botão de reticências (...) ao lado da propriedade Borda.
3. No editor de bordas do JPanel exibido, selecione o nó TitledBorder no painel Bordas disponíveis.
4. No painel Propriedades abaixo, insira `Name` para a propriedade Título.
5. Clique nas reticências (...) ao lado da propriedade Fonte, selecione Negrito em Estilo de fonte e insira 12 em Tamanho. Clique em OK para fechar as caixas de diálogo.
6. Selecione o JPanel inferior e repita as etapas de 2 a 5, mas dessa vez clique com o botão direito do mouse no JPanel e acesse a janela Propriedades usando o menu pop-up. Insira `E-mail` para a propriedade Título.
As bordas com título serão adicionadas a ambos componentes JPanel.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#)

Adicionando componentes individuais ao formulário

Agora precisamos começar a adição dos componentes que apresentarão as informações de contato reais na nossa lista de contatos. Nessa tarefa, adicionaremos quatro JTextFields que exibirão as informações de contato e os JLabels que as descreverão. Enquanto faz isso, observe as linhas de guia horizontais e verticais que o Construtor de GUIs exibe, sugerindo o espaçamento preferido dos componentes conforme definido pela aparência do sistema operacional. Isso assegura que nossa GUI seja processada automaticamente respeitando a aparência do sistema operacional de destino em tempo de execução.

Para adicionar um JLabel ao formulário:

1. Na janela Paleta, selecione o componente Rótulo na categoria Swing.
2. Mova o cursor acima do JPanel `Name` adicionado anteriormente. Quando as linhas de guia aparecerem, indicando que o JLabel está posicionado no canto superior esquerdo do JPanel com pequena margem nos cantos superior e esquerdo, clique para posicionar o rótulo.

O JLabel será adicionado ao formulário e um nó correspondente, representando o componente, será adicionado à janela Inspetor.

Antes de prosseguir, precisamos editar o texto de exibição do JLabel que acabamos de adicionar. Embora seja possível editar o texto de exibição do componente em qualquer ponto, a maneira mais fácil é fazer isso à medida que o adiciona.

Para editar o texto de exibição de um JLabel:

1. Clique duas vezes no JLabel para selecionar a exibição de texto.
2. Digite `Nome :` e pressione Enter.

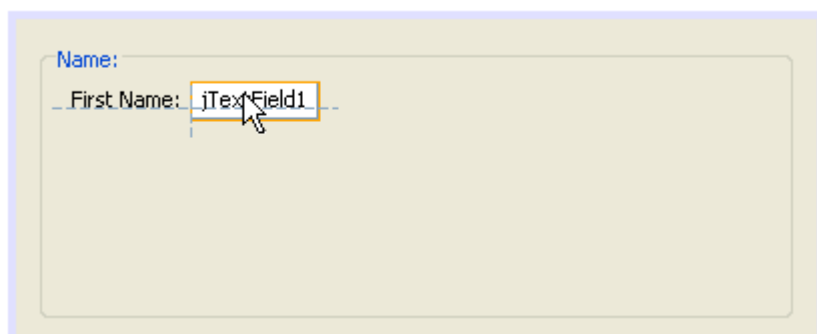
O novo nome do JLabel será exibido e a largura do componente será ajustada como resultado da edição.

Agora adicionaremos um JTextField para que possamos ter uma vista rápida do recurso de alinhamento da linha de base do Construtor de GUIs.

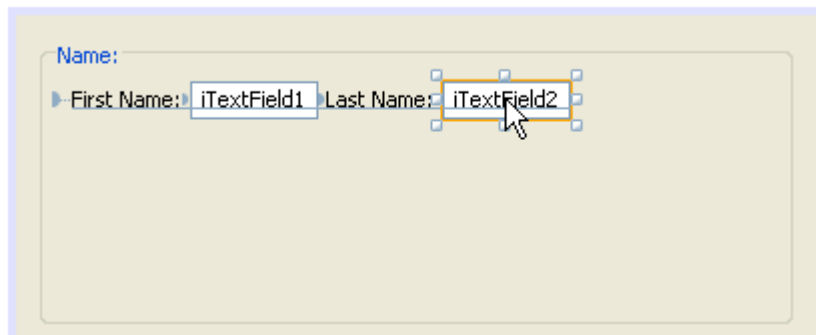
Para adicionar um JTextField ao formulário:

1. Na janela Paleta, selecione o componente Campo de texto na categoria Swing.
2. Mova o cursor imediatamente à direita do JLabel `Nome :` recém-adicionado. Quando a linha de guia horizontal aparecer, indicando que a linha de base do JTextField está alinhada com aquela do JLabel e o espaçamento entre os dois componentes for sugerido com uma linha de guia vertical, clique para posicionar o JTextField.

O JTextField se encaixará na posição no formulário alinhado com a linha de base do JLabel, conforme mostra a ilustração abaixo. Observe que o JLabel mudou ligeiramente para baixo para se alinhar à linha de base do campo de texto mais alto. Como sempre, um nó, representando o componente, será adicionado à janela Inspetor.



Antes de prosseguir, precisamos adicionar mais um JLabel e JTextField imediatamente à direita desses já adicionados, conforme mostra a ilustração abaixo. Agora insira `Sobrenome :` como texto de exibição do JLabel e mantenha, por enquanto, o texto do espaço reservado dos JTextFields como está.




Para redimensionar um JTextField:

1. Selecione o JTextField que nós acabamos de adicionar à direita do JLabel `Sobrenome`.
2. Arraste a alça de redimensionamento direita do JTextField em direção à margem direita do JPanel de fechamento.
3. Quando as linhas de guia de alinhamento vertical aparecerem sugerindo a margem entre o campo de texto e a margem direita do JPanel, solte o botão do mouse para redimensionar o JTextField.

A margem direita do JTextField se encaixará em alinhamento à margem recomendada do JPanel, conforme mostra a ilustração abaixo.



Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#)

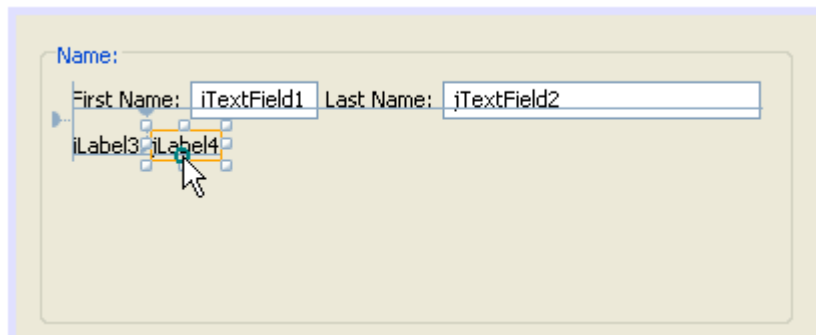
Adicionando vários componentes ao formulário

Agora adicionaremos os JLabels `Título` e `Apelido` que descrevem dois JTextFields que adicionaremos em um minuto. Arrastaremos e soltaremos os componentes enquanto pressionamos a tecla Shift para adicioná-los rapidamente ao formulário. Enquanto fazemos isso, observe novamente que o Construtor de GUIs exibe linhas de guia horizontais e verticais sugerindo o espaçamento preferido do componente.

Para adicionar vários JLabels ao formulário:

1. Na janela Paleta, selecione o componente Rótulo na categoria Swing, clicando e soltando o botão do mouse.
2. Mova o cursor sobre o formulário diretamente abaixo do JLabel `Nome` adicionado anteriormente. Quando as linhas de guia aparecerem, indicando que a nova borda esquerda do JLabel está alinhada à do JLabel acima e houver uma pequena margem entre ambos, clique pressionando a tecla Shift para posicionar o primeiro JLabel.
3. Ainda pressionando a tecla Shift, clique para colocar outro JLabel imediatamente à direita do primeiro. Certifique-se de soltar a tecla Shift antes de posicionar o segundo JLabel. Se você se esquecer de soltar a tecla Shift antes de posicionar o último JLabel, basta pressionar a tecla Escape.

Os JLabels serão adicionados ao formulário criando uma segunda linha, como mostra a ilustração abaixo. Os nós que representam os componentes serão adicionados à janela Inspetor.




Antes de continuar, precisamos editar o nome do JLabels para que possamos ver o efeito dos alinhamentos que definiremos posteriormente.

Para editar o texto de exibição de JLabels:

1. Clique duas vezes no primeiro JLabel para selecionar a exibição de texto.
2. Digite `Título:` e pressione Enter.
3. Repita as etapas 1 e 2, inserindo `Apelido:` para a propriedade nome do segundo JLabel.

Os novos nomes dos JLabels serão exibidos no formulário e mudarão como resultado das larguras editadas, como mostra a ilustração abaixo.



Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#).

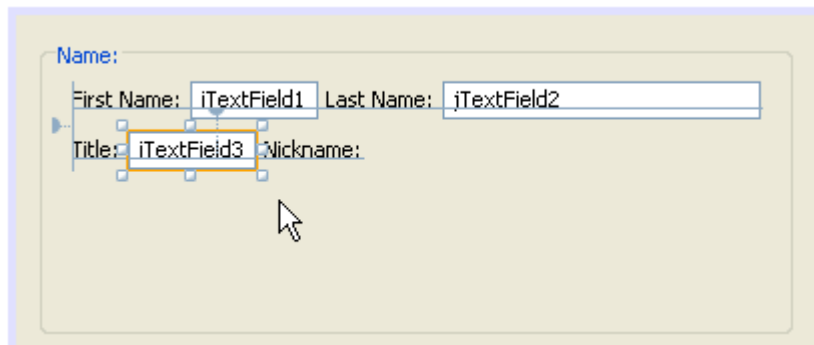
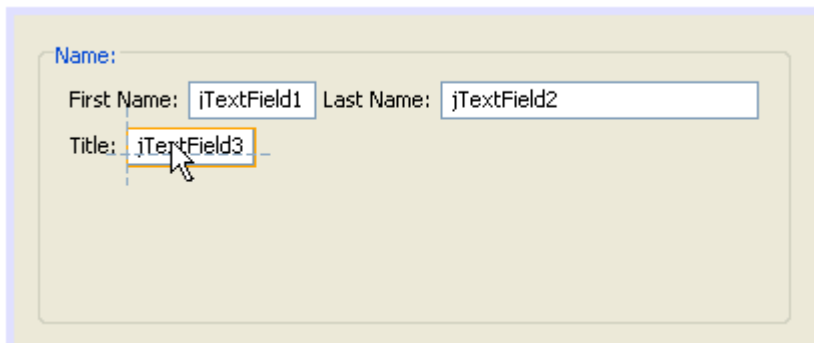
Inserindo componentes

Geralmente, é necessário adicionar um componente entre componentes já colocados em um formulário. Quando você adicionar um componente entre dois componentes existentes, o Construtor de GUIs automaticamente os mudará para dar espaço para o novo componente. Para demonstrar isso, inseriremos um JTextField entre os JLabels adicionados anteriormente, como mostram as duas ilustrações abaixo.

Para inserir um JTextField entre dois JLabels:

1. Na janela Paleta, selecione o componente Campo de texto na categoria Swing.
2. Mova o cursor acima dos JLabels `Título:` e `Apelido:` na segunda linha de maneira que o JTextField sobreponha ambos e fique alinhado às suas linhas de base. Se você tiver dificuldade em posicionar o novo campo de texto, poderá encaixá-lo à linha de guia esquerda do JLabel `Apelido:`, como mostra a primeira imagem abaixo.
3. Clique para posicionar o JTextField entre os JLabels `Título:` e `Apelido:`.

O JTextField se encaixará entre os dois JLabels. O JLabel da extremidade direita muda em direção à direita do JTextField para acomodar o deslocamento horizontal sugerido.



Ainda precisamos adicionar outro JTextField ao formulário que exibirá o apelido do contato no lado direito do formulário.

Para adicionar um JTextField:


1. Na janela Paleta, selecione o componente Campo de texto na categoria Swing.
2. Mova o cursor para a direita do rótulo `Apelido` e clique para posicionar o campo de texto.

