

Windows Form

La **Windows Form**, ovvero la finestra dell'applicazione, è il fulcro di ogni applicazione Windows. La versione 2.0 del framework .NET introduce parecchie novità nella gestione delle Windows Form, che cercheremo di analizzare nel seguito.

Creiamo una nuova Windows Application (applicazione Windows) con il linguaggio C#. Il progetto include di default una Windows Form, contenuta in un file di nome **Form1.cs**.

Premiamo il tasto F7 per visualizzare il codice sorgente; in alternativa, è possibile fare clic con il tasto destro del mouse sul file «Form1.cs» visualizzato nel **Solution Explorer** (Esplora soluzioni) e selezionare il comando **View code** (Visualizza codice). Viene mostrato il codice:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace WindowsApplication1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

Innanzitutto, notiamo varie clausole `using` che specificano quali sono i namespace in cui cercare le classi che verranno utilizzate.

L'insieme di tutte le classi fornite dal Framework .NET prende il nome **Base class library**. Si tratta di una raccolta vastissima, ed è impossibile anche solo citare tutte le classi che contiene.

Viene poi definito il namespace `WindowsApplication1`. Tutte le Windows Form create con Visual Studio, così come le classi, le interfacce, gli user control, etc., sono automaticamente inserite all'interno di un namespace, che per impostazione predefinita ha lo stesso nome del progetto.

Questa impostazione può essere cambiata facendo clic con il pulsante destro del mouse sul **nome del progetto** all'interno del Solution Explorer e selezionando il comando `Properties` (Proprietà).

Il codice evidenzia che anche una **Windows Form è una classe**, che eredita dalla classe «Form».

Osserviamo la parola chiave `partial`, una novità della versione 2.0 del Framework che permette di definire **classi “parziali”**.

Di norma, ogni classe risiede in un file separato, ma utilizzando `partial` è possibile creare **classi definite su più file**. Da ciascun file parziale è possibile accedere alle variabili, ai metodi, alle proprietà, di tutti gli altri file parziali relativi alla stessa classe, proprio come se l'intera dichiarazione della classe fosse contenuta in un unico file.

Questo nuovo concetto favorisce una maggiore pulizia e leggibilità del codice della Windows Form: il codice di progettazione, che imposta le proprietà della finestra e dei controlli in essa inseriti, in precedenza veniva salvato nello stesso file del form, all'interno della `region Windows Form Designer generated code` (Codice generato da Progettazione Windows Form), mentre ora è memorizzato in un file di supporto gestito internamente dall'ambiente di sviluppo e chiamato `Form1.Designer.cs` (al posto di `Form1` c'è il nome del form). Vedremo che questo concetto somiglia molto a quello di **code behind** nelle applicazioni Web.

L'unico indizio dell'esistenza di questo file è **il metodo `InitializeComponent()`** che viene richiamato nel costruttore del form: esso, come nelle precedenti versioni, ha il compito di inizializzare tutti gli oggetti contenuti nel form.

Per visualizzare il contenuto del file `Form1.Designer.cs`, premere il pulsante `+` posto a fianco del nome del form nel Solution Explorer.

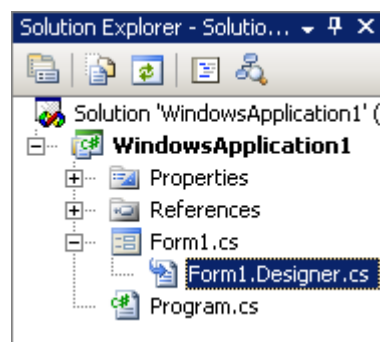


Figura 1. Il Solution Explorer

Le classi parziali e il loro utilizzo nelle Windows Form potrebbero sembrare novità di poco conto che, però, si fanno apprezzare quando si crea una finestra con un **numero elevato di controlli**.

Nelle precedenti versioni il file di definizione della Windows Form si reimpiva di una gran quantità di codice mantenuto dall'ambiente di sviluppo (che, quindi, l'utente non avrebbe dovuto modificare, come indicato chiaramente dai commenti), a discapito della leggibilità.

Visual Studio 2005 consente un **posizionamento dei controlli** sulla Windows Form molto più agevole che in passato: è sparita la griglia che ricopriva il form in fase di progettazione, poiché ora i controlli sembrano "calamitati" dai bordi della finestra e dagli altri oggetti; inoltre, durante le operazioni di trascinamento, vengono visualizzate delle **linee guida** per consentire un posizionamento più preciso degli elementi nella finestra.