

Desenvolvendo uma aplicação em Visual Basic

Leva somente alguns minutos para construir sua primeira aplicação em Visual Basic. Você cria a interface de usuário "puxando" controles, como caixas de texto e botões de comando, em um formulário. Depois, você ajusta as propriedades para o formulário e controles para especificar cada valor como legendas, color, e tamanho. Finalmente, você escreve o código para dar vida para a aplicação. Os passos básicos que você leva criando sua primeira aplicação mostrarão para você os princípios que você usará com cada outra aplicação que você desenvolver.

Este capítulo provê uma avaliação do processo de desenvolvimento de aplicação, descreve as condições e habilidades que você precisa para usar o Visual Basic, e o leva passo a passo por várias aplicações simples.

Conceitos Visual Basic

Para entender o processo de desenvolvimento da aplicação, é útil entender alguns dos conceitos chaves em qual o Visual Básico é construído. Porque o Visual Basic é uma linguagem de desenvolvimento para Windows, um pouco de familiaridade com o ambiente Windows é necessária. Se você é novo em programação Windows, você precisa estar atento de algumas diferenças fundamentais entre programar para Windows e em outros ambientes.

Como o Windows Trabalha: Windows, Eventos e Mensagens

Uma discussão completa dos funcionamentos internos do Windows requereria um livro inteiro. Um entendimento fundo de todos os detalhes técnicos não é necessário. Uma versão simplificada dos funcionamentos do Windows envolve três conceitos chaves: janelas, eventos e mensagens. Pense em uma janela como simplesmente uma região retangular com seus próprios limites. Você provavelmente já está atento de vários tipos diferentes de janelas: uma janela do Explorer no Windows 95, uma janela de documento dentro de seu programa processador de textos (Word), ou uma caixa de diálogo instantânea que o faz lembrar de um compromisso. Enquanto estes são os exemplos mais comuns, há realmente muitos outros tipos de janelas. Um botão de comando é uma janela. Ícones, caixas de texto, botões de opção e barras de menu são todas janelas.

O sistema operacional Microsoft Windows administra todas as várias janelas nomeando para cada uma um número de identificação (window handle ou hWnd). O sistema monitora continuamente os sinais de atividade ou eventos de cada uma destas janelas. Eventos podem acontecer por ações de usuário como um click do mouse ou uma tecla pressionada, por controle de programação, ou até mesmo como resultado das ações de outra janela.

Cada vez que um evento acontece causa uma mensagem a ser enviada ao sistema operacional. O sistema processa a mensagem e transmite dados para as outras janelas. Cada janela pode entrar em ação apropriada então baseado em suas próprias instruções por lidar com aquela mensagem particular (por exemplo, se redesenhar quando foi descoberta por outra janela).

Como você poderia imaginar e poderia lidar com todos as possíveis combinações de janelas, eventos e mensagens. Afortunadamente, o Visual Básico o separa de ter que lidar com todo a manipulação de mensagem de baixo nível. Muitas das mensagens são dirigidas automaticamente pelo Visual Basic; outras estão expostas como procedimentos de Evento para sua conveniência. Isto lhe permite criar aplicações poderosas rapidamente sem ter que lidar com detalhes desnecessários.

Entendendo o Modelo Evento-dirigido

Em aplicações tradicionais ou " processuais ", a própria aplicação controla que porções de código executam e em que seqüência. A execução começa com a primeira linha de código e segue um caminho de predefinido pela aplicação e chama procedimentos como precisou.

