

## 第40章 名前を隠そう



クラスの継承先で、基本クラスと同じ名前のメンバを定義したらどうなるのでしょうか。内容が書きかえられてしまうのでしょうか。

これも実験してみるとすぐにわかります。

基本クラスと同じ名前のメンバを派生クラスで定義すると、コンパイル時に警告がでます。派生クラスで基本クラスのメンバを書きかえたいときはnew修飾子をつけます。これを「**名前の隠蔽**」といいます。情報開示の時代で、「隠蔽」というと何となく、うさんくさい感じがしますが、OOPでは重要な概念の一つです。

クラスのインスタンスを生成するときに使っていたnewとは意味が全く異なるので注意してください。

派生クラスで、new修飾子でメンバを定義し直すと、派生クラスからはもう元のメンバを呼び出せないのかというと、そんなことはありません。これには、baseキーワードを使います。前章で基本クラスのコンストラクタを呼び出すときにもbaseを使いましたね。

さて、new修飾子で書きかえたメンバを持つクラスのインスタンスの参照を、基本クラスの参照変数に代入すると、これは書きかえ前の基本クラスのメンバをそのまま使えます。

では、簡単なサンプルを見てみましょう。

```
// hidename01.cs

using System;

class MyClass1
{
    public int x = 10;

    public int show()
    {
        Console.WriteLine("MyClass1のshowメソッドです");
        return 0;
    }
}

class MyClass2 : MyClass1
{
    new public int x = 20;

    new public int show()
    {
        Console.WriteLine("x = {0}", x);
        return 0;
    }

    public int callold()
    {
        Console.WriteLine("MyClass1のshowメソッドを呼び出します");
        base.show();
        Console.WriteLine("MyClass1のx = {0}", base.x);
        return 0;
    }
}

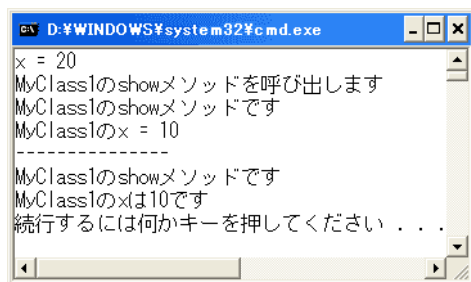
class hidename01
{
    public static void Main()
    {
        MyClass2 mc2 = new MyClass2();
        mc2.show();
        mc2.callold();

        Console.WriteLine("-----");

        MyClass1 mc1; //MyClass1のインスタンスは生成していない
        mc1 = mc2;
        mc1.show();
        Console.WriteLine("MyClass1のxは{0}です", mc1.x);
    }
}
```

この、プログラムを眺めて実行結果がどうなるか想像してみてください。

実行結果は次のようになります。



```

D:\WINDOWS\system32\cmd.exe
x = 20
MyClass1のshowメソッドを呼び出します
MyClass1のshowメソッドです
MyClass1のx = 10
-----
MyClass1のshowメソッドです
MyClass1のxは10です
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 15/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。