O JTextField se encaixará no lado esquerdo do JLabel.

Para redimensionar um JTextField:

1. Arraste as alças de redimensionamento do JTextField do rótulo `Apelido`: adicionado na tarefa anterior à direita do JPanel de fechamento.
2. Quando as linhas de guia de alinhamento vertical aparecerem sugerindo a margem entre o campo de texto e as margens do JPanel, solte o botão do mouse para redimensionar o JTextField.

A margem direita do JTextField se encaixará em alinhamento com a margem recomendada do JPanel e o Construtor de GUIs deduzirá o comportamento de redimensionamento correto.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#)

[início](#)


Avançar

Alinhamento é um dos aspectos mais fundamentais da criação de GUIs com aparência profissional. Na seção anterior, tivemos uma vista rápida dos recursos de alinhamento do IDE durante a adição dos componentes JLabel e JTextField ao nosso formulário ContactEditorUI. Depois, veremos de maneira mais profunda os recursos de alinhamento do Construtor de GUIs, conforme trabalhamos com os outros componentes necessários para a nossa aplicação.

Alinhamento de componentes

Sempre que você adiciona um componente a um formulário, o Construtor de GUIs os alinha efetivamente, conforme evidenciado pelas linhas de guia de alinhamento exibidas. Entretanto, às vezes, também é necessário especificar diferentes relacionamentos entre os grupos de componentes. Anteriormente, nós adicionamos quatro JLabels necessários para a nossa GUI ContactEditor, mas nós não os alhamos. Agora alinharemos as duas colunas de JLabels para que as margens direitas se alinhem.

Para alinhar componentes:

1. Selecione os JLabels `Nome:` e `Título:` no lado esquerdo do formulário.
2. Clique no botão Alinhar à direita na coluna () na barra de ferramentas. Como alternativa, clique com o botão direito do mouse em um deles e escolha Alinhar > À direita na coluna no menu pop-up.

3. Repita isso também para os JLabels `Sobrenome:` e `Apelido:`.

A posição dos JLabels muda para que as margens direitas do texto de exibição sejam alinhadas. Os relacionamentos de ancoragem serão atualizados, indicando que os componentes foram agrupados.

Antes de terminarmos com os JTextFields adicionados anteriormente, precisamos nos certificar de que os dois JTextFields que inserimos entre os JLabels estejam definidos para se redimensionarem corretamente. Ao contrário dos dois JTextFields que foram esticados para a margem direita do formulário, o comportamento de redimensionamento dos componentes inseridos não é definido automaticamente.

Para definir o comportamento de redimensionamento de um componente:

1. Clique pressionando a tecla Control nos dois componentes JTextField inseridos para selecioná-los no Construtor de GUIs.
2. Com os dois JTextFields selecionados, clique com o botão direito do mouse em um deles e escolha Redimensionamento automático > Horizontal no menu pop-up.

Os JTextFields serão configurados com redimensionamento horizontal em tempo de execução. As linhas de guia de alinhamento e os indicadores de ancoragem serão atualizados, fornecendo comentário visual dos relacionamentos dos componentes.

Para configurar os componentes para que tenham o mesmo tamanho:

1. Clique pressionando a tecla Control nos quatro JTextFields do formulário para selecioná-los.
2. Com os JTextFields selecionados, clique com o botão direito do mouse em um deles e escolha Definir tamanho padrão no menu pop-up.

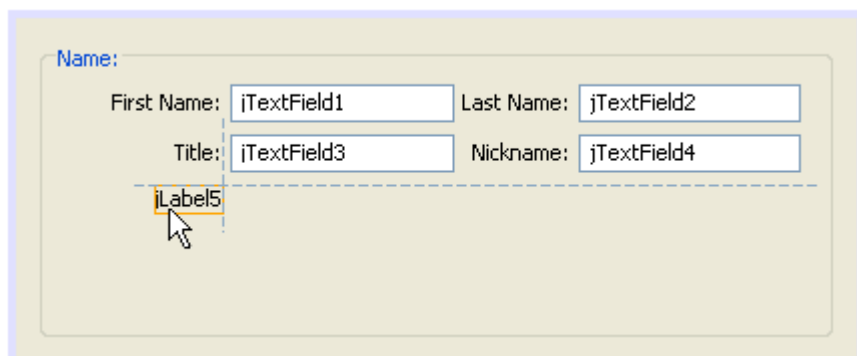
Todos os JTextFields terão a mesma largura definida e os indicadores serão adicionados na parte superior destes, fornecendo comentário visual dos relacionamentos dos componentes.

Agora precisamos adicionar outro JLabel que descreva a JComboBox que permitirá aos usuários selecionar o formato das informações que nossa aplicação ContactEditor exibirá.


Para alinhar um JLabel a um grupo de componentes:

1. Na janela Paleta, selecione o componente Rótulo na categoria Swing.
2. Mova o cursor abaixo dos JLabels `Nome:` e `Título:` no lado esquerdo do JPanel. Quando as linhas de guia aparecerem, indicando que a nova borda direita do JLabel está alinhada às margens direitas do grupo de componentes acima (os dois JLabels), clique para posicionar o componente.

O JLabel se encaixará em uma posição alinhada à direita com a coluna dos JLabels acima, conforme mostra a ilustração abaixo. O Construtor de GUIs atualiza as linhas de status de alinhamento, indicando as relações de espaçamento e ancoragem do componente.



