

これを知らなきゃ年は越せない!?

# NET プログラミング Tips 歳末大放出

特集 2

## Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2002
- ASP.NET
- Internet Information Services
- Other:
  - .NET Framework SDK

## 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、翔泳社のdotNETマガジンWebサイトで公開しています。
- ・誌面ではVisual Basic .NETを対象にしていますが、サイトではVisual C#のソースコードも公開しています。

### [サイト]

<http://www.shoeisha.com/mag/dotnet/>

- ・2003年12月号のページへと進んでください。

## 基本はやっぱりファイルの読み書き

Level



リスト1：テキストファイルの読み込み／書き込み

```
' ファイルの読み込み
Private Sub Button1_Click(...)
    Dim Buff1, Buff2 As String
    Dim fileName As String

    If OpenFileDialog1.ShowDialog = _
        DialogResult.OK Then
        fileName = OpenFileDialog1.FileName
    Else
        Exit Sub
    End If

    FileOpen(1, fileName, OpenMode.Input)
    While Not EOF(1)
        Buff1 = LineInput(1) & Chr(13) + Chr(10)
        Buff2 = Buff2 + Buff1
    End While
    FileClose()
End Sub
```

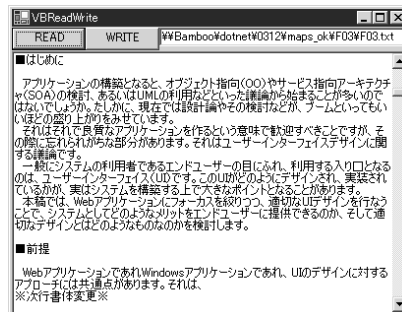
```
TextBox2.Text = Buff2
TextBox1.Text = fileName
End Sub

' ファイルの書き込み
Private Sub Button2_Click(...)
    Dim fileName As String

    If SaveFileDialog1.ShowDialog = _
        DialogResult.OK Then
        fileName = SaveFileDialog1.FileName
    Else
        Exit Sub
    End If

    FileOpen(1, fileName, OpenMode.Output)
    PrintLine(1, TextBox2.Text)
    FileClose()
End Sub
```

図1：サンプル実行画面



## ■ テキストファイルの読み込み

このサンプルでは [READ] ボタンをクリックすると「ファイルを開く」ダイアログボックスが表示されるので、開きたいテキストファイルを選択後、FileOpen()を使用してファイルのデータをテキストボックスに表示します(図1)。

リスト1では、FileOpen()だと改行(Chr(13)+Chr(10))までを一行と認識して読み込むため、表示するにはこの改行コードがなくなってしまう点。そのため、String型のBuff1とBuff2を用意し、読み込んだ行に改行コードを付与してからテキストボックスに送るようにしています。

言うまでもありませんが、FileOpen()で開いたファイルはFileClose()で閉じます。

[WRITE] ボタンをクリックすると、「ファイルを保存する」ダイアログボックスが表示され、保存場所やファイル

### 文字コード

Shift-JISやJIS、UTF-8など、ファイルに使用されている文字コードは千差万別。これらを意識せずに使用するには「System.Text.Encoding()」を使用します。記述方法は以下のとおり。Encodingを指定しない時はUTF-8となります。

<b>[Shift-JIS]</b> System.Text.Encoding.GetEncoding("932") または、 System.Text.Encoding.GetEncoding("shift-jis")	または、 System.Text.Encoding.GetEncoding("iso-2022-jp")
<b>[JIS]</b> System.Text.Encoding.GetEncoding(50220)	<b>[EUC]</b> System.Text.Encoding.GetEncoding(51932) または、 System.Text.Encoding.GetEncoding("euc-jp")

名を入力後、テキストボックスに入力されたデータを保存します。こちらはFileOpen()のOpenModeで

OutPutを使用し、PrintFile()で出力。FileOpen()を使用したのでFileClose()を使うのはお約束。

## .NETでもINIファイルを使いたい

Level



リスト2：INIファイルの読み込み／書き込み

```
Public Declare Function GetPrivateProfileString _
    Lib "kernel32.dll" Alias "GetPrivateProfileStringA" _
    (ByVal lpApplicationName As String, _
    ByVal lpKeyName As String, ByVal lpDefault As String, _
    ByVal lpReturnedString As System.Text.StringBuilder, _
    ByVal nSize As Integer, ByVal lpFileName As String) _
    As Integer
Public Declare Function WritePrivateProfileString _
    Lib "kernel32.dll" Alias "WritePrivateProfileStringA" _
    (ByVal lpApplicationName As String, _
    ByVal lpKeyName As String, ByVal lpEntryString As String, _
    ByVal lpFileName As String) As Integer

Private Sub Button1_Click(...)
    Dim entryLength As Integer
    Dim strEntryStringsValue As String
    Dim strEntryStrings As New System.Text.StringBuilder(256)

    entryLength = GetPrivateProfileString("FOO", _
        "BAR", "Nothing", strEntryStrings, _
        strEntryStrings.Capacity, "C:\%HOGE.ini")
    strEntryStringsValue = strEntryStrings.ToString
    Label1.Text = strEntryStringsValue
End Sub

Private Sub button2_click(...)
    Dim entryLength As Integer
    Dim strSECTION As String

    strSECTION = TextBox1.Text
    entryLength = WritePrivateProfileString("FOO", _
        "BAR", strSECTION, "C:\%HOGE.ini")
End Sub
```

### APIを使って読み書きする

このサンプルでは禁断のAPIを使用してINIファイルを扱います。本来ならレジストリやXMLファイルに設定を保存するのが.NET流ですが、ここでは「どうしてもINIファイルを使いたい」というケースを想定してサンプルを作ってみました。INIファイル用のクラスがないので、ここではAPIを使用します。

リスト2と図2を見ればわかるように、[WRITE] ボタンをクリックすると、下段のテキストボックスに入力した値を「FOO」セクションの「BAR」の値としてC:\%HOGE.iniファイルに書き込みます。[READ] ボタンをクリックすると、上段のラベルにC:\%HOGE.iniの値を表示します。

APIの宣言部分は、書き込み用は、

図2：サンプルを実行し作成されたINIファイルをメモ帳で開く

