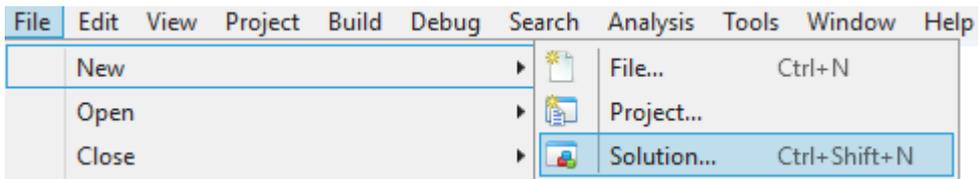
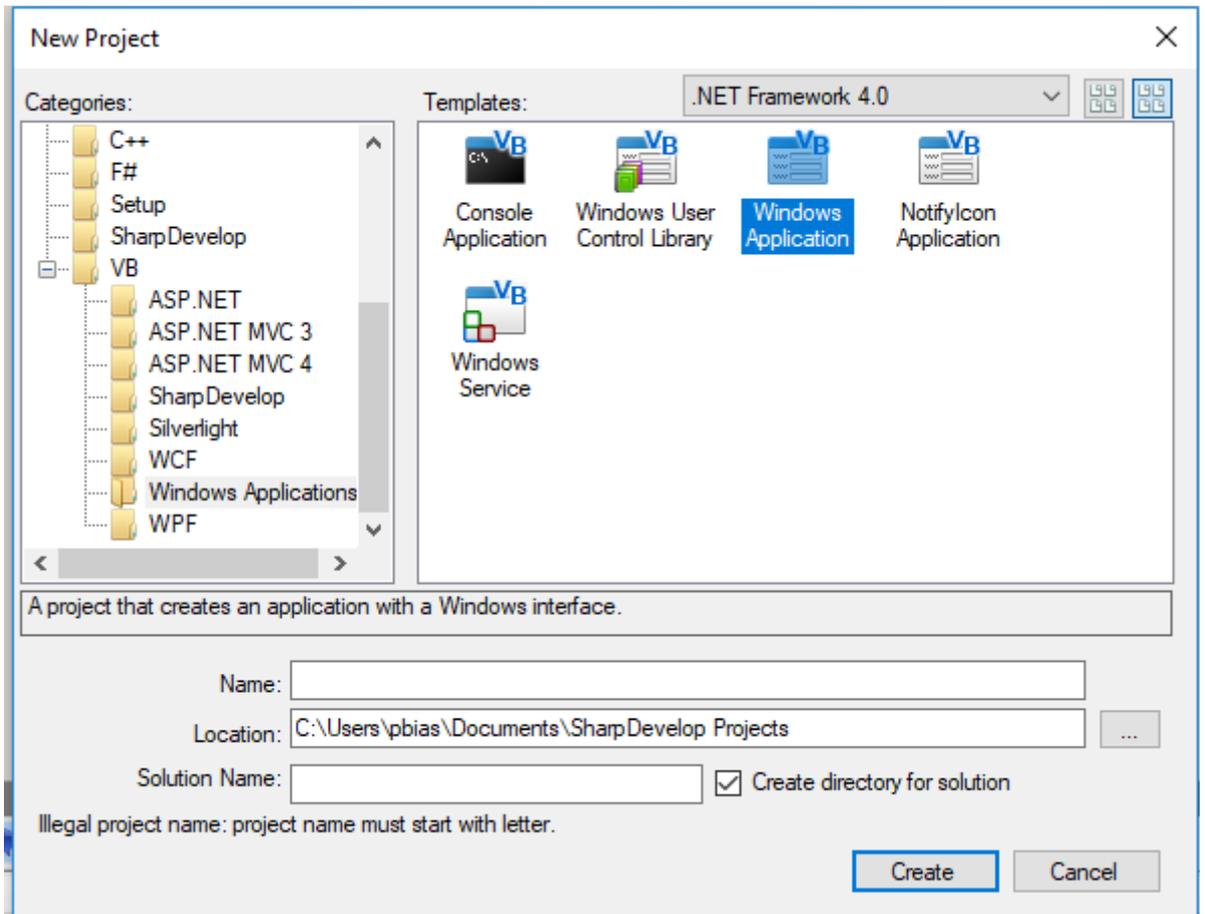


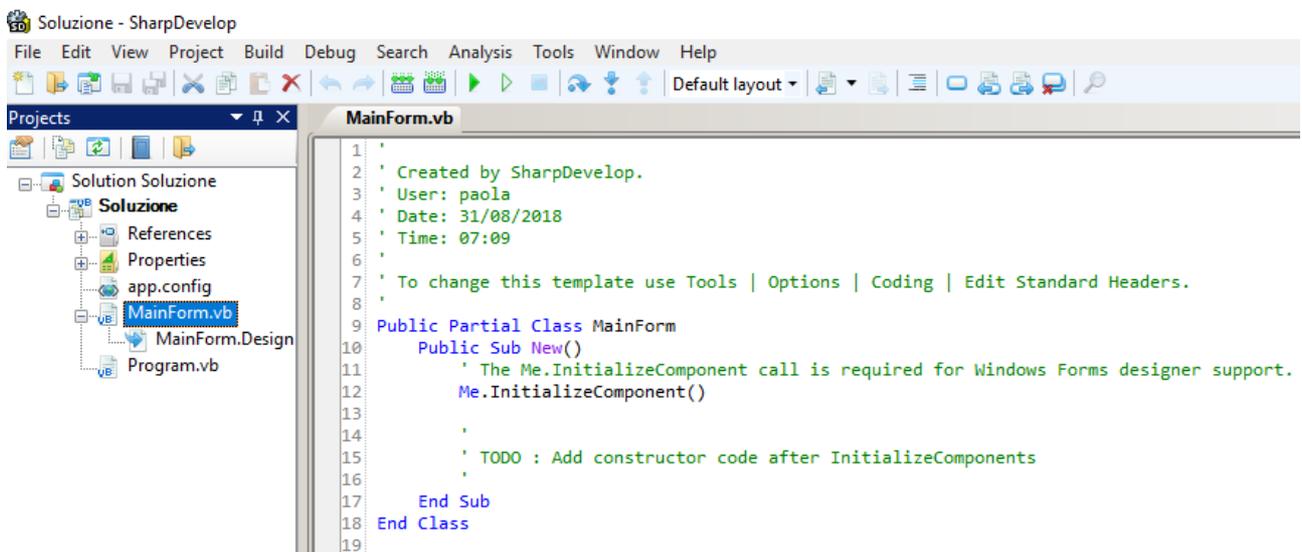
Per creare un **Progetto** selezionare **New > Solution**