Em uma aplicação evento-dirigido, o código não segue um caminho predeterminado - executa seções de código diferentes em resposta para eventos. Eventos podem ser ativados pelas ações do usuário, através de mensagens das aplicações de sistema ou outras, ou até mesmo da própria

aplicação. A seqüência destes eventos determina a seqüência na qual o código é executado, assim o caminho pelo código da aplicação difere em cada execução do programa. Porque você não pode prever a seqüência de eventos, seu código tem que fazer certas suposições sobre o " estado do mundo " quando executa. Quando você faz suposições (por exemplo, que um campo de entrada tem que conter um valor antes de correr um procedimento para processar aquele valor), você deveria estruturar sua aplicação de tal modo para ter certeza que a suposição sempre será válida (por exemplo, incapacitando o botão de comando que começa o procedimento até o campo de entrada contém um valor). Seu código também pode ativar eventos durante execução. Por exemplo, programavelmente mudando o texto em uma caixa de texto faria o evento de Mudança da caixa de texto acontecer. Isto causaria a execução do código (se qualquer) contido no evento de Mudança. Se você assumisse que este evento só seria ativado através de interação de usuário, você poderia ver resultados inesperados. É por isto que é importante entender o modelo evento-dirigido e se lembrar isto de quando estiver desenvolvendo sua aplicação.

Desenvolvimento interativo

O processo tradicional de desenvolvimento da aplicação pode ser quebrado em três passos distintos: escrever, compilar, e testar o código. Diferente das linguagens tradicionais, o Visual Basic usa um acesso interativo para desenvolvimento, encobrimdo a distinção entre os três passos. Com a maioria das linguagens, se você comete um erro na escrita de seu código, o erro é pego pelo compilador quando você começa a compilar sua aplicação. Você deve então achar e corrigir o erro e tem que começar o ciclo de compilação novamente e repetir o processo para cada erro encontrado. O Visual Basic interpreta seu código como você o entrou, pegando e destacando a maioria da sintaxe ou soletrando erros na mosca. É quase como ter uma vigilância especialista em cima de seu ombro como você entra seu código. Além de pegar erros na mosca, o Visual Basic também compila parcialmente o código como é entrado. Quando você está pronto a execução e testa sua aplicação, há só uma demora breve para terminar a compilação. Se o compilador acha um erro, é destacado em seu código. Você pode corrigir o erro e pode continuar compilando sem ter que recomeçar. Por causa da natureza interativa do Visual Básico, você se achará freqüentemente executando sua aplicação como você a desenvolve. Deste modo você pode testar os efeitos de seu código quando você trabalha em lugar de esperar para compilar depois.

Elementos do ambiente integrado de desenvolvimento

O ambiente de trabalho do Visual Básico é freqüentemente chamado de ambiente de desenvolvimento integrado ou IDE porque integra muitas funções diferentes como criar, editar, compilar, e depurar dentro de um ambiente comum. Em ferramentas de desenvolvimento mais tradicionais, cada uma destas funções operaria como um programa separado, cada um com sua própria interface. Nesta seção, são discutidos os tópicos seguintes:

- **Iniciando o IDE Visual Basic Abrindo e executando.**

Quando você executa o programa de Instalação do Visual Basic, lhe permite colocar os itens de programa em um grupo de programa existente ou criar um grupo de programa novo e itens de programa novos para Visual Basic em Windows. Você está então pronto iniciar o Visual Basic para Windows.

Para iniciar o Visual Basic para Windows

1. Clique em **Iniciar** na barra de Tarefas
2. Selecione **Programas**, e então **Visual Basic 5.0**.

– ou –

Clique em **Iniciar** na barra de Tarefas.

Selecione **Programas**.

Use o **Windows Explorer** para encontrar o arquivo executável Visual Basic (VB5.exe).

3. Clique duplamente no ícone Visual Basic

Você também pode criar um atalho para Visual Básico, e clicar duas vezes no atalho. Quando você inicia o Visual Básico, você vê a interface do ambiente de desenvolvimento integrado, como mostrado em Figura 2.1.

- **Elementos do Ambiente de Desenvolvimento integrados** Uma introdução para as várias partes do IDE.

