

# Visual Basic NET

# のツボ

## 第28回 ADO.NETを利用した データベース処理 —その8—

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。

#### ¥OLD

前回作成した「ADOTest3」プロジェクト

#### ¥ADOCODE4

今回作成したサンプルアプリケーション

#### ¥DATA

サンプルデータベース (Shimei.mdb)

\*) 今回のサンプルプログラムは、データファイルを「E:\dotNET\Magazine¥VB21¥Data¥」ディレクトリに配置しているという前提で作成しています。サンプルを実行するには、上記のディレクトリにデータファイルを配置するか、「Imprc ConnectMake」プロシージャの接続文字列をご自分の環境に合わせて修正してください。

本連載では、ADO.NETの使い方を、いろいろ実験をしながら作成してきました。これまで、ウィザードで作成したコードを参考にして、ADO.NETを利用するコードをすべて手作業で書くことに挑戦しました。そのために、ADO.NETに関するオブジェクトが、「ウィザードで作成したもの」「コードを書いて作成したもの」の2つになってしまいました。また、実験のためのボタンをたくさん作成したために、画面が煩雑になってしまいました。

今回はまず、プロジェクトをわかりやすいものに作り替えてから、実験を進めたいと思います。



### 新しいプロジェクト を作ろう

はじめに、Visual Basic .NET (以下VB.NET) を起動し、「スタート」画面で [新しいプロジェクト] ボタンをクリックして「ADOTest4」というプロジェクトを作成します。

メニューから、[プロジェクト] - [既存項目の追加] と選択すると「既存項目の追加」ダイアログボックスが開きます。先月号で作成した「ADOTest3」プロジェクトの以下のファイルを選択して (図1)、[開く] ボタンをクリックしましょう。

dsShimei.vb  
dsShimei.xsd  
Form1.vb

「Form1.resx」というファイルは既に存在します。置き換えますか?」など、続けていくつかのメッセージボックスが開きますから、すべて [はい] を選択してください。

### 参考

「ADOTest3」プロジェクトは、付録CD-ROMの¥DOTNET¥TUBO¥OLDフォルダに収録しています。適当な場所にコピーして使用してください。

図1: 「既存項目の追加」で3つのファイルを選択

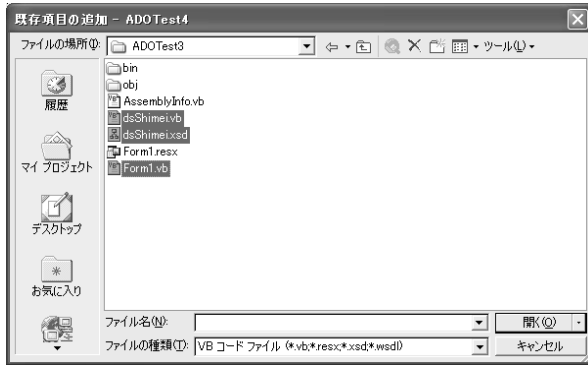
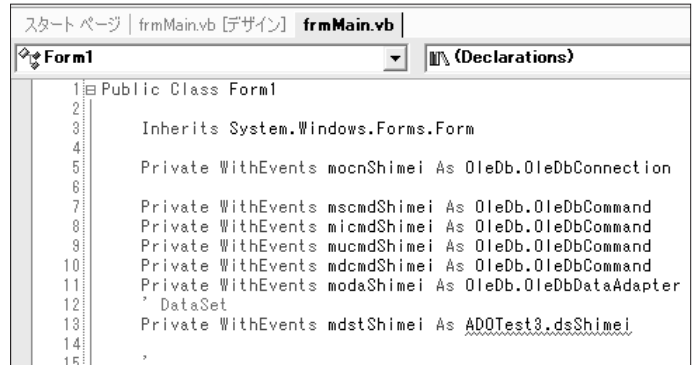


図2: “ADOTest3.dsShimei” の下に波線が付く



続いて、ソリューションエクスプローラで「Form1.vb」を選択し、プロパティウィンドウで「ファイル名」プロパティを“frmMain.vb”に変更します。

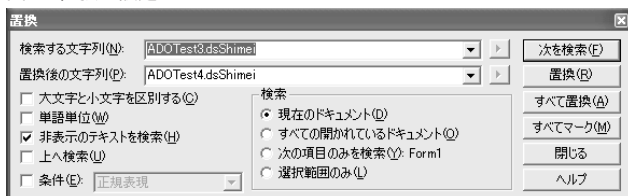
「frmMain.vb」のコードウィンドウを開くと、

```
Private WithEvents mdsShimei As ADOTest3.dsShimei
```

というコードの「ADOTest3.dsShimei」部分に、エラーを意味する波線が付いています(図2)。データセット「dsShimei」を新しいプロジェクトにコピーしたので、当たり前ですね。

この「ADOTest3.dsShimei」を選択状態にして [Ctrl] + [H] キーを押します。「置換」ダイアログボックスが開きますから、「置換後の文字列」テキストボックスに“ADOTest4.dsShimei”と入力します。「非表示のテキストを検索」チェックボックスにチェックがあることを確認して(図3)、[すべて置換]

図3: 置換の設定



ボタンをクリックしてください。「3個のアイテムを置換しました」というメッセージが出るはずですが。

この段階で、正しく訂正できたかどうかテストしてみましょう。メニューから、[ビルド] - [ソリューションのビルド] を選択して、このプロジェクトをビルドし、ビルドに成功したら [F5] キーでプログラムを実行します。

[Connect] → [Adapter] → [Select] → [DataSet] → [Fill2]

の順にボタンをクリックしてみましょう。「接続」をオープンしたことを表わすメッセージボックスが現われるので [OK] ボタンをクリックし、データが表示すれば成功です。

### ●ボタンのイベントハンドラをプロシージャに変更する

次に、デザインモードに戻り、フォー

ムの [Connect] ボタンをダブルクリックして、イベントハンドラを表示します。そして、

```
Private Sub btnConnect_Click( _
    ByVal sender As System.Object, _
    ByVal e As System.EventArgs _
) Handles btnConnect.Click
```

の部分、

```
Private Sub mprcConnectMake()
```

に変更します。表1のボタンに対しても同じ要領で各イベントハンドラを変更してください。

続いて、以下の5つのイベントハンドラをすべて消去します。

```
btnFill1_Click
btnFirst1_Click
btnPrevious1_Click
btnNext1_Click
btnLast1_Click
```

コードの変更/削除が終わったら、フォームのデザインモードに戻って、図4のように不要なボタンとラベルを消去します。残ったコントロールの位置を少し動かして、フォームのデザインをスッキリさせましょう(図5)。

最後に、フォームの下のコンポーネントトレイにある“OleDbConnection”