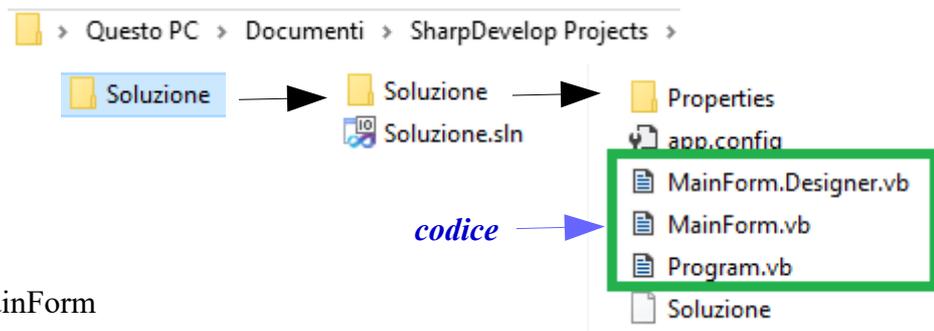


selezionare **VB**



scegliere tipo (ad esempio **Windows Application**) e un nome / salvataggio default in **SharpDevelop Projects**





Public Partial Class MainForm

Public Sub New()

' The Me.InitializeComponent call is required for Windows Forms designer support.

Me.InitializeComponent()

'

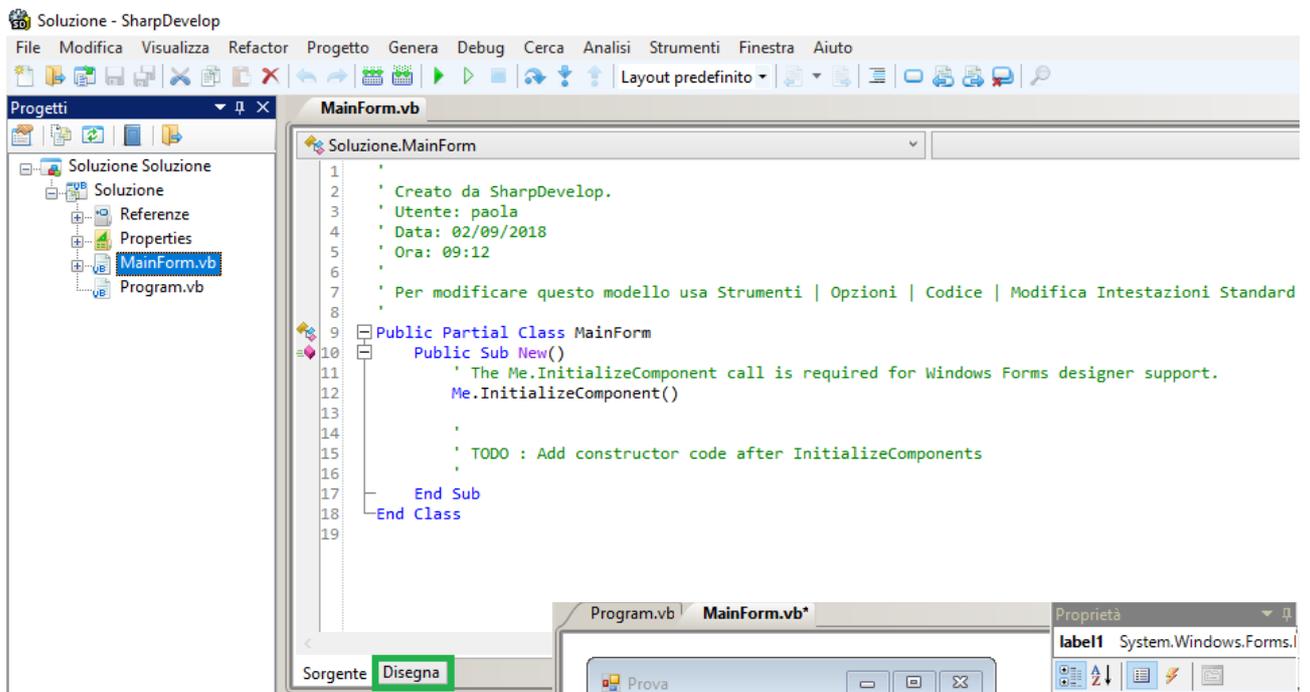
' TODO : Add constructor code after InitializeComponents

'