Como nos exemplos anteriores, clique duas vezes no JLabel para selecionar o texto de exibição e insira `Exibir formato:` para nome de exibição. Observe que quando o JLabel se encaixa na posição, os outros componentes mudam para acomodar o texto de exibição maior.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#)

Alinhamento da linha de base

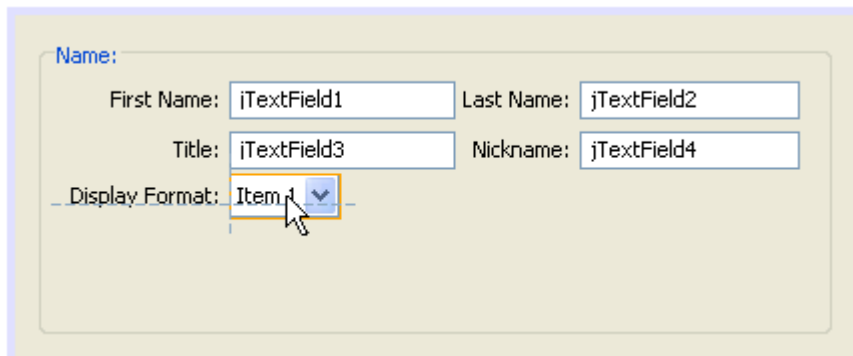
Quando você adiciona ou move componentes que incluem texto (JLabels, JTextFields, etc.), o IDE sugere alinhamentos baseados nas linhas de base do texto nos componentes. Quando inserimos o JTextField anteriormente, por exemplo, sua linha de base foi automaticamente alinhada aos JLabels adjacentes.

Agora adicionaremos a caixa de combinação que permitirá aos usuários selecionar o formato das informações que nossa aplicação ContactEditor exibirá. À medida que adicionamos JComboBox, alinharemos a linha de base com o texto do JLabel. Observe mais uma vez as linhas de guia de alinhamento da linha de base exibidas para nos auxiliar com o posicionamento.

Para alinhar as linhas de base dos componentes:

1. Na janela Paleta, selecione o componente Caixa de combinação na categoria Swing.
2. Mova o cursor à direita do JLabel recém-adicionado. Quando a linha de guia horizontal aparecer, indicando que a linha de base do JComboBox está alinhada à linha de base do texto do JLabel e o espaçamento entre os dois componentes é sugerido com uma linha de guia vertical, clique para posicionar a caixa de combinação.

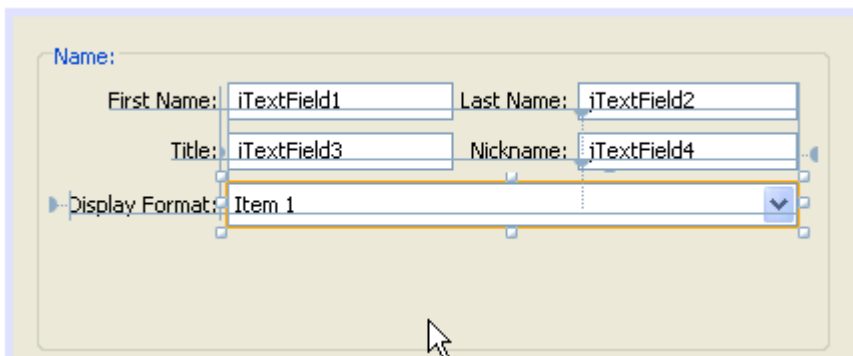
O componente se encaixará em uma posição alinhada com a linha de base do JLabel à esquerda, conforme mostra a ilustração abaixo. O Construtor de GUIs exibe as linhas de status, indicando as relações de espaçamento e ancoragem do componente.




Para redimensionar o JComboBox:

1. Selecione a Caixa de combinação no Construtor de GUIs.
2. Arraste a alça de redimensionamento na margem direita do JComboBox até que as linhas de guia de alinhamento sejam exibidas sugerindo o deslocamento preferido entre as margens do JComboBox e do JPanel.

Conforme mostra a ilustração abaixo, a margem direita do JComboBox se encaixará em alinhamento com a margem recomendada do JPanel e a largura do componente será automaticamente definida para se redimensionar ao formulário.



A edição de modelos de componentes está além do escopo desse tutorial, portanto deixaremos, por enquanto, a lista de itens de espaço reservado da JComboBox como está.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#)

[início](#)

Revisando que foi aprendido

Saímos de um bom início de construção da nossa GUI ContactEditor, mas tomemos um minuto para recapitular o que aprendemos enquanto adicionamos mais alguns componentes que nossa interface requer.

Até agora, nós nos concentramos na adição de componentes à nossa GUI ContactEditor usando as linhas de guia de alinhamento do IDE para nos ajudar no posicionamento. Entretanto, é importante compreender que a ancoragem é outra parte integrante do posicionamento de componentes. Embora ainda não tenhamos discutido isso, você já se beneficiou com esse recurso sem perceber. Como mencionado antes, quando você adiciona um componente a um formulário, o IDE sugere o posicionamento preferido da aparência de destino com linhas de guia. Depois de posicionados, os novos componentes também são ancorados na margem mais próxima do contêiner ou componente para assegurar que os relacionamentos entre os componentes sejam mantidos em tempo de execução. Nesta seção, nós nos concentraremos em concluir as tarefas de maneira mais otimizada enquanto ressaltamos o trabalho que o Construtor de GUIs está fazendo nos bastidores.

Adicionando, alinhando e ancorando

O Construtor de GUIs permite fazer o layout de formulários rápida e facilmente otimizando gestos típicos do fluxo de trabalho. Quando você adiciona um componente a um formulário, o Construtor de GUIs o encaixa automaticamente na posição preferida e define as relações de encadeamento necessárias para que você se concentre no design dos formulários em vez de brigar com detalhes de implementação complicados.

Para adicionar, alinhar e editar o texto de exibição de um JLabel:

1. Na janela Paleta, selecione o componente Rótulo na categoria Swing.
2. Mova o cursor sobre o formulário imediatamente abaixo do título E-mail do JPanel inferior. Quando as linhas de guia aparecerem, indicando que está posicionado no canto superior esquerdo do JPanel com uma pequena margem nos cantos superior e esquerdo, clique para posicionar o JLabel.
3. Clique duas vezes no JLabel para selecionar a exibição de texto. Depois digite `Endereço de e-mail:` e pressione Enter.

O JLabel se encaixará na posição preferida no formulário e será ancorado nas margens superior e esquerda do JPanel de fechamento. Assim como ocorreu anteriormente, um nó correspondente, representando o componente, também será adicionado à janela Inspetor.

