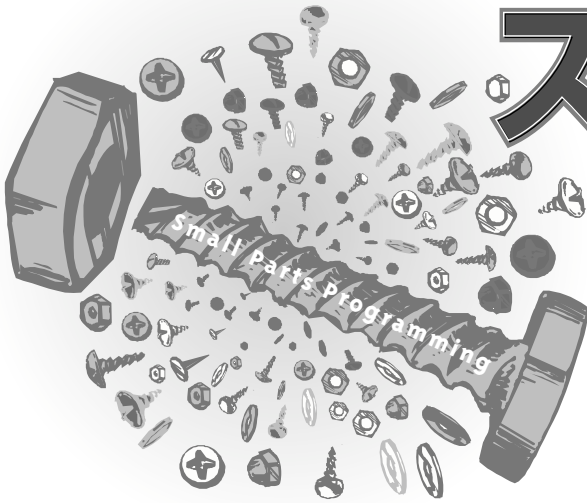


VB.NET

コンポーネント活用講座
～ 教養課程 ～

スモールパーツ プログラミング

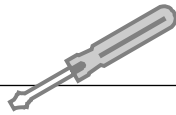


第3回

小森 大輔 KOMORI, Daisuke

TextBoxコントロール ～Part2～

はじめに



本連載は、Visual Basic.NET (以下VB.NET) の標準コントロールにスポットを当て、とことん使い込んで、その機能や役割を探ってゆきます。

今回は、TextBoxコントロールの紹介を行ない、プロパティやメソッド、クリップボードとのやり取りや、デ

ータの検証などについてお話ししましたが、ページ数との戦いで途中でことごとくお話を中断してしまいました。

第3回目である今回は、前回の続きということで、TextBoxコントロールへのデータ入力の規制や、取得したテキストデータの整形といった部分についてお話しします。



本稿で前提となるもの

OS Windows XP

開発環境 Visual Basic.NET



この記事で解説したサンプルプログラムは、付録CD-ROMの¥DMAG¥SMALLフォルダ以下に収録しています

¥LIMIT	: 入力文字数の制限、数値以外の入力文字の制限
¥LIMIT2	: 制限文字判別プロシージャ
¥FUNCTION¥INDEXOF	: IndexOfメソッド使用例
¥FUNCTION¥INSERT	: Insertメソッド使用例
¥FUNCTION¥TRIM	: Trimメソッド使用例
¥FUNCTION¥TOCHARARRAY	: ToCharArrayメソッド使用例
¥FUNCTION¥SPLIT	: Splitメソッド使用例
¥FUNCTION¥JOIN	: Joinメソッド使用例

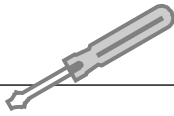
テキストデータの制限と変換



前回お話ししたValidating() イベントプロシージャで行なうデータ検証のように、入力されたデータを検証して規則に反していたら注意を促すのも良いのですが、できればユーザーがルール違反を行わないようにしてあげるのが一番です。ユーザーが思惑通りのデータを入力してくれればこのようなことは考えずに済みますが、ユーザーの行為は必ずと言っていいほどプログラマの考えたほうとは違う方向へ向かいますからね。

まずはTextBoxコントロールへの入力の規制と、コントロールから取得したデータを、ある一定のフォーマットへ変換する処理を考えてみましょう。

データの制限



まずは、TextBoxコントロールへの入力を制限する機能を見てみましょう。

◆ 文字数の制限

入力を制限する方法ですが、文字数の規制を行なう場合には、MaxLengthというプロパティがあります。このプロパティはTextBoxコントロールへ入力することができる最大文字数を表わし、入力文字数が設定した値を超えると文字の入力ができなくなります。

MaxLengthプロパティは文字列のバイト数を表わすものではなく、単純に文字数を指定することができます。「MaxLength = 5」と指定したコントロールには、半角英数字で“ABCDE”も、全角で“あいうえお”も入力することができます (図1)。

.NET Frameworkのドキュメントでは、MaxLengthプロパティのデフォルト値は“0”と記載されていますが、実際には“32767”です。このプロパティを“0”に設定した場合は、無制限 (正確には使用されるメモリの上限) に設定されたこととなります (“0”は入力が一文字も行なえないという設定ではありません)。入力を行なえないようにするには、テキストを読み取り専用を設定するReadOnlyプロパティをTrueに設定します。

◆ 数値以外の文字を制限

たとえば郵便番号を入力するTextBoxコントロールがあ

った場合、入力文字数をMaxLengthプロパティで“7”に制限したとしても、天邪鬼なユーザーは「1234ABC」と入力を行なうかもしれません。このような場合は数値のみの入力を許可し、それ以外の文字は入力できないようにしたいですね。しかし、残念ながらこのような設定を行なうプロパティはありません。

このような場合には、TextBoxコントロールのKeyPressイベントプロシージャで、KeyPressEventArgs引数の値を判断して自前で処理を行ないます (リスト1)。

KeyPressEventArgsクラスは、KeyPressイベントが処理されたかどうかを表わすHandledプロパティと、押されたキーに対応する文字を表わすKeyCharプロパティをもっています。

処理としては、まずKeyCharプロパティで押されたキーを判断し、そのキーが制限したい文字に対応するキーであればHandledプロパティをTrueにして、Windowsによるキーの処理を無効にします。

先の郵便番号のように数値のみを入力させたいのであれば、0～9以外のキーが押された場合に、HandledプロパティをTrueに設定します。

数値の場合、キーに対応する文字をひとつずつ判断しても良いのですが、0～9の数値を表わす文字コードは48～57と連続したものになっているので、文字コードがこの範囲に入っているか否かで判断したほうがすっきりしたコードになります。リスト1は、KeyCharプロパティの値 (文字) をAsc関数で文字コードに変換し、0～9の数値を表わす48～57という文字コードでない場合は無効にする例です (図2)。図2は、試しにテキストボック

図1：MaxLengthプロパティとReadOnlyプロパティ

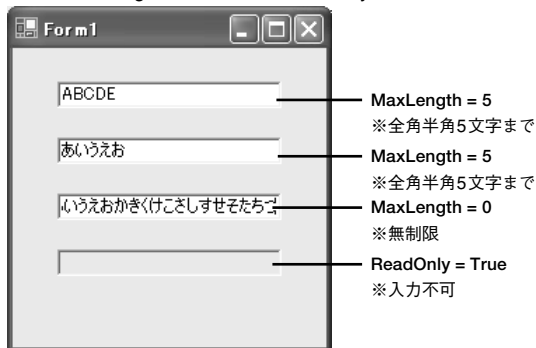


図2：サンプルLimitの実行画面と出力画面

