

西田雅昭
の

新

Visual Basic!

西田 雅昭
NISHIDA, Masaaki

基本からの
業務アプリケーション構築技法

「ロボ」

第4回 プロシージャ



プロシージャとは

前回は、フォームの設計とコントロールの配置についてお話ししました。今回は、プロシージャの使い方について紹介します。

大きなプログラムを作成するには、コードを細かく分割しておく管理が簡単になりますし、読みやすくなります。

頻繁に使われる計算や繰り返し実行

するタスクなどのコードをひとかたまりにしておけば便利です。この分割したコードのかたまりを「プロシージャ」と呼びます。

実際のプログラムでは、このプロシージャに名前を付けて、コード内の他の場所から呼び出すことになります。これを「呼び出し元のコード」と呼びます。

Visual Basic 2005のプロシージャには、次の5つの種類があります。

① Sub プロシージャ

何らかの処理を実行しますが、呼び出し元には値を返しません。

② Function プロシージャ

Sub プロシージャと同じく何らかの処理を実行しますが、呼び出し元に値を返します。

③ イベントハンドラ

Sub プロシージャの一種で、イベントに応じて処理を実行します。

④ Property プロシージャ

オブジェクトのプロパティの値を取得または設定します。

⑤ 演算子プロシージャ

定義したクラスまたは構造体で、標準の演算子 (*、<>、And など) の動作を定義します (構造体については記事末コラムを参照)。

レベル >>> Level

1 2 3 4 5

サンプル >>> Sample

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、<http://www.shoeisha.com/mag/windev/>からダウンロード可能です。

言語 >>> Language

Visual Basic

ツール >>> Tool

Visual Studio 2005 Professional

*) 本連載では Visual Studio 2005 Professional を使用していますが、解説内容は特に明記しない限り、Visual Basic 2005 Express でも利用できます。Express Edition をお持ちの方もぜひ挑戦してみてください。



プロシージャの作成

まずはプロシージャを記述してみましょう。コードエディタで、「Public Class Form1」の下に、

```
private sub mprcTest
```

と入力し [Enter] キーを押してください

図1：プロシージャを宣言

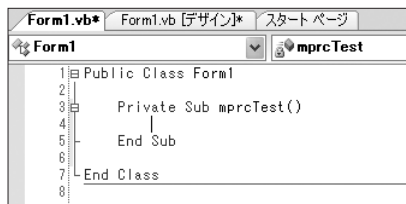


図2：自動生成されたイベントハンドラ

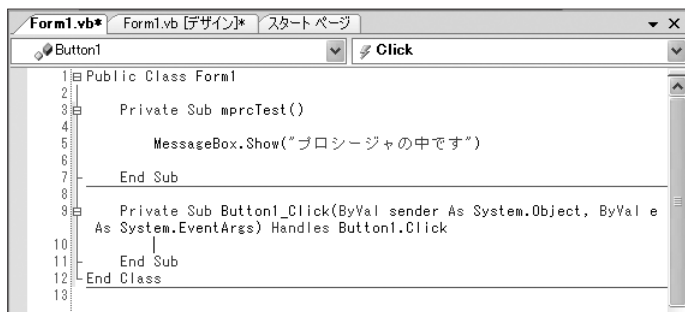


図3：[Ctrl]+[Space] キーで表示した入力一覧



い（プロシージャ名以外には大文字は使いません）。すると瞬時に入力した文字が以下のように変わります（図1）。

```
Private Sub mprcTest()
```

```
End Sub
```

宣言を記述している際に、ミスタッチをすると、あとでひどい目にあいます。そこで私は、自分が作成した名前（プロシージャ名や変数名など）以外は必ず小文字で入力するようにしています。大文字に変わらなければ、間違っているとわかるからです。

図1のように、予約語など、必要な部分は自動で大文字になりますし、ちゃんとインデントのかかった位置でカーソルが点滅しています。さらに、プロ

シージャ名の後ろに括弧も付けてくれますし、1行空けて「End Sub」も記述してくれます。

次に、[Enter] キーを押して空行を作り、

```
MessageBox.Show("プロシージャの中です")
```

と記述してください。これで完成です。

続いて、フォームを表示してフォーム上にボタンをひとつ配置し、そのボタンをダブルクリックしてみましょう。

自動的にコードエディタに戻り、

```
Private Sub Button1_Click(……
```

と表示していますね（図2）。これはボタンをユーザーがクリックした際に起動するプロシージャで、「イベントハンドラ」と呼びます。

先ほど作成した「mprcTest」プロシージャの「End Sub」の下に横線が入り、さらに空行を作っています。コードを読みやすくし、プロシージャの内容を記述するコメントのための行を確保してくれているのです。

カーソル位置で「m」と入力し、[Ctrl]+[Space] キーを押すと選択一覧が出ます。この状態でさらに「p」を入力すると一覧に「mprcTest」が出ますから（図3）、[Enter] キーを押します。すると、次のように後ろにカッコが付きました。

```
mprcTest()
```

Visual Basic 2005では、パラメータ/戻り値のあるなしにかかわらず、