

けん太の プログラミング 修行記



第11回

「けん太、Visual Basic.NETとC#について考える」の巻 - その3 - プログラミング言語を選択する

碗仔 けん太 (Pochi Company)
WANKO, Kenta

.NETプログラムの開発には、いくつかのプログラミング言語を使うことができます。そして、.NETのプログラミングシーンで急速に普及しようとしている言語は、Visual Basic.NET (以下VB.NET) と Visual C#.NET (以下C#) です。これまでの連載でVB.NETとC#の外観と内部 (CPUで実行される命令: アセンブリ言語) を眺めてみました。その結果わかったことは、“これら2つの言語を使って作ったプログラムは、ほとんど同じものといっ

てよいほど似ている”ということです。今後、VB.NETとC#のどちらを使えば良いのでしょうか。また、どちらかの言語を選択するとしたらどのような理由でどちらを選択すべきなのか、プログラミング言語を選択したあとでそれを効率的にマスターしたり利用したりするためのキーポイントは何か、などについて考えてみましょう。



前回までの あらすじ

.NETプログラミングで、最も関心をもたれているものに**VB.NET**と**C#**があります。どちらも強力な.NETプログラム開発ツールです。そして、似ているようで違うところもある、VB.NETとC#の2つのプログラミング言語に、さまざまな立場からさまざまな意見がでてきます。

実際のところ、VB.NETとC#はどこが同じで、どこがどのように異なるのでしょうか？

この疑問を解くために、本誌2003年1月号でVB.NETとC#を使って同じプログラムを作成し、そのソースプログラムを比較してみました。その結果、ソースプログラムの外観は明らかに異なるということがわかりました。

たとえば、あるキーワードの働く範囲やコードブロックを明示するために、VB.NETでは、次のようにコードを記述します。

本稿で前提となるもの

OS Windows 2000 Professional (SP1)
開発環境 Visual Studio.NET
Microsoft Development Environment 2002 Version 7.0.9486
.NET Framework 1.0 Version 1.0.3705 (SP2)



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

¥MONTHLY : 今回のサンプルプロジェクト
¥SCHEDULE : 今回のサンプルソリューションおよび修正サンプルプロジェクト



VB.NETのコードブロックの形式

```
キーワード  
...  
End キーワード
```

これに対して、C#では次のようにコードを記述します。

C#のコードブロックの形式

```
キーワード  
{  
...  
}
```

また、VB.NETの“行”の概念は、次のようになります。

VB.NETの行の概念

VB.NETでは原則として1行のコード(ステートメント)は改行までであり、VB.NETで改行をこえて論理的な行を繋ぐためには行継続文字「_」(アンダースコア)を使わなければならない

これに対してC#の行の概念は、次のようになります。

C#の行の概念

改行の文字コードで区切られる物理的な行とは関係なく、「;」(セミicolon)があるところまでがひとつのステートメントとみなされる

これらの点が、ソースコードの外観から受ける印象に大きな違いをもたらしていることがわかりました。

しかし、制御構文や例外処理の構文、代入や演算の方法などは、まったく同じではないがかなり似ていて、同じ機

能のプログラムをほぼ同じように記述できるほど、VB.NETとC#が似ていることもわかりました。

次に本誌2003年2月号では、VB.NETとC#とで作った同じ機能をもつメソッドを呼び出して、コンピュータの内部で行なわれていることを調べてみました。その結果わかったことは、同じことをする2つのプログラミング言語がコンパイルされて実際にコンピュータの内部で実行されるCPUの命令(アセンブリ言語)は同じであるということです。

これらの結果から、VB.NETとC#という2つのプログラミング言語について、外観こそ少し違うものの以下のような結論に達しようとしています。

VB.NETとC#についてこれまでに調べてみた結果

- VB.NETとC#は、外観こそ少し違うものの、ほとんど同じ言語である
- VB.NETとC#は、プログラムを同じように書くことができる
- VB.NETとC#は、できたプログラムで実際に実行されるときのコードも同じである

しかし、同じことをするための同じようなものであるなら、プログラミング言語をわざわざ2種類も作る必要はないように思えます。それにも関わらずVB.NETとC#があるのはなぜでしょうか? 今後、VB.NETとC#のどちらかを選択するとしたらどのような理由でどちらを選択するべきなのでしょう? また、どちらかのプログラミング言語を選択したあとでそれを効率的

にマスターする方法や、効果的に使うためのキーポイントは何かあるんでしょうか?

今回は、“VB.NETとC#のどちらを選択してどのように使うべきか”ということについて考えてみます。また、VB.NETとC#を併用するもうひとつの方法も記事末のコラムで紹介합니다。

VB.NETとC#の相 似点とそのメリット

これまでVB.NETとC#について調べてきた結果、この2つのプログラミング言語で作成したプログラムは、ソースコードの外観もコンピュータの内部で実行されるコードも、きわめてよく似ているということが明らかになりました。

制御構文や例外処理、データ型など、プログラミング言語として備えるべき最も基本的な部分は、VB.NETもC#も同じとみなしてよさそうです。なにしろ、どちらの言語も.NET Frameworkを利用してCommon Language Runtime(共通言語ランタイム:以下CLR)上で実行されるプログラムを開発するためのプログラミング言語です。

したがって、VB.NETとC#とのソースプログラムが似ている理由は、次のようにいくつかあります。

VB.NETとC#とのソースプログラムが似ている理由

- データ型はプラットフォームである.NET Frameworkと同じであれば便利。だからデータ型が同じなのは当然。
- 例外処理もプラットフォームに依存