

完全掌握

DBアプリケーションなんて
カンタンだ!

SQL Server

プログラミング

再入門

第

2 回

デザイナーを使って
アプリケーションを
作ってみよう

株式会社システムインテグレータ
湯尾 守 YUO, Mamoru
<http://www.sint.co.jp/>

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¥SQLServerディレクトリに収録しています。

¥SINGLETABLE

Departmentsテーブルを操作するアプリケーションのソースファイル

¥MULTITABLE

2つのテーブルを操作するアプリケーションのソースファイル

¥SQL

テーブルとサンプルデータを作成するSQLスクリプトの収められたファイル

はじめに

Visual Studioにはコード量を減らせるような工夫が随所に盛り込まれています。先日、とあるところで生産性を高めるためのツールの紹介を受ける機会があったのですが、Visual Studioよりもさらに開発者の記述するコード量を減らすことは難しいという話でした。これについては私も同感です。

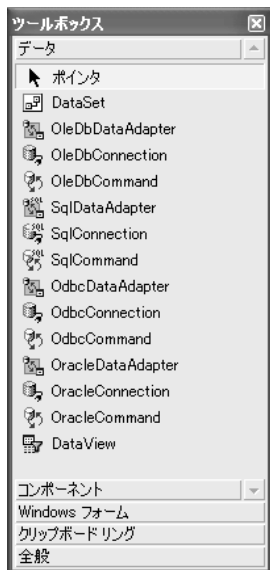
Visual Studioのデザイナーを使うことは「この恩恵を最大限に受ける」ということを意味します。そしてデザイナーによって生成されたコードを編集することもできます。開発ではラクのできる場所はラクをするのが基本です。ということで今回のテーマは、

デザイナーを使ったデータベースアプリケーション作成
です。

ツールボックスにある クラス

Windows フォームのデザイナー画面のとき、ツールボックスには「データ」「コンポーネント」「Windows フォーム」などのタブにさまざまなクラスが分類されて置いてあります。このうち「データ」タブにあるのがADO.NETのクラス群です(図1)。ADO.NETと聞くとそれだけで難しそうですが、要はこの部品をフォームにドラッグ&ドロップすればADO.NETを使うことができるということです。Visual Studio .NET 2003の場合、「データ」タブにはデフォルトで、「DataSet」「DataAdapter」「Connection」「Command」「DataView」の5種類のクラスがあります。このうちデータベースアクセスに使われるのはDataAdapter、Connection、Commandの3つで、頭に“OleDb”“Sql”“Odbc”“Oracle”が付いた4つのタイプ(Visual

図1: ツールボックスの「データ」タブ



Column

接続プール

SQL Serverの接続には「接続プール」というものが使用できます。SqlConnectionオブジェクトを使ってSQL Serverに接続するとSQL Serverは接続文字列に関連付けられた接続プールというものを作成し、それを通じてSQL Serverへのアクセスを可能にします。そして接続が閉じられてもこの接続プールは削除せず、プール内にとどめておきます。再度、同じ接続文字列を持つSqlConnectionオブジェクトから接続を求められた場合、SQL Serverは新しい接続プールは作成せずにプール内に同じ接続文字列の接続プールがないかどうか検索し、先ほど作成した接続プールを再利用します。

つまりSQL Serverの接続の確立や接続の解放は、SQL Serverとの接続そのものの作成や破棄を行なっているのではなく、一時的な接続の取得と接続の放棄を行なっているに過ぎません。したがってSQL Serverへの接続とその解放はこまめに行なってもパフォーマンスはさほど落ちません。むしろ同時に多数の接続を抱えることのほうがデータベースの負荷は大きくなります。

Studio .NET 2002では“OleDb”“Sql”の2つ)があるはずですが、Sqlが頭に付いたものがSQL ServerやMSDEへ、Oracleの付いたものはOracleへネイティブ接続するためのクラスで、これらは非常に高速に動作します。また、OleDbの付いたものはOLE DB接続するためのクラス、Odbcの付いたものはODBC接続のためのクラスで若干速度は劣ります。

SQL Serverへの接続はOLE DBやODBCを利用してよいのですが、この連載ではSqlが頭に付いたクラスを利用することにします。

■ DataSetクラスとDataViewクラス

かつてのデータベースアプリケーションは開始時またはログイン時にデータベースに接続し、終了時またはログアウト時に接続を閉じていました。これは接続を開いたり閉じたりするコストが高かったためです。しかし今日のデータベースは接続に関するコストは低くなり、むしろ同時接続数が増えるほうが問題なので、こまめに接続を開いたり閉じたりするほうがパフォーマンス的に優位であることが多いようです。またWeb環境では接続したままにすることはあり得ないでしょう。

そこでADO.NETではデータベースに常に接続しているのではなく、必要なときにだけ接続し、データをアプ

リケーション側にいったん退避させ、アプリケーション側で編集した後にデータベースと同期を取るような仕組みを用意しています。そのアプリケーション側のデータの退避先が「DataSetオブジェクト」です。そして退避したデータを表示用のデータに加工するのが「DataViewオブジェクト」の役目になります。

■ データベースアクセスのためのクラス

SQL Serverへの接続を提供するのが「SqlConnectionオブジェクト」です。そして接続されたSQL Serverに対してSQLステートメントを実行するのが「SqlCommandオブジェクト」です。またツールボックスにはありませんが、SELECT文の実行結果を取得するには「SqlDataReaderオブジェクト」を利用します。SQL Serverへのアクセスはこの3つのオブジェクトが中心になります。

「SqlDataAdapterクラス」はSQL ServerとDataSetオブジェクトの同期を取るためのクラスです。SqlDataAdapterオブジェクトでは内部でSqlConnection、SqlCommand、SqlDataReaderの3種類のクラスを利用し、SQL ServerのデータをDataSetオブジェクトに書き出したり、DataSetオブジェクトとSQL Serverとのデータの同期を取ったりします。