

Visual Basic NET

のツボ

2003

第17回 フォームの継承

西田 雅昭
NISHIDA, Masaaki

Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2002
- ASP.NET
- Internet Information Services
- Other:

Level

Samples

・この記事で取り上げたソースコードおよびサンプルプログラムは、付録CD-ROMの¥DOTNET¥TUBOディレクトリに収録しています。

¥LOGINFORM
ログインフォーム

¥INHERITFORM
ログインフォームを継承して作成したテストフォーム

サンプルInheritFormを実行するには、サンプルInheritFormのプロジェクトを開き、LoginForm¥binフォルダ以下のLoginForm.dllを参照設定する必要があります。

前回までに、「クラスの継承」という便利な機能を紹介しました。Visual Studio .NETでは、フォームもまたクラスですから、フォームも継承できるはずですが。今回は、フォームの基本クラスを作成し、これを継承して使う方法についてお話ししましょう。



はじめに

継承して使う基本フォームの題材は何にしようかと迷ったのですが、比較的内容が簡単で、よく使うフォームとして「ログイン」フォームを作成することにしました。

VB6.0では、「フォームモジュールの追加」を選択すると「ログインダイアログ」というテンプレートがあり、「ログイン」フォームを追加することができました。これと似たような機能をVisual Studio .NETで実現することにしましょう。

なお、今回からVB6.0と.NETでの記述方法の違いについては説明を省きます。コードの記述方法に関して、疑問を感じた際にはヘルプを参照してください。その代わりに、Visual Studio .NET

で便利になった機能などについての説明を織り交ぜてゆくことにします。

また、今までは実験でしたが、今回は実用になるようなプログラムにするため、きちんとしたコードを書くことにします。



ログインフォームを作る ～「LoginForm」

まずログインフォームを作ることから始めましょう。

もしこれから、どんどん継承元になるいろいろなフォームを作るのなら、「MyLibrary」というフォルダを作っておくと便利です。

Visual Studio .NETを起動して、「スタートページ」で[新しいプロジェクト]ボタンをクリックし、この「MyLibrary」

図1：新規プロジェクトの作成



に「LoginForm」というプロジェクトを指定します(図1)

フォーム上にラベル、テキストボックス、ボタンを2つずつ配置し、表1のようにフォームとコントロールのプロパティ設定を行なってください。

これでフォームの外観はでき上がりました(図2)

「Visible」プロパティの設定について、少し説明しておきましょう。このフォームを表示した際に、「ユーザー名」テキストボックスに入力するか、[キャンセル]ボタンを選択する(ログインをやめる)以外の操作ができないようにしています(図3)。こんな簡単なフォームでは、あまり気にする必要はないかもしれませんが、ユーザーの誤操作をできるだけ減らす方法をお見せしたかったのです。

「btnCancel」ボタンの「CausesValidation」プロパティについては、後で説明します。

ここでよく見ると、ラベルの文字とテキストボックスの水平位置がずれています。見苦しいので直すことにします。コントロールの位置関係を手動で揃えるのは、結構面倒です。

「ユーザー名」ラベルと右隣のテキス

トボックスを同時に選択しておいて([Ctrl]キーを押しながら2つのコントロールを順にクリック)、メニューから[書式]-[整列]-[左右中央]を選択すると(図4)この2つのコントロールの上下の位置が揃います。同様にして「パスワード」ラベルと右隣のテキストボックスの位置も調整してください。

参考

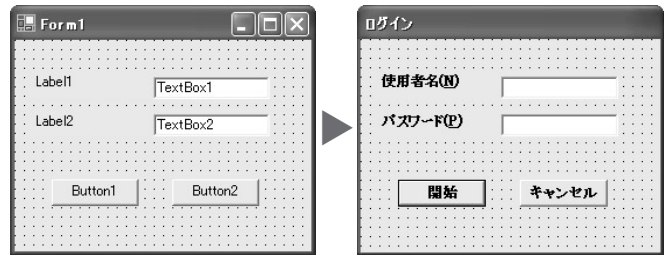
[書式]メニューのサブ項目の名前はわかりにくいと思います。[上]と[下]の間にある[左右中央]が、上下の位置を揃える機能で、[右]と[左]の間にある[上下中央]が左右の位置を揃える機能です。名前の付け間違いかもしれません。

さらにこの段階で、メニューから[書式]-[コントロールのロック]を選択してコントロールが動かないようにしておくといいでしょう。

●タブオーダーとボタンの動作設定

ラベルの「Text」プロパティにアクセスキーの設定(&N)を行なっているので、「TabIndex」をきちんとしなければなりません(ラベルのアクセスキーを選択すると、タブオーダーで次のテ

図2：フォームのユーザーインターフェイス



ラベル、テキストボックス、ボタンを2つずつ配置

プロパティ設定後のフォーム

図3：ユーザー名を入力するか、[キャンセル]ボタンを選択することしかできない

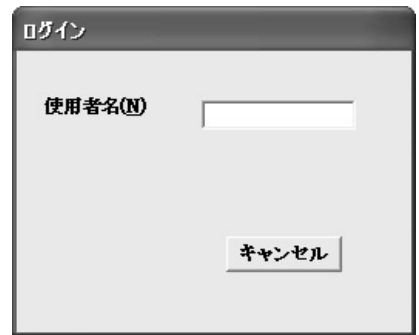
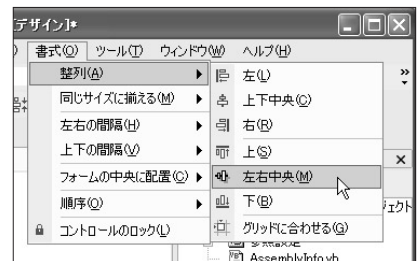


図4：[書式]メニュー



キストボックスにフォーカスが移動します)

Visual Basic .NETでは、タブオーダーの設定は非常に楽になりました。メニューから[表示]-[タブオーダー]を選択すると、コントロールの左上に反転した番号を表示してくれます。ここで、設定したいタブオーダーの順序に従って、この数字をクリックしてゆくと、「0」から始まるタブオーダーを順に付けてくれます(図5)

もう一度メニューで[表示]-[タブ