

# VB.NET

コンポーネント活用講座  
～ 教養課程 ～



# スモールパーツ プログラミング

第19回

小森 大輔 KOMORI, Daisuke

## MainMenuコントロール、 ContextMenuコントロール — Part2 —

### 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¥SMALLディレクトリに収録しています。

¥IMAGEMENU1  
メニューへのアイコンの表示

¥IMAGEMENU2  
アクセスキーの表示を正常にしたサンプル

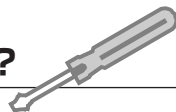
¥IMAGEMENU3  
ショートカットキーを表示させたサンプル

¥IMAGEMENU4  
ショートカットキーの表示を正常にしたサンプル

本連載は、Visual Basic .NET (以下VB.NET) の標準コントロールにスポットを当て、とことん使い込んで、その機能や役割を探ってゆきます。

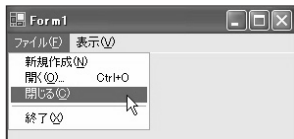
前号からメニューを表示させるコントロールを取り上げ、MainMenuコントロールの基本的な使用方法までを説明してきました。今回はこのメニューをもう少し見栄え良く使用する方法について解説してゆきましょう。

### 標準では 付いてない?



MainMenuコントロールで作成するメニューの操作方法はわかりましたが、前号

図1: MainMenuコントロールのメニュー



で作ったメニュー(図1)はなにか物足りないような気がしませんか?

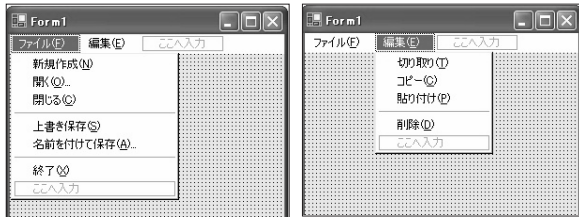
Excelや、Visual Studio .NETのメニュー(図2)と見比べてみるとわかんと思いますが、MainMenuコントロールで作成したメニューには、アイコンが付いていないんです。それでは、MainMenuコントロールのImageListプロパティに、アイコンを格納したImageListコントロールを設定して、メニュー項目を表わすMenuItemオブジェクトのImageIndexプロパティに、表示するアイコンのインデックスを指定すれば? なーんて言ってもそんなプロパティはありゃしません。悲

図2: 他のアプリケーションのメニュー



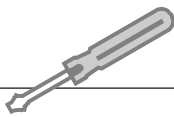
Small Parts Programming

図3：使用するメニュー



しいですね……。

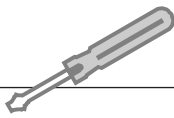
## 付いてないなら 付けましょう!



標準でアイコンを付けることができないのなら、“描く”までです。MenuItemオブジェクトにはOwnerDrawプロパティがあり、このプロパティを“True”に設定することにより、メニューアイテムの描画をWindowsに任せるのではなく、コード上で行なうことができるようになります。

メニュー項目を独自に描くことを考える前に、まずはプロジェクトを作成し、フォームに貼ったMainMenuコントロールに表示させるメニューを作成してしまいましょう。今回は図3のようなメニューを作成しておきます。

## メニュー項目の オーナーードロー



メニュー項目のオーナーードローを行なうには、まずオーナーードローを行なうMenuItemオブジェクトのOwnerDrawプロパティを“True”に設定します。こうすることにより、MenuItemオブジェクトに関連付けられた「MeasureItemイベント」「DrawItemイベント」が発生します。

表1：MeasureItemEventArgsクラスのプロパティ

プロパティ	内容
Graphics	メニュー項目のグラフィックスオブジェクト
Index	メニュー項目のインデックス
ItemHeight	メニュー項目の高さ
ItemWidth	メニュー項目の幅

## MeasureItemイベント

MeasureItemイベントは、「メニューの描画が必要になる前」に発生します。このイベントには、表1のようなMeasureItemEventArgsクラスのプロパティとしてイベントデータが送られます。

MeasureItemイベントプロシージャでは主に、メニュー項目の描画サイズを決定し、ItemHeightプロパティとItemWidthプロパティへ値をセットします。

## DrawItemイベント

DrawItemイベントは、「メニュー項目の描画が必要なタイミング」で発生し、このイベントプロシージャでは実際の描画処理を行ないます。DrawItemイベントには表2のようなDrawItemEventArgsクラスのプロパティと

表2：DrawItemEventArgsクラスのプロパティ

プロパティ	内容
BackColor	メニュー項目の背景色
Bounds	メニュー項目の境界を表わす四角形
Font	メニュー項目のフォント
ForeColor	メニュー項目の前景色
Graphics	メニュー項目のグラフィックスオブジェクト
Index	メニュー項目のインデックス
State	メニュー項目の状態

表3：DrawItemState列挙体

メンバ名	内容
DrawItemState.Checked	メニュー項目がチェックされている状態
DrawItemState.ComboBoxEdit	ComboBoxの編集部分になっている状態
DrawItemState.Default	デフォルト状態の表示
DrawItemState.Disabled	無効状態
DrawItemState.Focus	フォーカスを得ている状態
DrawItemState.Grayed	使用不可の状態（淡色表示）
DrawItemState.HotLight	ホットトラッキングの状態
DrawItemState.Inactive	アクティブではない状態
DrawItemState.NoAccelerator	キーボードアクセラレータなしでの表示
DrawItemState.NoFocusRect	フォーカスを表わす表示がない状態
DrawItemState.None	状態がない
DrawItemState.Selected	選択されている状態