End Sub

End Class

Selezionando la visualizzazione **Disegna**

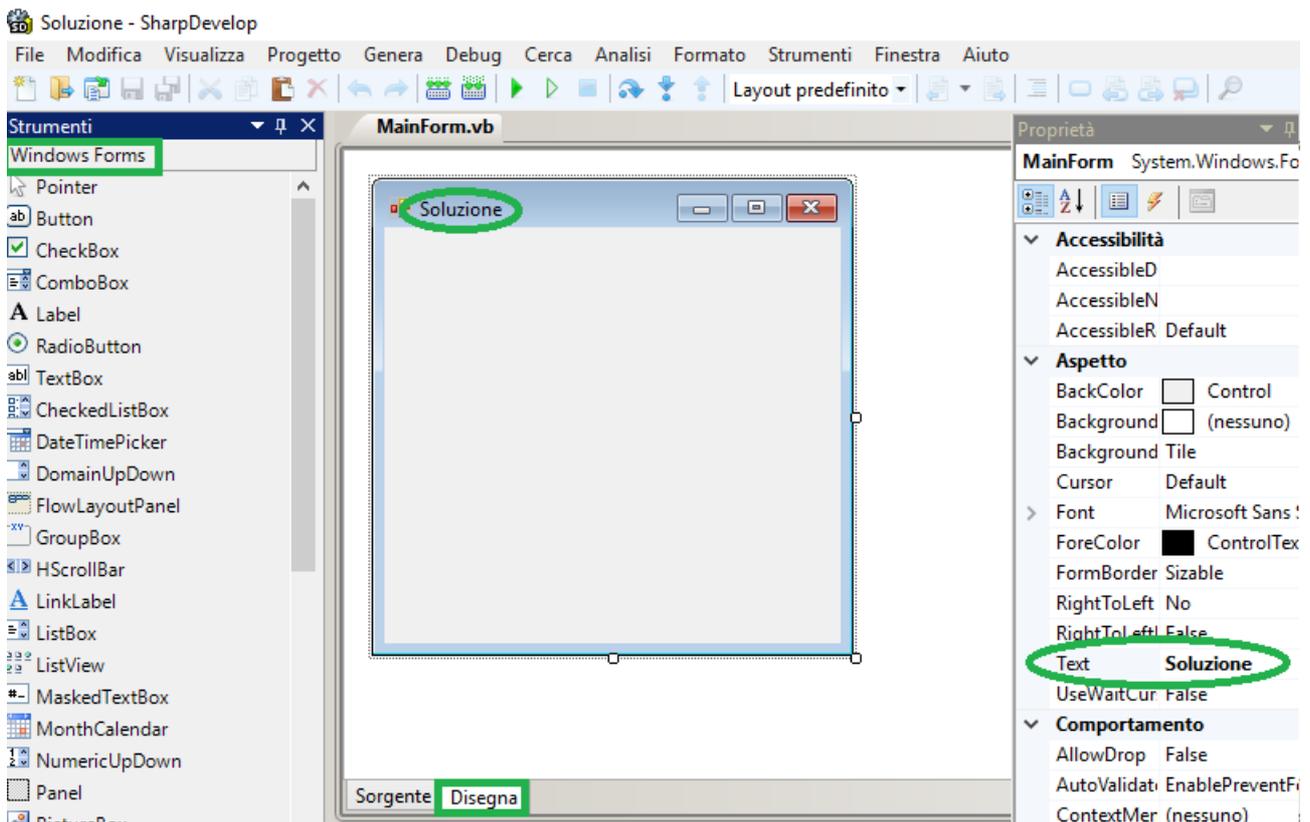


si può realizzare il layout voluto:

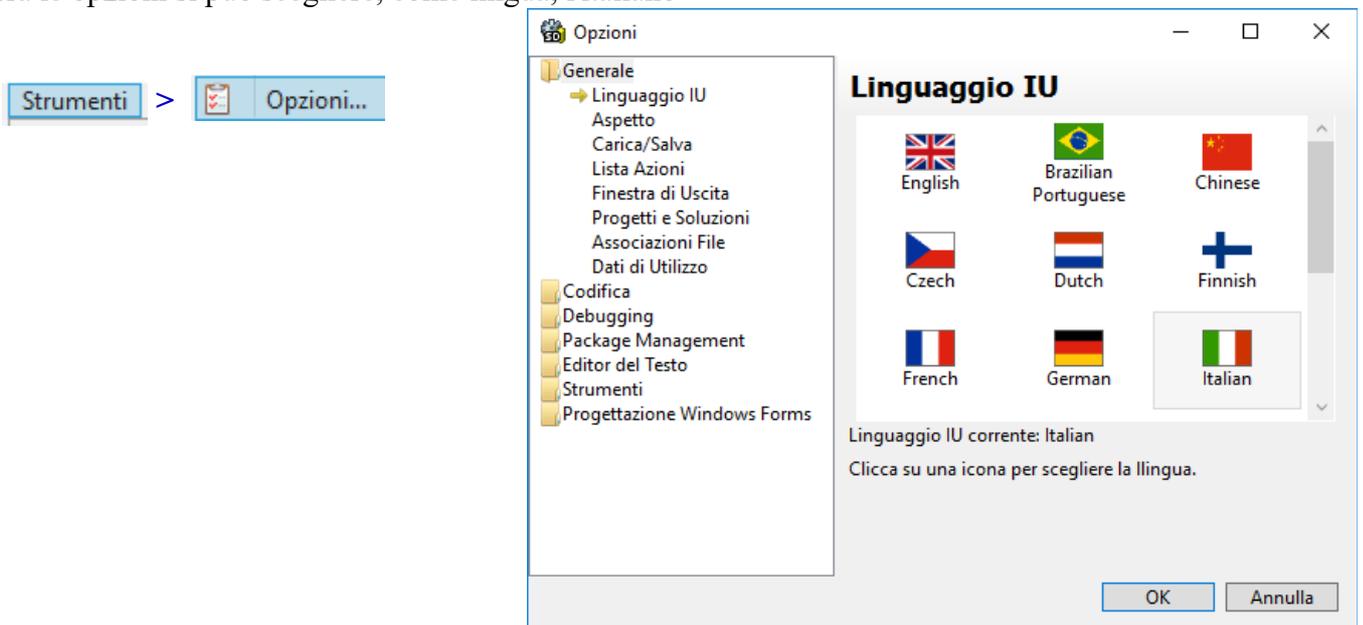
- inserire componenti
- modificarne le proprietà
- **gestire eventi**

