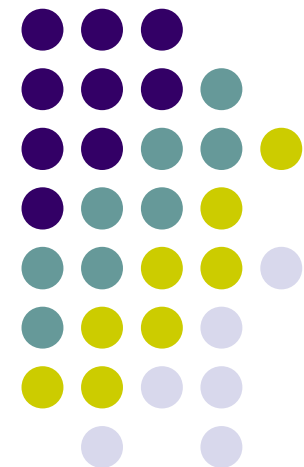


C#入門

加藤 勇也

yuya@katoden.co.jp

<http://nuance.cc/>





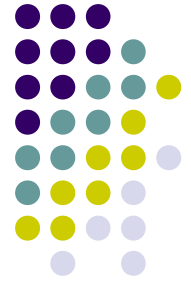
C#とは？

- Microsoftが開発した、新しいオブジェクト指向プログラミング言語
- .NETの中核となる言語
- C++の進化形であるが、Javaを始めとする他言語の影響を大きく受けている
- ECMA-334として標準化



C#の特徴

- C++に比べて、気軽にコーディングできる
 - 開発者に優しい細かい機能が一杯
- クラスライブラリが豊富
 - というか、豊富すぎ
 - クラスライブラリリファレンス 30巻セット125,000円で好評発売中！ (単体合計だと180,000円)
- 高い生産性、拡張性を有している
 - とMicrosoftは主張している
- .NET Framework上で動作



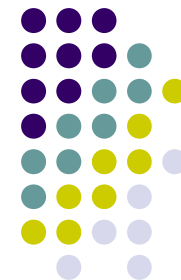
.NET Frameworkとは？

- 実行環境とクラスライブラリの総称
 - CLR (Common Language Runtime)
 - CLI (Common Language Infrastructure)
 - ECMA-335として標準化
- Linux用の.NET Framework
 - Mono Project <<http://www.go-mono.com/>>
 - dotGNU Project <<http://dotgnu.org/>>



.NET Frameworkの特徴

- 複数のプログラミング言語の混在が可能
 - .NET対応のコンパイラはILコードを出力
 - C#コンパイラ、C++コンパイラ、VisualBasicコンパイラなどなど
 - IL(Intermediate Language)コードをCLRで実行
 - データ型はCTS (Common Type System)で規定
 - 言語の連携はCLS (Common Language Specification)で規定
- セキュア
 - 少なくともそうであることを目指している



開発環境 必要なモノ

- Microsoft .NET Framework SDK
 - <http://www.microsoft.com/japan/msdn/netframework/>
 - 無償でダウンロード可能
 - サイズは約130MBytes
 - C#コンパイラ、サンプル、ドキュメントなどが含まれている
 - Microsoft .NET Frameworkを先にインストールしておく必要がある



開発環境 :あると便利なモノ

- 統合開発環境
 - Microsoft Visual Studio.NET / Visual C#.NET
 - Borland C# Builder for the Microsoft .NET Framework
 - C# Builder Personalは無償
 - SharpDevelop
 - オープンソースソフトウェア
 - Eclipse
 - オープンソースソフトウェア
 - プラグインでC#に対応



まずはお約束のHello World

```
1 // hello.cs
2 namespace MyNameSpace
3 {
4     using System;
5
6     public class MyClass
7     {
8         public static void Main(string[] args)
9         {
10             Console.WriteLine("Hello, World!");
11         }
12     }
13 }
```

```
$ csc hello.cs
$ hello.exe
Hello, World!
```

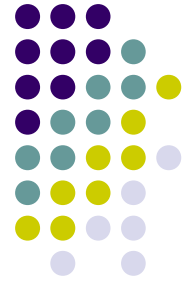

GUI版Hello World



```
1 // hello2.cs
2 namespace MyNameSpace
3 {
4     using System;
5     using System.Windows.Forms;
6     using System.Drawing;
7
8     public class MyForm : Form
9     {
10        public MyForm()
11        {
12            this.ClientSize = new Size(168, 45);
13            this.Text = "Hello World";
14
15            Label label1 = new Label();
16            label1.Text = "Hello, World!";
17            label1.AutoSize = true;
18            label1.Location = new Point(48, 16);
19            this.Controls.Add(label1);
20        }
21
22        public static void Main()
23        {
24            Application.Run(new MyForm());
25        }
26    }
27 }
```

```
$ csc /t:winexe hello2.cs
$ hello2.exe
```

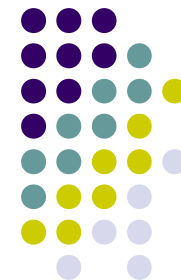




C#の新機能

- property method
- boxing / unboxing
- foreach statement
- delegate
- attribute
- assembly
- etc...

C#の新機能 property



```
1 public class MyClass
2 {
3     protected int id = 0;
4
5     public int ID
6     {
7         get { return this.id; }
8         set { this.id = value; }
9     }
10
11     public static void Main()
12     {
13         MyClass my = new MyClass();
14         my.ID = 123;
15         Console.WriteLine("ID = {0}", my.ID);
16     }
17 }
```

getter/setterを
1つのメソッド内に
記述することができる

getterのみ
setterのみ
でももちろんOK

C#の新機能 boxing / unboxing



```
1 int    a = 123;  
2 object b = (object)a;  
3 int    c = (int)b;  
4 Console.WriteLine("c = " + c.ToString());
```

値型を参照型に変換することをBoxingと言い
参照型を値型に変換することをUnboxingと言う



C#の新機能 foreach

```
1 string[] string_array = {"foo", "bar", "baz"};
2 foreach ( string str in string_array )
3 {
4     Console.WriteLine(str);
5 }
6
7 object[] object_array = {1, 2, "foo", "baz"};
8 foreach ( object obj in object_array )
9 {
10     Console.WriteLine(obj);
11 }
```

foreach文は受動的イテレータ (内部イテレータ)
foreach文に対応させる場合は、IEnumerableを実装する



おしまい

- 要望があれば第2弾も!
 - 新機能の紹介や
 - クラスライブラリの紹介など
- 質問があればどうぞ