Para adicionar um JTextField:

1. Na janela Paleta, selecione o componente Campo de texto na categoria Swing.
2. Mova o cursor imediatamente à direita do rótulo `Endereço de e-mail` recém-adicionado. Quando as linhas de guia aparecerem, indicando que a linha de base do JTextField está alinhada à linha de base do texto no JLabel e a margem entre os dois componentes for sugerida com uma linha de guia vertical, clique para posicionar o campo de texto.

O JTextField se encaixará à direita do JLabel `Endereço de e-mail:` e será encadeado ao JLabel. O nó correspondente também será adicionado à janela Inspetor.

3. Arraste a alça de redimensionamento do JTextField em direção à direita do JPanel até que as linhas de guia de alinhamento sejam exibidas sugerindo o deslocamento entre as margens do JTextField e do JPanel.

A margem direita do JTextField se encaixará na linha de guia de alinhamento indicando as margens preferidas.


Agora precisamos adicionar o JList que exibirá a lista de contatos completa do ContactEditor.

Para adicionar e redimensionar um JList:

1. Na janela Paleta, selecione o componente Lista na categoria Swing.
2. Mova o cursor imediatamente abaixo do JLabel `Endereço de e-mail` adicionado anteriormente. Quando as linhas de guia aparecerem, indicando que as margens superior e esquerda do JList estão alinhadas às margens preferidas ao longo da margem esquerda do JPanel e do JLabel acima, clique para posicionar o JList.
3. Arraste a alça de redimensionamento do JList em direção à direita do JPanel de fechamento até que as linhas de guia de alinhamento sejam exibidas indicando que tem a mesma largura do JTextField acima.

O JList se encaixará na posição designada pelas linhas de guia de alinhamento e seu nó correspondente será exibido na janela Inspetor. Observe também que o formulário se expande para acomodar o JList recém-adicionado.

Como as JLists são usadas para exibir longas listas de dados, geralmente requerem a adição de um JScrollPane. Quando você adiciona um componente que requer um JScrollPane, o Construtor de GUIs o adiciona automaticamente para você. Como os JScrollPane são componentes não visuais, você precisa usar a janela Inspetor para exibir ou editar os JScrollPane que o Construtor de GUIs criou.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#)

Dimensionamento de componentes

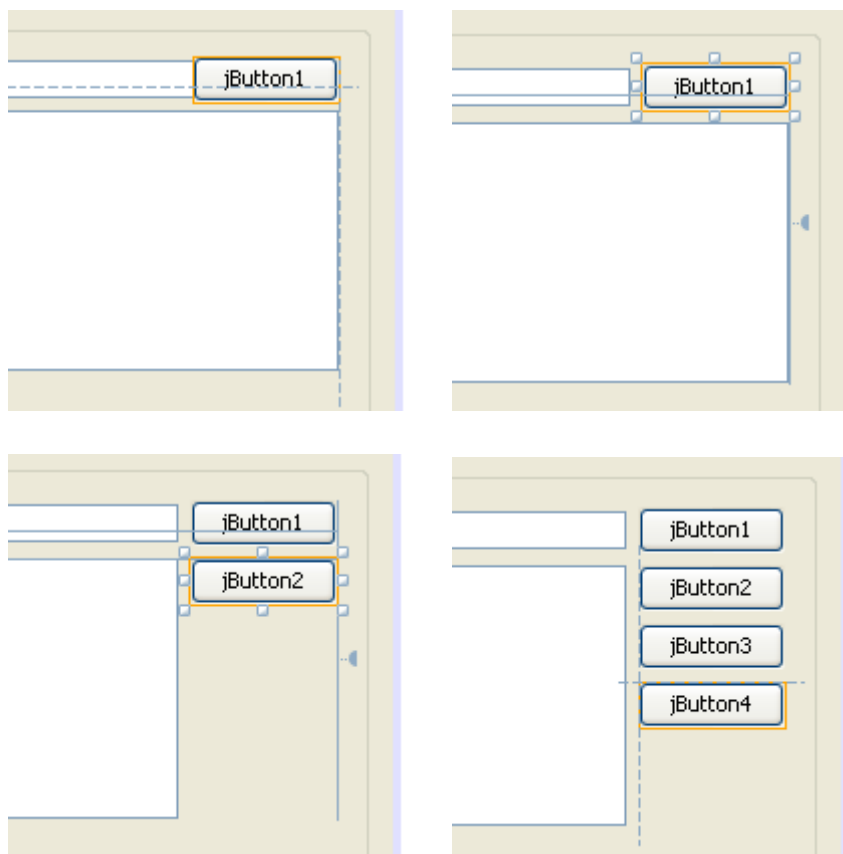
É sempre benéfico definir vários componentes relacionados, como botões em caixas de diálogo modais, para que tenham o mesmo tamanho a fim de oferecer consistência visual. Para demonstrar isso, adicionaremos quatro JButtons ao nosso formulário ContactEditor que permitirão adicionar, editar e remover entradas individuais da nossa lista de contatos, como mostram as ilustrações abaixo. Depois, definiremos os quatro botões para que tenham o mesmo tamanho e possam ser facilmente reconhecidos e ofereçam a funcionalidade relacionada.

Para adicionar, alinhar e editar o texto de exibição de vários botões:

1. Na janela Paleta, selecione o componente Botão.
2. Mova o JButton acima da margem direita do JTextField `Endereço de e-mail` no JPanel inferior. Quando as linhas de guia aparecerem, indicando que a linha de base e a margem direita do JButton estão alinhadas às margens do JTextField, clique pressionando Shift para posicionar o primeiro botão ao longo da margem direita do JFrame. A largura do JTextField diminui para acomodar o JButton onde você soltar o botão do mouse.

- Mova o cursor sobre o canto superior direito do JList no JPanel inferior. Quando as linhas de guia aparecerem, indicando que as margens superior e direita do JButton estão alinhadas às margens do JList, clique pressionando Shift para posicionar o segundo botão ao longo da margem direita do JFrame.
- Adicione dois outros JButtons abaixo dos dois já adicionados para criar uma coluna. Certifique-se de posicionar os JButtons de maneira que o espaçamento sugerido seja respeitado e consistente. Se você se esquecer de soltar a tecla Shift antes de posicionar o último JButton, basta pressionar a tecla Escape.
- Defina o texto de exibição de cada JButton. (Você pode editar o texto de um botão, clicando com o botão direito do mouse no botão e escolhendo Editar texto. Ou você pode clicar no botão, pausar e clicar novamente.) Insira `Adicionar` para o botão superior, `Editar` para o segundo, `Remover` para o terceiro e `Padrão` para o quarto.