NB: di default – indipendentemente dal nome del progetto – il nome del form è **MainForm.vb**

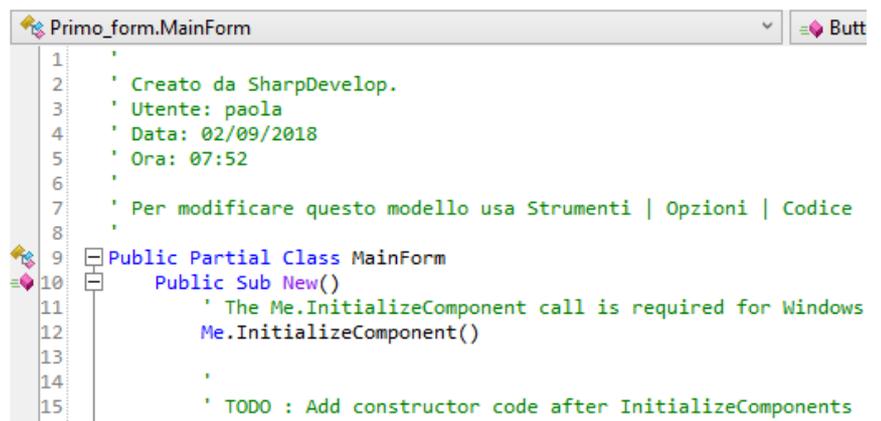
Definizione e proprietà dei componenti GUI sono salvati in **MainForm.Designer.vb**



Tra le opzioni si può scegliere, come lingua, l'italiano



ma solo alcuni commenti sono tradotti oltre al nome dei campi



Gestione eventi:

segmento in  MainForm.Designer.vb

```
MainForm.Designer.vb MainForm.vb
Primo_form.MainForm
35 Private Sub InitializeComponent()
36     Me.button1 = New System.Windows.Forms.Button()
37     Me.label1 = New System.Windows.Forms.Label()
38     Me.textBox1 = New System.Windows.Forms.TextBox()
39     Me.SuspendLayout
40     '
41     'button1
42     '
43     Me.button1.Location = New System.Drawing.Point(100, 145)
44     Me.button1.Name = "button1"
45     Me.button1.Size = New System.Drawing.Size(75, 23)
46     Me.button1.TabIndex = 0
47     Me.button1.Text = "Premi"
48     Me.button1.UseVisualStyleBackColor = true
49     AddHandler Me.button1.Click, AddressOf Me.Button1Click
50     '
51     'label1
52     '
53     Me.label1.Location = New System.Drawing.Point(69, 52)
54     Me.label1.Name = "label1"
55     Me.label1.Size = New System.Drawing.Size(100, 23)
56     Me.label1.TabIndex = 1
57     Me.label1.Text = "Digita testo"
```

*gestione evento
di click*

Disegno

modificando il nome del form

inserendo un **bottone**
un'etichetta
una casella di testo

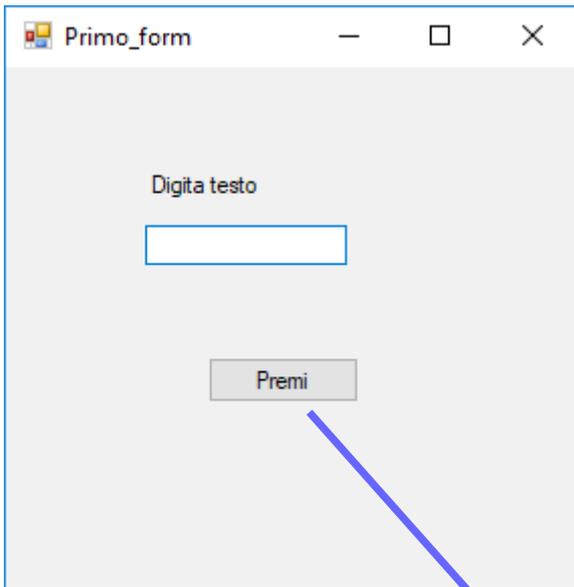
nb: un doppio click sul
componente inserito lo associa
all'evento di click

