

大澤 文孝
OSAWA, Fumitaka

クラスライブラリ活用法

このクラスはココに効く ~ 各種実践編 ~

Technology Tools

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

Level

Samples

はじめに

今回の特集記事「クラスライブラリ活用法~文字列基礎編~」では、文字列を中心とした基礎的なクラスライブラリの使い方を紹介しました。

本稿では、もう少し実用的な処理をいくつか見てゆくことにします。

文字コードの変換

すでに「文字列基礎編」でも説明しましたが、.NET Frameworkで使われる文字コードはUnicodeです。

それ以外の文字コードを扱いたい場合には、何らかの変換が必要となります。

任意の文字コードのストリーム

ファイルを読み書きするには、Stream Reader オブジェクトやStream Writer オブジェクトを使います。

このとき、これらのクラスのコンス

トラクタの第2引数にEncoding オブジェクトを指定すると、任意の文字コードで読み書きできるようになります。

たとえば、「C:%test_sjis.txt」というシフトJISのファイルを読み込んで、EUCに変換して「C:%test_euc.txt」というファイルに書き出す処理をしたいならば、リスト1のようにします。

リスト1- や の部分にあるように、Encoding オブジェクトは、Encoding クラスのGetEncoding メソッドを使って得ます。

```
Dim objEnc As System.Text.Encoding
objEnc = _
System.Text.Encoding.GetEncoding(_
"キャラクタセット")
```

GetEncoding メソッドの引数に指定できる主なキャラクタセットを、表1に示します。キャラクタセットは大文字と小文字の区別はありません。

なお、いくつかの特定の文字コードを示すEncoding オブジェクトは、GetEncoding メソッドを使って取得しなくても、Encoding クラスの静的なプロパティを使って取得できます(表2)。たとえば、

リスト1：シフトJISのファイルを読み、EUCとしてファイルに出力する

```

'読み込むファイル(シフトJIS)
Dim in_fs As New IO.StreamReader( _
    "C:%test_sjis.txt", _
    System.Text.Encoding.GetEncoding("Shift_JIS"))

'書き込むファイル(EUC)
Dim out_fs As New IO.StreamWriter( _
    "C:%test_euc.txt", False, _
    System.Text.Encoding.GetEncoding("EUC-JP"))

Dim oneline As String
oneline = in_fs.ReadLine 'シフトJISとして読み込む

While (Not IsNothing(oneline))
    out_fs.WriteLine(oneline) 'EUCとして書き込む
    oneline = in_fs.ReadLine 'シフトJISとして読み込む
End While

in_fs.Close()
out_fs.Close()
    
```

表1：代表的なキャラクタセット

キャラクタセット	意味
iso-8859-1	ASCII文字
EUC-JP	日本語EUC
ISO-2022-JP	JIS
Shift_JIS	シフトJIS
Unicode	Unicode
UTF-7	UTF-7
UTF-8	UTF-8

全キャラクタセットについては、RFC1766 (<http://ietf.org/rfc.html>)やレジストリの「HKEY_CLASSES_ROOT\MIME\Database\Charset」以下を参照。

```
System.Encoding.GetEncoding("UTF-8")
```

と記述する代わりに、

```
System.Encoding.UTF8
```

と記述してもかまいません。

テキストファイルの読み書き処理で注意したいのは、.NET Frameworkにおいて、Stringオブジェクトが格納しているのは、必ず“Unicode”であるという点です。Stringオブジェクトの中にシフトJISコードの文字列やEUCの文字列が格納されることはありません。

つまり、リスト1の処理では、ReadLineメソッドを呼び出すときに、StreamReaderオブジェクトによって「シフトJIS Unicode」の変換処理が行なわれます。

表2：Encodingクラスの静的なプロパティ

プロパティ	意味
ASCII	7ビットのASCII文字。ISO-8859-1と同じ
BigEdianUnicode	ビッグエディアンUnicode
Default	現在のコードページに合致する文字コード
Unicode	リトルエディアンUnicode
UTF7	UTF-7
UTF8	UTF-8

そして、WriteLineメソッドを呼び出すときに、StreamWriterオブジェクトによって「Unicode EUC」の変換処理が行なわれることとなります。

バイト列を変換する

アプリケーションでは、シフトJISコードで記述されたバイト列（Unicode以外で記述された文字列は、.NET Frameworkではバイト列として扱われます）からUnicode文字列に変換したり、逆に、Unicode文字列をシフトJISコードで記述されたバイト列に変換したりしたい場面があります。

そのような場合には、次のように処理します。

バイト列をUnicodeに変換する

リスト2は、任意の文字コードで格納されているバイト列をUnicodeに変換

する関数です。

たとえば、リスト2に示すBytesToString関数を使って次のように記述すると、文字列“漢字”が得られます。

```

'シフトJISコードのバイト列
Dim b() As Byte = _
    {&H8A, &HBF, &H8E, &H9A}
'Unicodeに変換する
Dim destStr As String = _
    BytesToString(b, "Shift_JIS")
'“漢字”を返す
Debug.WriteLine(destStr)
    
```

バイト列をUnicodeに変換するには、EncodingオブジェクトのGetStringメソッドを使います。

まずは、リスト2にあるようにGetEncodingメソッドを使って目的の文字コードに対応するEncodingオブジェクトを得ます。

```
Dim objEnc As System.Text.Encoding = _
    System.Text.Encoding.GetEncoding( _
    encode)
```