Os componentes JButton se encaixam nas posições designadas pelas linhas de guia de alinhamento. A largura dos botões muda para acomodar os novos nomes.



Agora que os botões estão posicionados onde os queremos, nós definiremos os quatro botões para que tenham o mesmo tamanho e consistência visual, bem como para esclarecer a funcionalidade relacionada.

Para definir componente do mesmo tamanho:

- Selecione os quatro JButtons pressionando a tecla Control enquanto faz a seleção.
- Clique com o botão direito do mouse em um deles e escolha `Mesmo tamanho > Mesma largura` no menu pop-up. Os JButtons serão definidos com o mesmo tamanho do botão que tiver o nome mais longo.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#).

Recuo

Freqüentemente é necessário unir vários componentes em outro para que fique claro que pertencem a um grupo de funções relacionadas. Um caso típico, por exemplo, é colocar várias caixas de verificação relacionadas abaixo de um rótulo comum. O Construtor de GUIs permite fazer o recuo facilmente oferecendo linhas de guia especiais que sugerem o deslocamento preferido para a aparência do seu sistema operacional.

Nesta seção, adicionaremos alguns JRadioButtons abaixo de um JLabel que permitirá aos usuários personalizar a maneira como a aplicação exibe dados. Veja as ilustrações abaixo enquanto faz isso ou clique no link [Visualizar demonstração](#) seguindo o procedimento para ver uma demonstração interativa.

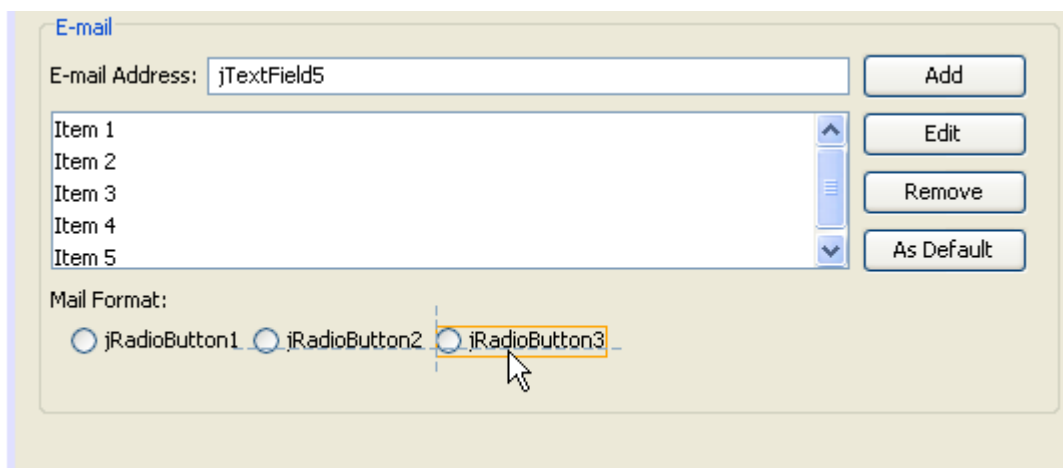
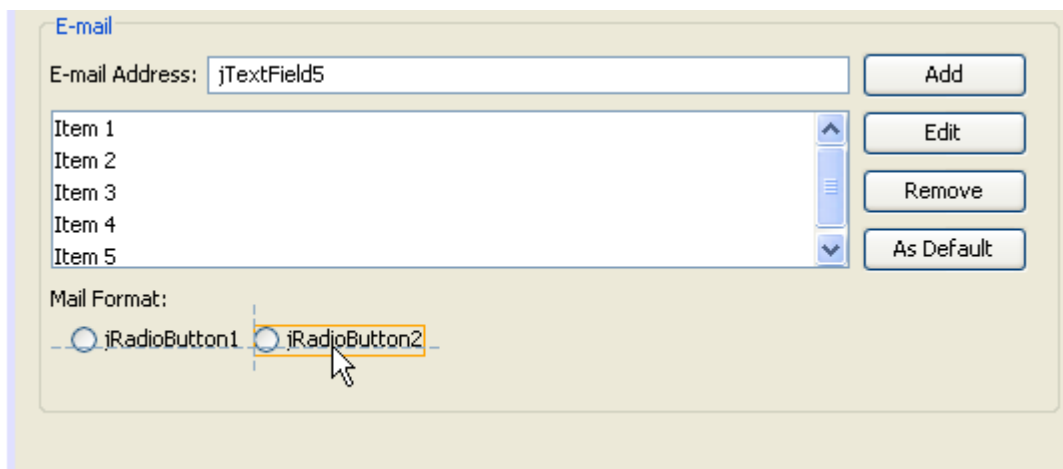
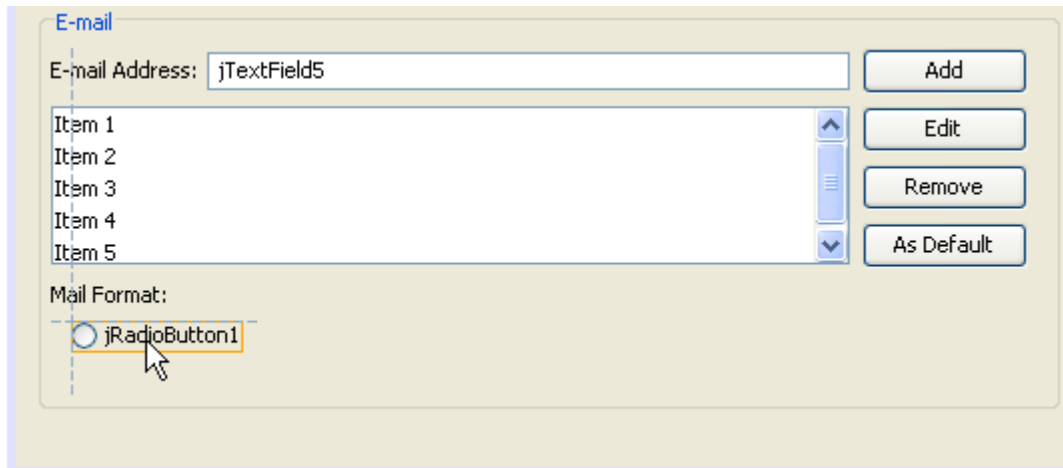
Para recuar JRadioButtons abaixo de um JLabel:

- Adicione um JLabel denominado `Formato de correspondência` ao formulário abaixo do JList.