Sorgente

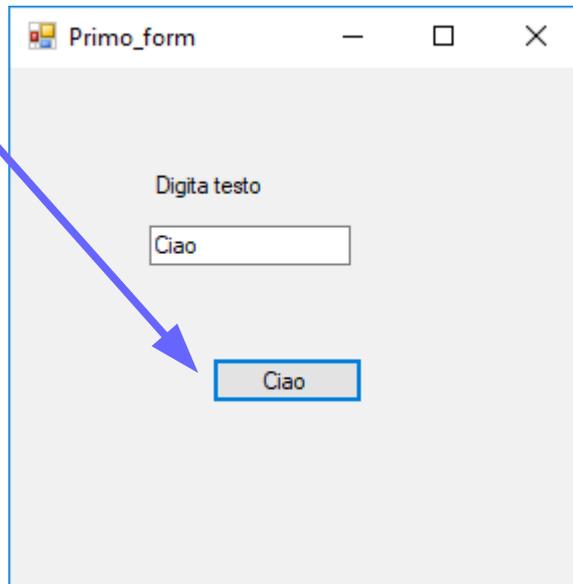
```
Primo_form.MainForm
10 Public Sub New()
11     ' The Me.InitializeComponent call is required for Windows Forms designer support.
12     Me.InitializeComponent()
13     '
14     '
15     ' TODO : Add constructor code after InitializeComponents
16     '
17     '
18 End Sub
19
20 Sub Button1Click(sender As Object, e As EventArgs)
21     button1.Text = TextBox1.Text
22
23 End Sub
24
25 End Class
```

handle code

Esegui Progetto



dopo aver digitato nella *casella di testo*
ad esempio il testo Ciao
premendo il *bottone*
si esegue l'*handle code*
cioè si modifica il testo del bottone stesso



aggiungendo altre istruzioni nell'*handle code*
ad esempio:

```
Sub Button1Click(sender As Object, e As EventArgs)
```

```
    button1.Text = TextBox1.Text
```

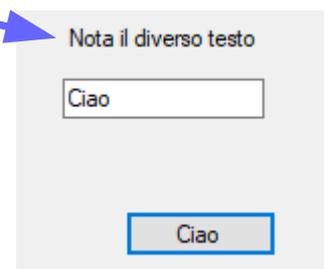
```
    label1.Text = "Nota il diverso testo sul bottone"
```

```
End Sub
```

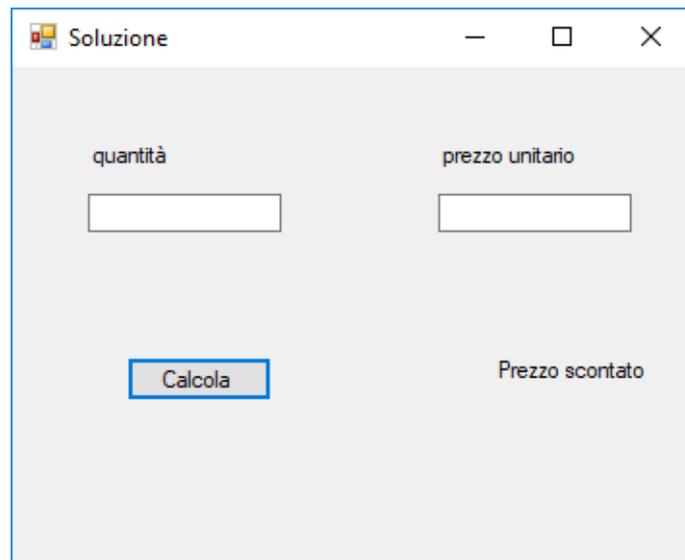
si ottiene il seguente risultato

se non si sono modificate le dimensioni dell'etichetta:

> Size 100; 23



Altro esempio

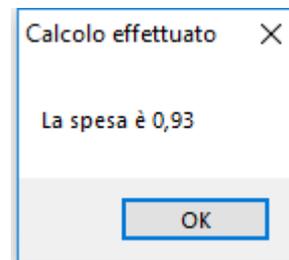
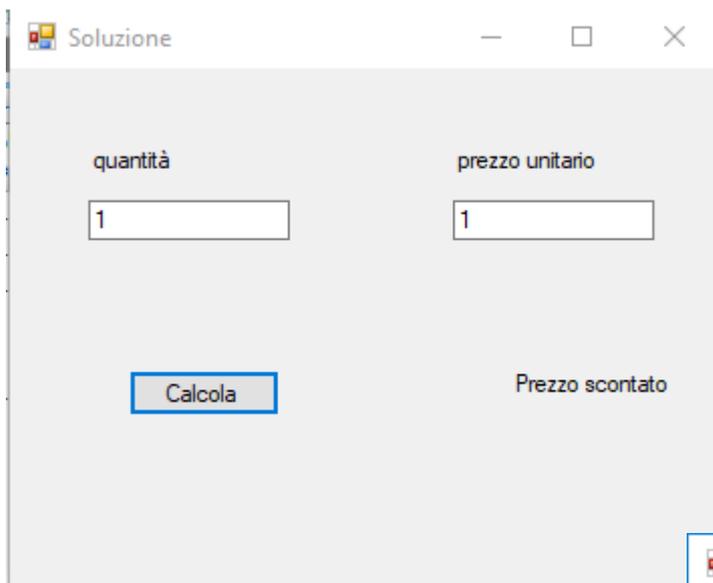


Sub Button1Click(sender As Object, e As EventArgs)

