

第20章 クラスを作ってみよう



クラスは、オブジェクトの設計図です。実態はありませんが、この設計図から実態を持つインスタンス(オブジェクト)が作られます。

クラスの構成要素をメンバといいます。代表的なメンバは、データを格納するフィールドと動作を定義するメソッドです。

フィールドはクラス変数とも呼ばれます。フィールドにデータを格納したり、ここから読み出したりするのに独自の制限を設けたいときは、「プロパティ」というものを使います。

また、インスタンスを配列のように使うことのできる「インデクサ」というものもあります。

メンバには、クラスをインスタンス化してから利用するインスタンスメンバと、そのまま使えるstatic(静的)メンバがあります。

静的メンバは

クラス名 .メンバ名

で呼び出します。これに対してインスタンスメンバは

オブジェクト名 .メンバ名

で呼び出します。インスタンス(オブジェクト)は

クラス名 インスタンス変数 = new クラス名 (あれば引数) ;

で生成します。

静的メンバはstaticキーワードをつけて宣言します。

また、メンバはどこからでも呼び出せるpublicメンバと、そのクラスからしか呼び出せないprivateメンバ、自分のクラスと継承先からしか呼び出せないprotectedメンバなどがあります。これを、アクセス修飾子といいます。(このほかにもありますが、今のところこの3つを知っていれば不自由はありません。)

アクセス修飾子を省略するとprivateと解釈されます。

クラス定義は

```
class クラス名
{
    メンバ宣言 ;
}
```

のように行います。フィールドやメソッドメンバは

アクセス修飾子 データ型 フィールド名 ;
アクセス修飾子 戻り値のデータ型 メソッド名 (パラメータリスト) ;

のように宣言します。

では、メンバとしてインスタンスフィールドを1つだけ持つもっとも単純なクラスを定義して、このフィールドにデータを代入し、これを読み出すプログラムを作ってみましょう。

```
// simpleclass01.cs

using System;

class MyClass
{
    public int x;
}

class simpleclass01
{
    public static void Main()
    {
        MyClass myclass = new MyClass();

        myclass.x = 10;

        Console.WriteLine("myclass.x = {0}", myclass.x);
    }
}
```

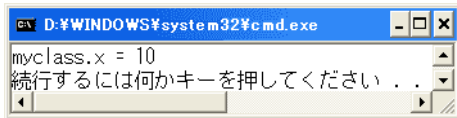
ここでは、MyClassというクラスを自作しています。メンバはインスタンスメンバのxのみです。

Mainメソッドでは、まずMyClassクラスをインスタンス化して、その参照をmyclassに代入しています。

フィールドxに10を代入し、これを読み出しています。

こうやってみると、クラスを作るのは簡単そうですね。(作るのは簡単ですが、役に立つクラスを作るというのは別物です。)

このプログラムの実行結果は次のようになります。



今度は、静的フィールドを1つだけ持つクラスを作ってみましょう。

```
// simpleclass02.cs

using System;

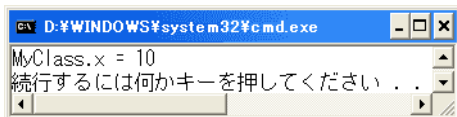
class MyClass
{
    public static int x;
}

class simpleclass02
{
    public static void Main()
    {
        MyClass.x = 10;

        Console.WriteLine("MyClass.x = {0}", MyClass.x);
    }
}
```

静的メンバにアクセスするには、「クラス名.メンバ名」である点に注意してください。

実行結果は次のようになります。



この2つの例では、Mainメソッドのあるクラスの外に新規にクラス定義をしていましたが、もちろんMainメソッドにあるクラス内に、フィールドを定義することもできます。

```
// simpleclass03.cs

using System;

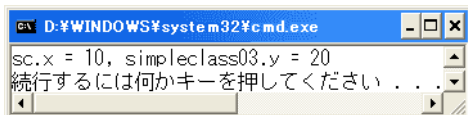
class simpleclass03
{
    int x;
    static int y;

    public static void Main()
    {
        simpleclass03 sc = new simpleclass03();
        sc.x = 10;

        simpleclass03.y = 20;

        Console.WriteLine("sc.x = {0}, simpleclass03.y = {1}",
            sc.x, simpleclass03.y);
    }
}
```

実行結果は、次のようになります。



インスタンスメンバとstaticメンバの呼び出し方の違いに注意してください。また、x,yの宣言をMain関数の宣言の後にしてもらえません。

フィールドx,yにはpublic修飾子がついていないのでprivate扱いとなります。同じクラスからのみ呼び出し可能です。