

第51章 構造体



C#の構造体は、C/C++の構造体とは全く異なるものです。

構造体(structure)は、どちらかというクラスに近いものです。しかし、決定的に違うことがいくつかあります。クラスは参照型ですが、構造体は値型です。

構造体は、構造体やクラスを継承することはできません。また、継承元になることもできません。

構造体は、次のように定義します。

```
struct 構造体名
{
    ...
}
```

...に構造体のメンバを定義します。構造体のメンバはクラスと同様のメンバを定義できます。(引数無しコンストラクタとデストラクタをのぞく)

では、簡単な例を見てみましょう。

```
// struct01.cs

using System;

struct MyStruct
{
    int struct_x;
    int[] ar;

    public int Show()
    {
        Console.WriteLine("struct_x = {0}", struct_x);
        return 0;
    }

    public int x
    {
        get
        {
            return struct_x;
        }
        set
        {
            struct_x = value;
        }
    }

    public int this[int n]
    {
        get
        {
            return ar[n];
        }
        set
        {
            ar[n] = value;
        }
    }

    public MyStruct(int a)
    {
        ar = new int[a];
        struct_x = 0;
    }
}

class struct01
{
    public static void Main()
    {
        // 独自のコンストラクタの使用
        MyStruct ms = new MyStruct(5);
    }
}
```

```

// プロパティの使用
ms.x = 10;
Console.WriteLine("struct_x = {0}", ms.x);

// メソッドの使用
ms.Show();

// インデクサの使用
for (int i = 0; i < 5; i++)
    ms[i] = i * 10;

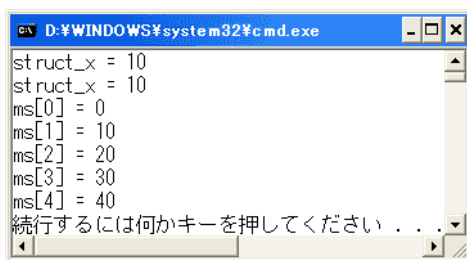
for (int i = 0; i < 5; i++)
{
    Console.WriteLine("ms[{0}] = {1}", i, ms[i]);
}
}
}

```

構造体MyStructには、privateなインスタンスフィールド、publicなメソッド、プロパティ、インデクサをメンバに持っています。また、引数付きコンストラクタもあります。

Mainメソッドでは、MyStruct構造体のインスタンスを使って、プロパティやメソッドやインデクサを利用しています。使い方はクラスと全く同じですね。

実行結果は、次のようになります。



```

D:\WINDOWS\system32\cmd.exe
struct_x = 10
struct_x = 10
ms[0] = 0
ms[1] = 10
ms[2] = 20
ms[3] = 30
ms[4] = 40
続行するには何かキーを押してください . . .

```

[\[C# Index\]](#) [\[総合Index\]](#) [\[Previous Chapter\]](#) [\[Next Chapter\]](#)

Update 26/Sep/2006 By Y.Kumei

当ホームページの一部または全部を無断で複写、複製、転載あるいはコンピュータ等のファイルに保存することを禁じます。