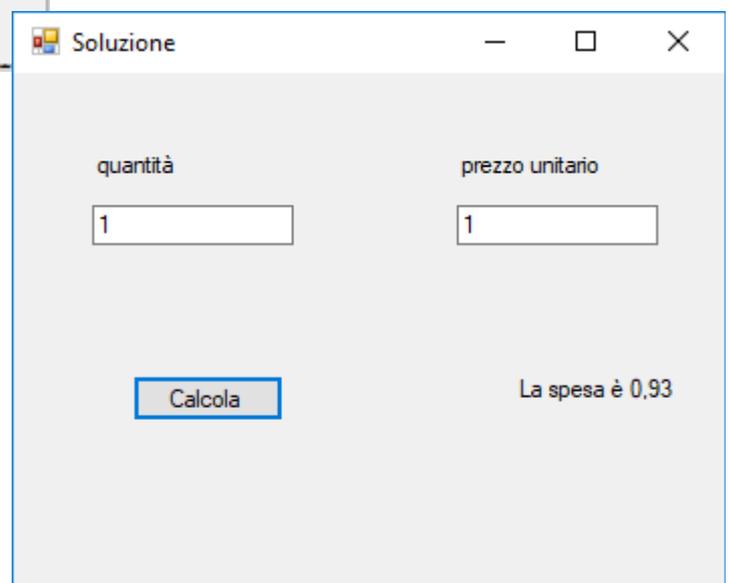
<<handle code >>

End Sub

dimenticando di inserire valori in input, forzati di default ad 1



alla chiusura della finestra modale



Handle code

```
Dim myValueq, myValuep As String
```

```
' controllo input
```

```
myValueq = textBox1.Text
```

```
If myValueq Is "" Then
```

```
    myValueq = "1"
```

```
    textBox1.Text = myValueq
```

```
End If
```

```
myValuep = textBox2.Text
```

```
If myValuep Is "" Then
```

```
    myValuep = "1"
```

```
    textBox2.Text = myValuep
```

```
End If
```

```
Dim q, sconto10, sconto7 As Integer, p, totale, spesa, soglia, sconto As Double 'oppure Single
```

```
soglia = 500.0
```

```
sconto10 = 10
```

```
sconto7 = 7
```

```
q = Int32.Parse(myValueq) ' conversione da String a Integer
```

```
p = Convert.ToDouble(myValuep)
```

```
totale = p * q
```

```
If totale > soglia Then
```

```
    sconto = sconto10
```

```
Else
```

```
    sconto = sconto7
```

```
End If
```

```
spesa = totale * (1 - sconto / 100)
```

```
Dim msg = "La spesa è " & spesa
```

```
Dim titolo = "Calcolo effettuato"
```

```
' Aggiungendo un titolo per visualizzare la finestra modale
```

```
MsgBox(msg, , titolo)
```

```
label3.Text = msg
```