Certifique-se de que o rótulo está alinhado à esquerda do JList acima.

- Na janela Paleta, selecione o componente Botão de opção na categoria Swing.
- Mova o cursor abaixo do JLabel recém-adicionado. Quando as linhas de guia aparecerem, indicando que a borda esquerda do JButton está alinhada com a do JLabel, mova o JButton ligeiramente para a direita até as linhas de guia de recuo secundárias aparecerem. Clique pressionando a tecla Shift para posicionar o primeiro botão de opção.
- Mova o cursor à direita do primeiro JButton. Clique pressionando a tecla Shift para inserir o segundo e o terceiro JButtons, respeitando o espaçamento sugerido do componente. Certifique-se de soltar a tecla Shift antes de posicionar o último JButton.
- Defina o texto de exibição de cada JButton. (Você pode editar o texto de um botão, clicando com o botão direito do mouse no botão e escolhendo Editar texto. Ou você pode clicar no botão, pausar e clicar novamente.) Insira HTML para o botão de opção esquerdo, Texto simples para o segundo e Personalizado para o terceiro.

Três JButtons serão adicionados ao formulário e recuados abaixo do JLabel Formato de correspondência.




Agora, precisamos adicionar os três JButtons a um ButtonGroup para permitir o comportamento de alternância

esperado no qual somente um botão de opção pode ser selecionado por vez. Por sua vez, isso assegurará que as informações de contato do aplicativo ContactEditor serão exibidas no formato de email da nossa escolha.

Para adicionar JRadioButtons a um ButtonGroup:

1. Na janela Paleta, selecione o componente Grupo de botões na categoria Swing.
2. Clique em qualquer lugar na área de design do Construtor de GUIs para adicionar o componente ButtonGroup ao formulário. Observe que o Grupo de botões em si não é exibido no formulário, no entanto, é visível na área Outros componentes do Inspetor.
3. Selecione os três JRadioButtons do formulário.
4. Na janela Propriedades, selecione buttonGroup1 na caixa de combinação da propriedade buttonGroup. Três JRadioButtons serão adicionados ao grupo de botões.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#)

[início](#)

Fazendo os ajustes finais

Conseguimos esboçar nossa GUI do aplicativo ContactEditor, mas ainda há algumas coisas a serem feitas. Nesta seção, observaremos algumas das tarefas de layout típicas que o Construtor de GUIs simplifica.


Finalizando

Agora precisamos adicionar os botões que permitirão aos usuários confirmar as informações que inserirem de um contato individual e adicioná-las à lista de contatos ou cancelar, deixando o banco de dados inalterado. Nesta etapa, adicionaremos os dois botões requeridos e os editaremos para que sejam exibidos com o mesmo tamanho no formulário, embora seus textos de exibição tenham tamanhos diferentes.

Para adicionar e editar o texto de exibição dos botões.


1. Se o JPanel inferior estiver estendido na margem inferior do formulário JFrame, arraste a margem inferior do JFrame para baixo. Isso proporciona espaço entre as margens do JFrame e do JPanel para os botões OK e Cancelar.
2. Na janela Paleta, selecione o componente Botão na categoria Swing.
3. Mova o cursor sobre o formulário abaixo do JPanel E-mail. Quando as linhas de guia aparecerem, indicando que a borda direita do JButton está alinhada ao canto direito do JFrame, clique para posicionar o botão.
4. Adicione outro JButton à esquerda do primeiro, certificando-se de colocá-lo usando o espaçamento sugerido ao longo da margem inferior do JFrame.
5. Defina o texto de exibição de cada JButton. Insira `OK` para o botão esquerdo e `Cancelar` para o botão direito. Observe que a largura dos botões muda para acomodar os novos nomes.
6. Defina os dois JBButtons para terem o mesmo tamanho selecionando-os, clicando com o botão direito do mouse em um deles e selecionando `Mesmo tamanho > Mesma largura` no menu pop-up. Os componentes JButton serão exibidos no formulário e seus nós correspondentes serão exibidos na janela Inspetor. O código dos componentes JButton também será adicionado ao arquivo de origem do formulário, que é visível na exibição Fonte do editor. Todos os JBButtons serão definidos com o mesmo tamanho definido do botão que tiver o nome mais longo.

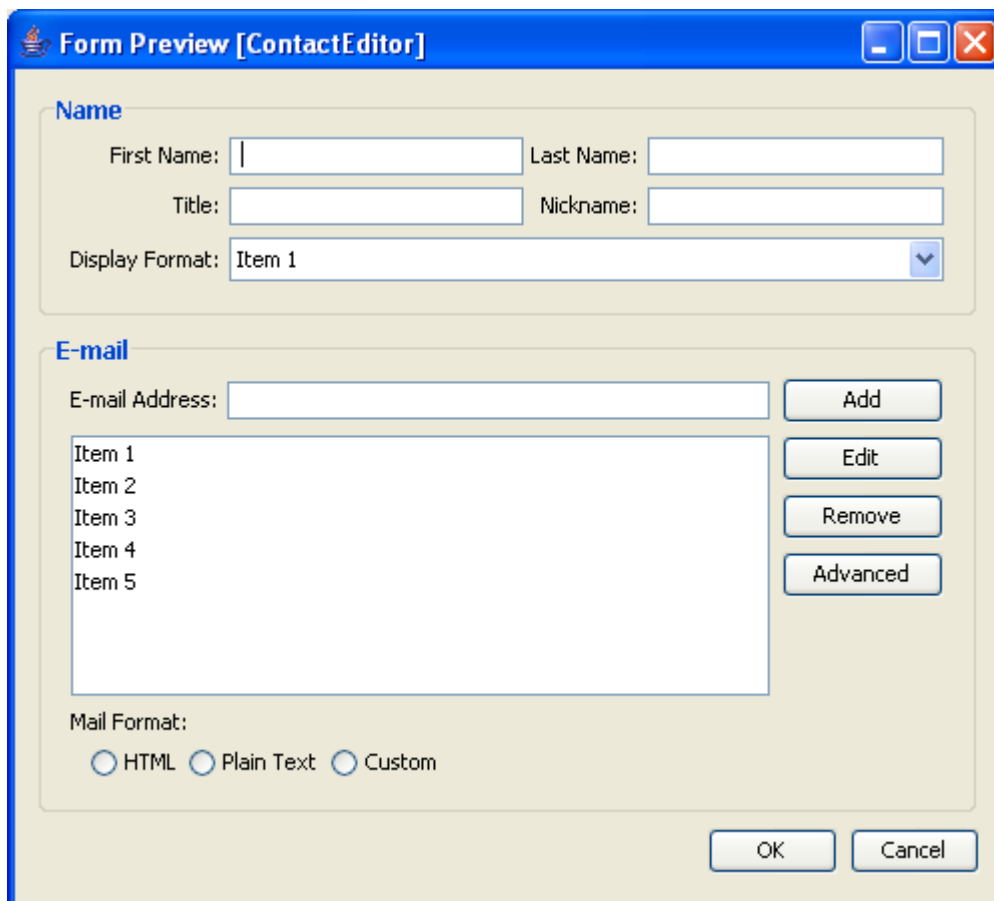
A última tarefa a ser realizada é excluir o texto do espaço reservado nos vários componentes. Observe que, embora a remoção do texto do espaço reservado após a configuração preliminar de um formulário possa ser uma técnica útil para evitar problemas de alinhamento e relações de ancoragem dos componentes, a maioria dos desenvolvedores geralmente remove esse texto no processo de layout de seus formulários. À medida que percorre o formulário, selecione e exclua o texto do espaço reservado de cada um dos JTextFields. Deixaremos os itens de espaço reservado no JComboBox e no JList para um tutorial posterior.

Para ver uma demonstração do procedimento anterior, clique em  [Exibir demonstração](#).

[início](#)

Visualizando a GUI

Agora que você criou com êxito a GUI ContactEditor, você pode experimentar a interface para ver os resultados. É possível visualizar o formulário à medida que trabalha clicando no botão Visualizar formulário () na barra de ferramentas do Construtor de GUIs. O formulário será aberto em sua própria janela, permitindo que você o teste antes da construção e execução.



[início](#)

Implantando aplicações de GUI

Para que as interfaces criadas com o Construtor de GUIs funcionem fora do IDE, a aplicação deve ser compilada nas classes do gerenciador de layout GroupLayout e essas classes também devem estar disponíveis no tempo de execução. Essas classes são incluídas no Java SE 6, mas não no Java SE 5. Se você desenvolver o aplicativo para ser executado no Java SE 5, seu aplicativo precisa usar a biblioteca de extensões de layout Swing.

Se o IDE estiver sendo executado no JDK 5, o IDE gerará o código da aplicação automaticamente para usar a biblioteca de extensões de layout Swing. Ao implantar a aplicação, você precisa incluir a biblioteca de extensões de layout Swing. Quando você constrói o aplicativo (Construir > Construir projeto principal), o IDE fornece automaticamente uma cópia do arquivo JAR da biblioteca na pasta `dist/lib` do aplicativo. O IDE também adiciona todos os arquivos JAR da pasta `dist/lib` no elemento `Class-Path` no arquivo `manifest.mf` do arquivo JAR da aplicação.

Se você estiver executando o IDE no JDK 6, o IDE gera o código do seu aplicativo para usar as classes GroupLayout que estão no Java SE 6. Isso significa que você pode implantar o aplicativo para ser executado em sistemas com o Java SE 6 instalado e não é necessário empacotar seu aplicativo com a biblioteca de extensões de layout Swing.

Observação: se a aplicação for criada com o JDK 6, mas você precisar que também seja executada no Java SE 5, poderá fazer com que o IDE gere seu código para usar a biblioteca de extensões de layout Swing em vez das classes no Java SE 6. Abra a classe ContactEditorUI no Editor de GUIs. No Inspetor, expanda o nó ContactEditorUI e selecione Formulário ContactEditorUI. Na janela Propriedades, altere o valor da propriedade Estilo de geração de layout para Biblioteca de extensões de layout Swing.

Distribuindo e executando aplicações de GUI independentes

Para preparar a aplicação de GUI para distribuição fora do IDE:

- Compacte a pasta `dist` do projeto em um arquivo ZIP. (A pasta `dist` também pode conter uma pasta `lib`, que também poderia necessitar ser incluída).

Para executar uma aplicação de GUI a partir da linha de comando:

1. Navegue até a pasta do `dist` projeto.
2. Digite o seguinte:

```
java -jar <nome do jar>.jar
```

Observação: se você encontrar este erro:

```
Exception in thread "main" java.lang.NoClassDefFoundError: org/jdesktop/layout/GroupLayout$Group
```


Verifique se o arquivo manifest.mf faz referência à versão atualmente instalada da biblioteca de extensões de layout Swing.

Para obter mais informações sobre a biblioteca de extensões de layout, visite o [projeto Extensões de layout Swing](#).

[início](#)

[Envie-nos seus comentários](#)

Consulte também

Agora você concluiu o tutorial Criando uma GUI Swing. Para obter informações sobre como adicionar funcionalidade às GUIs que você criar, consulte:

- [Introdução à construção de GUIs](#)
- [Criando uma aplicação de banco de dados de área de trabalho Java](#)
- [Criando um aplicativo de banco de dados de área de trabalho Java personalizado](#)
- [Perguntas freqüentes do Construtor de GUIs](#)
- [Trilha do aprendizado das aplicações de GUI em Java](#)

[início](#)