特集 SPECIAL FEATURE 1

文字コードに勝つ！

さまざまな文字コードを扱えるテキストビューアを作ろう

大澤 文孝　OSAWA, Fumitaka

Level

1 2 3 4 5

Technology Tools

- [x] Visual Basic
- [ ] Visual C#
- [ ] Visual C++
- [ ] SQL Server
- [ ] Oracle
- [ ] Access
- [ ] Excel
- [ ] ASP.NET
- [x] Other：
  ↓
  正規表現

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、
http://www.shoeisha.com/mag/windev/
からダウンロード可能です。

はじめに

.NET Frameworkでは、ファイルを読み書きするときの標準的な文字コードが、Unicodeをベースとした UTF-8と呼ばれるものになっています。

その一方で、メモ帳をはじめとするテキストエディタでは、シフトJISコードを使って文字を表現します。そのためテキストエディタで作られたテキストファイルを.NET Frameworkを使って読み込むと、文字化けが発生します。

文字コードは、シフトJISコードだけではありません。UNIX系のOSでは、EUC（Extended UNIX Code）が使われますし、電子メールでは、JISコードが使われます。

本稿では、さまざまな文字コードで書かれたテキストファイルを読み書きするテキストエディタを作りながら、.NET Frameworkにおける文字コードの扱い方を説明します。

各種文字コード対応のテキストエディタ

本稿で作るのは、簡単なテキストエディタです。

さまざまな文字コードのファイルを読み書きできるようにするため、図1のように、「開く」メニューや「名前を付けて保存」メニューから、文字コードを指定して開いたり保存できるようにします。「上書き保存」メニューを選んだときには、開いたのと同じ文字コードで保存できるようにします。

また、どの文字コードで開いたのかを示すため、ステータスバーに、その文字コードを表示することにします。

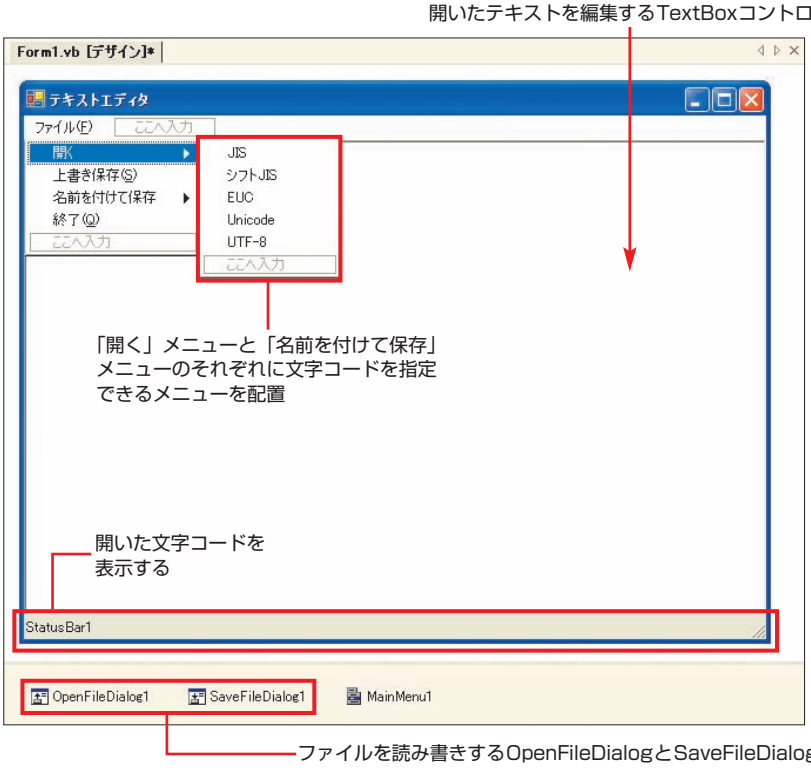
ファイルを開いたり保存したりするには、OpenFileDialogコントロールとSaveFileDialogコントロールを使います[注1]。

扱う文字の種類

世の中には、たくさんの文字コードがありますが、ここでは、次に挙げる5種

注1）このサンプルでは、テキストを編集するコントロールにTextBoxコントロールを用いています。Windows NT系（XP含む）では、MaxLengthプロパティを「0」にすることで、最大4,294,967,295文字の編集ができます。しかしWindows 9x系では、65,535文字までとなります。長いテキストを編集したいなら、RichTextBoxコントロールを使ったほうがよいでしょう。ちなみにTextBoxコントロールのMaxLengthプロパティのデフォルト値は32,767なので、忘れずに「0」に設定しておきましょう。

図1：作成するテキストエディタ



類の文字コードを扱うことにします[注2]。

①JISコード
電子メールなどで主に使われる文字コードです[注3]。

②シフトJISコード
Windows環境やMacOS環境などで主に使われる文字コードです。

③EUC
Unix環境で主に使われる文字コードです。

④Unicode
Unicodeコンソーシアムによって定義された、多言語の文字を2バイトで示す方式です。

⑤UTF-8
Unicodeを可変長にエンコードした表現方式です[注4]。

Unicodeでは、英数字部分（&H00～&H7F）も2バイト（&H0000～&H007F）となるので、英数字しか含まれていない場合、Unicodeで保存するとデータが2倍になります。

しかしUTF-8では、Unicodeにおける&H0080以降だけを可変長で示すことで、英数字部分は1バイトで、それ以外は2～3バイトで表現することで、データの増加を抑えます。UTF-8では、英数字部分のコードはアスキーコードと同じなので、Unicodeをサポートしないアプリケーションでも、英数字だけは読み取れるというメリットもあります。

UTF-8は、インターネットでデータをやりとりするときによく使われるようになりました。たとえば、ASP.NETにおける出力ページのデフォルトの文字コードは、UTF-8です。

テキストファイルの読み書きの基本

テキストファイルを読み書きするには、System.IO名前空間にあるStreamReaderとStreamWriterを使います。

```
Imports System.IO
```

まずはテキストファイルの読み書き処理の基本から説明します。

テキストファイルの読み込み

テキストファイルを読み込むには、StreamReaderオブジェクトを使います。

OpenFileDialogコントロールを使ってユーザーにファイルを選択させ、そのファイルをテキストファイルとして開く一連の処理は、次のようになります。

ファイルを開く

OpenFileDialogコントロールのShowDialogメソッドを呼び出すと、「ファイルを開く」ダイアログボックスが表示され、ユーザーは開くファイルを選択できます（図2）[注5]。

注2）ここでは概要のみ示します。より具体的なコード体系や特徴については、別稿「文字コードの自動判別機能実装」で説明します。
注3）厳密に言えば、電子メールでは、JISコードのサブセットである「ISO-2022-JP」が使われます。ISO-2022-JPでは半角カナは使えません。
注4）UTF-8に似た方式に、7ビットでエンコードする「UTF-7」という方式もありますが、本稿では割愛します。
注5）OpenFileDialogコントロールを使わずに、ファイル名を指定してファイルを開きたいときには、FileStreamオブジェクトを使います。