inserendo corretti valori in input,

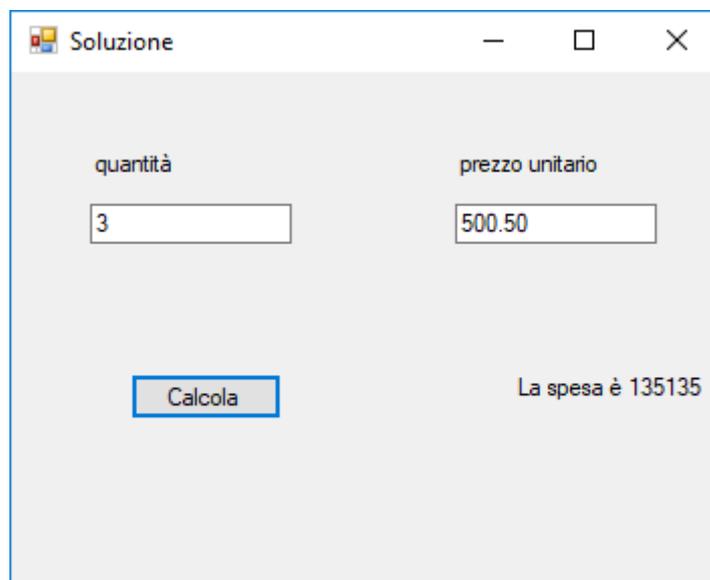
migliorare il progetto

modificando la posizione

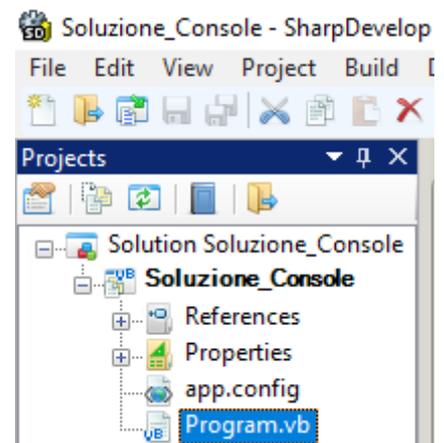
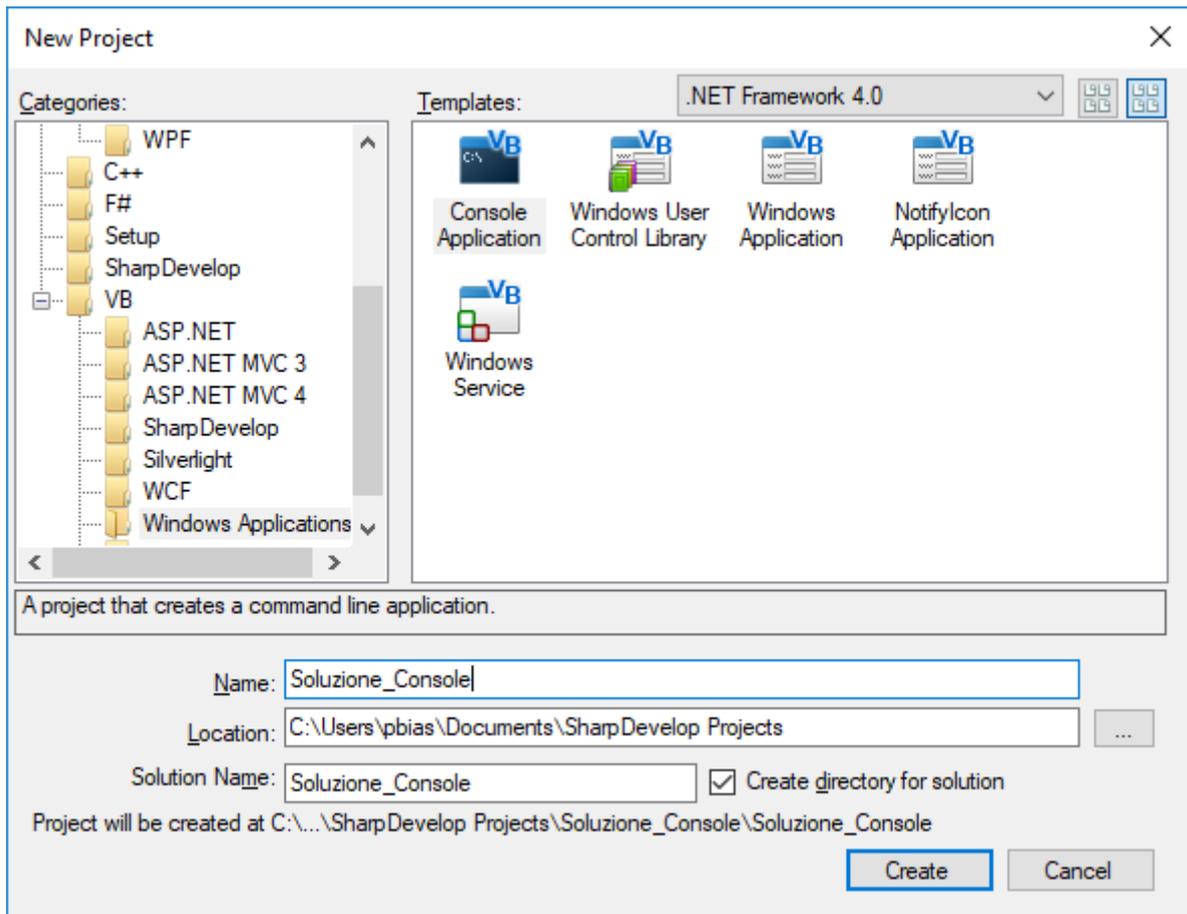
dell'etichetta risultato

oppure

cambiando il tipo di componente



scegliere tipo (ad esempio **Console Application**) e un nome / salvataggio default in  SharpDevelop Projects



' Created by SharpDevelop.

' User: paola

' Date: 31/08/2018

' Time: 14:33

,

' To change this template use Tools | Options | Coding | Edit Standard Headers.

,

Module Program

Sub Main()

Console.WriteLine("Hello World!") ' da eliminare ... automaticamente inserito per esempio

' TODO: Implement Functionality Here

<<codice aggiunto >>

Console.Write("Press any key to continue . . . ")

Console.ReadKey(True)

End Sub

End Module

codice aggiunto

```
Dim messageq, messagep, title, defaultValue As String
Dim myValueq, myValuep As Object

' personalizzazione e controllo input
' Set prompt
messageq = "Inserisci numero prodotti [un valore maggiore di 0]"
messagep = "Inserisci prezzo prodotto [un valore maggiore di 0]"

' Set title
title = "InputBox Personalizzata"
defaultValue = "1" ' Set default value

' Display message, title, and default value
myValueq = InputBox(messageq, title, defaultValue)
' If user has clicked Cancel, set myValue to defaultValue
If myValueq Is "" Then myValueq = defaultValue

' Display dialog box at position 100, 100
myValuep = InputBox(messagep, title, defaultValue, 100, 100)
' If user has clicked Cancel, set myValue to defaultValue
If myValuep Is "" Then myValuep = defaultValue

Dim q, sconto10, sconto7 As Integer, p, totale, spesa, soglia, sconto As Double 'oppure Single
soglia = 500.0
sconto10 = 10
sconto7 = 7

'q = myValueq      'warning conversione implicita da Object a Integer o Double
'p = myValuep

q = Int32.Parse(Convert.ToString(myValueq)) ' conversione da Object a String a Integer
p = Convert.ToDouble(myValuep)

totale = p * q

If totale > soglia Then
    sconto = sconto10
Else
    sconto = sconto7
End If

spesa = totale * (1 - sconto / 100)

Dim msg = "La spesa è " & spesa
Dim titolo = "Calcolo effettuato"

' Aggiungendo un titolo per visualizzare
MsgBox(msg, , titolo)
```



SharpDevelop

SharpDevelop è uno strumento **open source** sviluppato in C nelle prime versioni ed in C# nella versione più recente.

È compatibile con Visual Studio Express e Visual Studio dalla versione 2005, perché utilizza lo stesso tipo di formato per i file di progetto e il codice sorgente.

Altre caratteristiche interessanti sono:

- I linguaggi di programmazione supportati sono C (nelle prime versioni), C#, ASP.NET, ADO.NET, XML, HTML e, a partire dalla versione 3.0, F# e IronPython
- Evidenziazione delle sintassi per C (nelle prime versioni), C#, HTML, ASP, ASP.NET, VBScript, VB.NET e XML.
- Convertitore per C (nelle prime versioni), C#, verso VB.NET e viceversa (con le limitazioni: non converte progetti ma singoli elementi (file vb / cs) e viceversa; specie da vb a cs problemi nell'uso di array poichè non è in grado di sostituire le parentesi tonde con le quadre)
- Designer dei moduli per C (nelle prime versioni), C#, Visual Basic.NET e Boo.