O ambiente de desenvolvimento integrado Visual Basic (IDE) consiste nos seguintes elementos :

Barra de Menu

Exibe os comandos que você usa para trabalhar com Visual Basic. Além do padrão Arquivo, Editar, Visualizar, Janela, e Menu de Ajuda, são fornecidos menus para ter acesso à funções específicas para programar como: Projeto, Formatar, ou Depurar.

Menus de contexto

Contém atalhos para ações freqüentemente executadas. Para abrir um Menu de contexto, clique o botão direito do mouse no objeto que você está usando. A lista específica de atalhos disponível de Menus de contexto depende da parte do ambiente onde você clica o botão de direito do mouse. Por exemplo, o menu de contexto exibe quando você clica com o botão direito na Caixa de ferramentas lhe mostrará os Componentes caixa de diálogo, esconder Caixa de ferramentas, fixar ou não fixar a Caixa de ferramentas, ou adicionar uma aba de personalizada para a Caixa de ferramentas.

Barras de Ferramentas

Dá acesso rápido para comandos comumente usados no ambiente de programação. Você clica o botão uma vez na Barras de Ferramentas carrega a ação representada por aquele botão. Por padrão, é exibido a Barras de Ferramentas Padrão quando você inicia o Visual Basic. Barra de Ferramentas adicionais para editar, projetar o Form, e depurar podem ser colocadas ou removidas através do comando visualizar Barras de Ferramentas no menu Visualizar.

Barras de Ferramentas podem ser ancorado em baixo da barra de menu ou pode flutuar " se você seleciona a barra vertical na extremidade esquerda e arrastá-la para longe da barra de menu.

Caixa de ferramentas

Dá um conjunto de ferramentas que você usa em tempo de desenvolvimento para colocar controles em uma form. Além do esquema da caixa de ferramentas padrão, você pode criar seus próprios esquemas personalizados selecionando Adicionar Aba do menu de contexto e controles adicionais para a aba resultante.

Para Mais Informação para aprender mais sobre controles específicos, veja " Forms, Controles, e Menus " e " Usando Controles Padrões do Visual Basic ". Para informação em como adicionar controles para a Caixa de ferramentas, veja " Adicionando Controles para um Projeto " " Administrando Projetos ".

Janela Explorador de Projeto

Lista as forms e módulos em seu projeto atual. Um projeto é a coleção de arquivos que você usa para construir uma aplicação.

Para Mais Informação Para informação sobre projetos, veja " Administrando Projetos ".

Janela de propriedades

Lista as colocações de propriedade para a form selecionada ou controle. Uma propriedade é uma característica de um objeto, como tamanho, legenda, ou cor.

Para Mais Informação Para informação sobre propriedades, veja " Compreendendo Propriedades, Métodos, e Eventos " em " Forms, Controles, e Menus " .

Pesquisar Objeto (Browser Object)

Lista objetos disponíveis para uso em seu projeto e lhe dá um modo rápido para navegar por seu código. Você pode usar o Object Browser para explorar objetos em Visual Basic e outras aplicações, veja que métodos e propriedades estão disponíveis para esses objetos, e a pasta de procedimentos de código em sua aplicação.

Para Mais Informação Para mais informação sobre como usar o Object Browser para ver procedimentos, veja " Descobrimo Sobre Objetos " " Programando com Objetos " . Para detalhes em usar recursos adicionais para estender o ambiente de programação Visual Basic, veja " Usando os Assistentes e Recursos Adicionais " " Administrando Projetos " .

Projetar a Form

Serve como uma janela que você personaliza para projetar a interface de sua aplicação. Você adiciona controles, gráficos, e quadros em uma form para criar o visual que você quer. Cada form em sua aplicação tem sua própria janela de desenvolvimento de form.

Para Mais Informação aprender a adicionar controles para uma aplicação, veja " Sua Primeira Aplicação " Visual Basic depois neste capítulo. Aprender mais sobre projetar uma interface, veja " Criando uma Interface " de Usuário.

Janela Editor de Código

Serve como editor para entrar o código da aplicação. Uma janela de editor de código separada é criada para cada form ou módulo de código em sua aplicação.

Para Mais Informação aprender mais sobre entrar o código e usar o editor de código, veja " Fundamentos de Programação " .

Janela de Esquema de Form

A janela de Esquema de Form (Figura 2.2) lhe permite posicionar as forms em sua aplicação usando uma pequena representação gráfica na tela.

Figure 2.2 A janela de Esquema de Form

Imediato, Locais, e Janela de Vigilância

Estas janelas adicionais são providas para depurar sua aplicação. Elas só estarão disponíveis quando você está executando sua aplicação dentro do IDE.

Para Mais Informação aprender mais sobre depurar e usar as janelas de depuração, veja " Depurando Seu Código e Lidar com Erros " .

Nota Você também pode adicionar características para a interface Visual Basic usando um programa chamado um Add-in. Add-ins, que estão disponíveis da Microsoft e de três-grupos de desenvolvimento, pode prover características como controle de código de fonte que lhe permite apoiar projetos de desenvolvimento de grupo.

- **Opções de ambiente** Configurando o Visual Básico para suas preferências pessoais.

O Visual Basic oferece muita flexibilidade e lhe permite configurar o ambiente de trabalho para melhor adaptar o seu estilo individual. Você pode escolher entre uma única ou múltipla interface de documento, e você pode ajustar o tamanho e posicionando do vários Elementos do Ambientes de Desenvolvimento Integrado (IDE). Seu esquema persistirá entre as sessões de Visual Basic.

- **Opções de Ambiente** Configurando o Visual Basic para suas preferências pessoais.

SDI ou MDI Interface

Dois estilos diferentes estão disponíveis para o IDE Visual Basic: única interface de documento (SDI) ou interface múltipla de documento (MDI). Com a opção de SDI, todas as janelas de IDE são

livres para serem movidas em qualquer lugar da tela; enquanto o Visual Basic é a aplicação atual, eles permanecerão em cima de qualquer outra aplicação. Com a opção de MDI, todas as janelas de IDE são contidas dentro de uma única janela pai redimensionável.

Trocar entre os modos de SDI e MDI

1. Selecione Opções no menu Ferramentas.
As Caixa de Diálogo Opções é exibida.
2. selecione a aba Avançada.
3. Assinale ou não a caixa Ambiente de Desenvolvimento SDI.
O IDE começará no modo selecionado da próxima vez você iniciar o Visual Basic.
- ou -
Execute o Visual Básico da linha de comando com um parâmetro /sdi ou /mdi.

Ancorando Janelas

Muitas das janelas no IDE podem ser ancoradas, ou conectadas, para uma outra ou para a extremidade da tela. Estas incluem a Caixa de ferramentas, Janela de Esquema de Form, Explorador de Projeto, janela de Propriedades, Paleta de Cor, e Imediato, Locais, e janela de vigilância.

Com a opção MDI, podem ser ancoradas janelas a qualquer lado da janela de pai; com SDI elas podem ser ancoradas apenas em baixo da barra de menu. Capacidades de ancorar permitem estar dentro em ou fora para uma determinada janela selecionando a caixa de verificação apropriada na aba Janela da caixa de diálogo de Opções, disponível do comando Opções no menu Ferramentas.

Ancorar ou desancorar uma janela

1. selecione a janela você deseja ancorar ou desancorar.
2. arraste a janela para a localização desejada segurando o botão de esquerdo do mouse.
O esboço da janela será exibido enquanto você arrasta.
3. Solte o botão do mouse

```

\!!!!!!/
( õ õ )
-----oOOO--(_)-----
| Arquivo baixado da GEEK BRASIL      |
| O seu portal de informática e internet |
| http://www.geekbrasil.com.br         |
| Dúvidas ou Sugestões?                |
| webmaster@geekbrasil.com.br         |
-----oOOO-----
|_| |_|
||  ||
ooO  Ooo
```