

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

第 10 回

Webアプリケーション を作成しよう

再入門

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。

¥SQLWEBPROGRAMMING
今回作成したアプリケーションのソース

¥SQL
今回使用するテーブルのCREATE文とデータを作成するINSERT文

はじめに

みなさんご存知のとおり、ここ数年のインターネットの普及には目を見張るものがあります。そのような流れの中でWebを利用したアプリケーションに注目が集まっています。また、Webアプリケーションは導入や保守の手間がかからないことも注目を受ける一因になっています。そこで今回のテーマは、

Webアプリケーションを作成しよう

です。今回は今まで学んだDataSetやDataViewなどの知識を積極的に活用してゆきます。

Webアプリケーションの特徴

WebアプリケーションとWindowsアプリケーション

Windowsアプリケーションの作成に慣れている人でもWebアプリケーションの開発では戸惑うことが多いでしょう。それはWindowsアプリケーションとWebアプリケーションには以下に示すような大きな違いがあるためです。

違い 1 Windowsアプリケーションでは多くの種類のコントロールを利用できるが、Webアプリケーションでは原則としてHTMLで利用可能なユーザーインターフェイスしか利用できない。

違い 2 Windowsアプリケーションのコードはユーザーの利用するマシン上で実行されるのに対し、WebアプリケーションのコードはWebサーバー上で実行される。

違い3 Windowsアプリケーションではユーザーの操作に対して処理を記述するが、WebアプリケーションではWebサーバーがユーザーの操作を逐一捉えることはできず、Webサーバーに対するリクエストがあったときのみ処理が実行される。

違い4 Webアプリケーションはユーザーの使用するブラウザによって画面レイアウトや操作性が異なることがある。

Webアプリケーションを作成する際はこれらの特色を理解しなければいけません。また、Webアプリケーションの最大の利点は「ユーザーのマシンにWebブラウザさえインストールされていれば利用できる」ということにありますが、これは逆に言えば“HTMLやブラウザの仕様の影響を受ける”ということを意味します。この点がWebアプリケーションの非常に大きな制約となります。

Webアプリケーションの仕組み

今回は社員データの照会と更新が可能なWebアプリケーションを作成します(図1)。ユーザーがこのページのURLをリクエストするとWebサーバーはデータベースサーバーからデータを取得し、社員一覧のHTMLを作成し、それをユーザーに送信します。

ここで重要なのは「WebサーバーはHTMLを送信した後送信したページの情報をすべて破棄してしまう」ということです。たとえば図1は社員番号「96451」という社員の情報を修正しようとしている状態ですが、この

図1：サンプルアプリケーション

社員番号	社員氏名	部門	生年月日	入社日	
95798	山本紀子	知的財産部	1973/01/22	1995/06/01	編集
95064	相川作春	国内事業企画部	1972/06/24	1995/09/01	編集
96022	大木幸枝	国内カスタマーサポート部	1973/04/09	1996/04/01	編集
96451	中森明	国内法人営業部	1971/09/10	1996/04/01	更新 キャンセル
96601	本永雄三	海外営業企画部	1969/08/31	1996/04/01	編集
96241	曾根千春	総務部	1973/05/11	1996/07/01	編集
96157	加藤音代	グローバルサポート部	1971/01/19	1996/10/01	編集
97035	秋山英治	知的財産部	1974/06/22	1997/04/01	編集
97245	菅大輔	国内カスタマーサポート部	1974/07/30	1997/04/01	編集
97685	村田巖	海外事業企画部	1974/06/14	1997/04/01	編集

ページを送信した後、Webサーバーは何番の社員をユーザーが編集しようとしていたかを完全に忘れてしまいます。ユーザーが[更新]ボタンをクリックしたときにどの社員のデータを更新するのはWebサーバー上のメモリの情報からではなく、ユーザーが送信してきたページから直接読み取らなくてはなりません。もちろん、Webサーバー上に更新情報を確保する手段がないわけではありませんが、Webアプリケーションではサーバー上に送信したHTMLの情報をすべて確保しておくというのは現実的ではありません。

ViewStateの注意点

ユーザーが送信してきたページから必要な情報を読み取る仕組みを提供するのが「ViewState」というオブジェクトです。ASP.NETでは次のリクエストを処理するときに必要と思われる情報をViewStateというオブジェクトに格納し、そのデータをHTML内に埋め込んで送信します。そしてユーザーがデータを送信するときに前回HTMLに埋め込んだViewStateの情報も同時に送信してもらい、必要な情報を復元して利用します。

ViewStateのデータはHTMLに埋め込むために文字列化されるのですが、この文字列化の処理とその文字列を元のデータに復元する処理はかなりコストがかかります。そのためViewStateに大きなオブジェクトを埋め込むとWebサーバーには非常に大きな負荷がかかります。また、すべてのオブジェクトが文字列化できるわけではないので、ViewStateに格納できないデータもあります。さらに、Date型はViewStateに格納できますが、文字列化する際に秒以下が切り捨てられてしまうため、元の時刻を完全に復元することはできません。

Webアプリケーションではこれらのことに常に気をつけなくてはなりません。

データベースWebアプリケーションを作ってみよう

Webフォームの作成

今回のサンプルではDepartmentsテーブルとEmploy