[Guida ad una semplice Applicazione \(datata e altro esempio 2009\) o WFA](#)

[Guida pratica all'uso – Atlas](#)

[Prima applicazione \(inglese\)](#)

<http://www.icsharpcode.net/OpenSource/SD/Default.aspx>

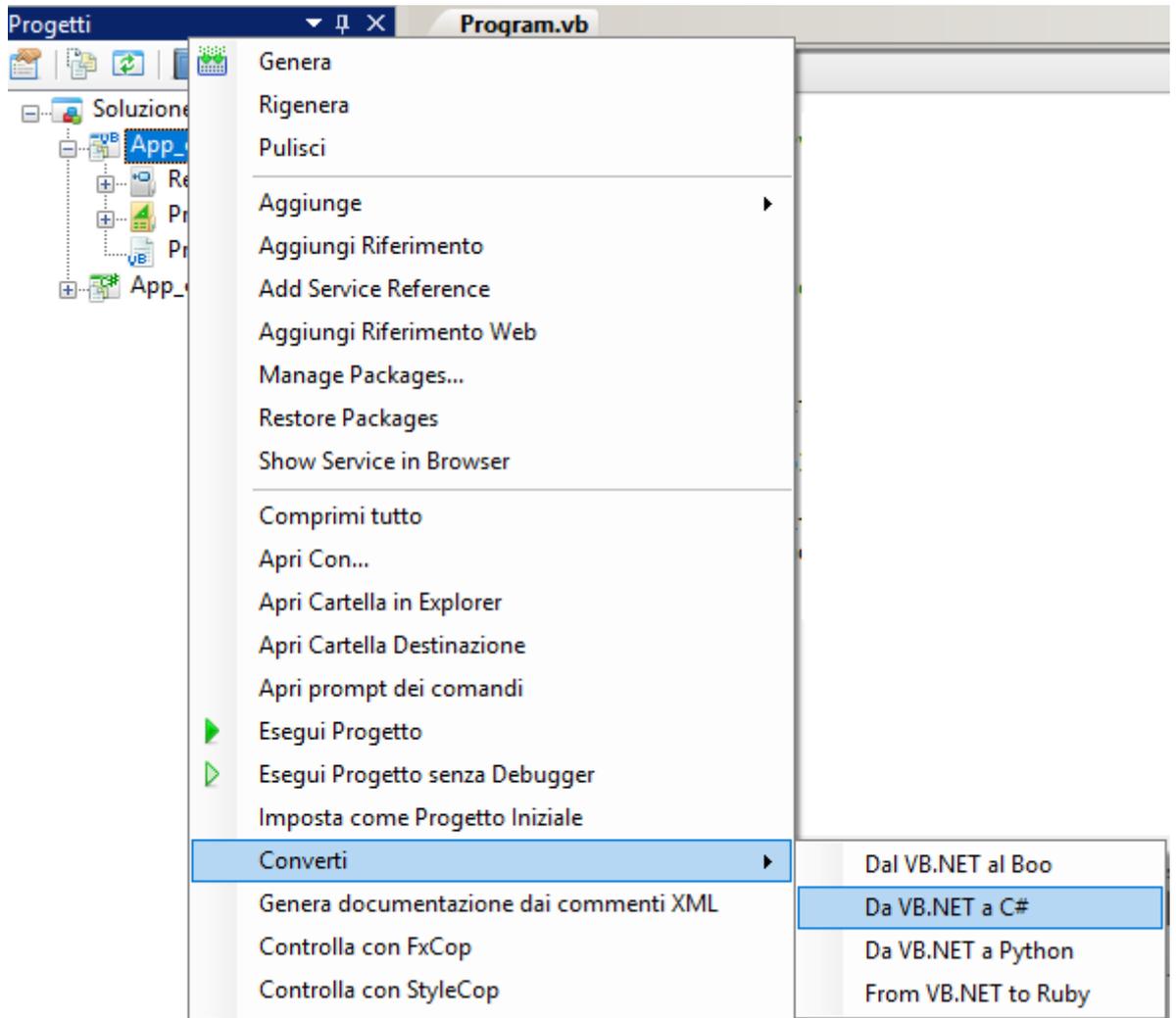
[Esempi \(C#\)](#)

[Dal C a VB](#)

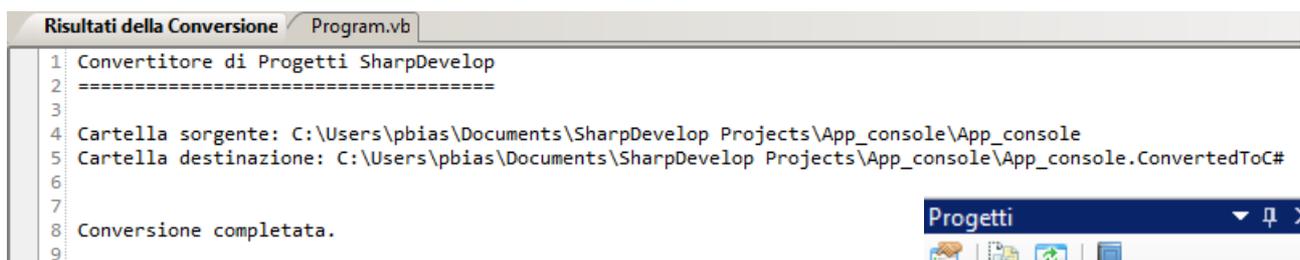


Per convertire:

- 1 aprire la soluzione
- 2 click con tasto destro, avendo selezionato il progetto cioè l'applicazione (nella finestra dei Progetti)



- 3 selezionare l'opzione di conversione cioè a quale linguaggio convertire



..... e viceversa:

