



## Premere un pulsante attraverso un altro form

[http://www.vbsimple.net/howto/ht\\_006.htm](http://www.vbsimple.net/howto/ht_006.htm)

Difficoltà: ► 1 / 5

L'operazione in questione è semplicissima, ma la sua soluzione, a volte, può essere misteriosa.

Supponiamo di avere due forms: **Form1** e **Form2**. All'interno di Form1 abbiamo un pulsante standard che effettua una determinata operazione.

Se volessimo far premere questo pulsante da una funzione del Form2?

In genere gli eventi di un oggetto sono di tipo *Private*, ovvero non possono essere chiamati da un'altra parte del progetto.

La soluzione più ovvia potrebbe sembrare quella di "*estendere*" il modificatore di accesso da *Private* a *Public*. Infatti, ricordiamo, una funzione *Public* può essere chiamata da qualunque punto del progetto.

Ma tale soluzione, seppur funzionante, è contro una delle fondamentali regole della buona [programmazione ad oggetti](#), l'*information hiding*.

La soluzione più pratica, e corretta, è quella di sfruttare la proprietà *Value* dei pulsanti di comando. Questa proprietà è di tipo booleano; la sua impostazione a *True* provoca la chiamata all'evento *Click* del pulsante.

In un nuovo progetto inseriamo due forms: **Form1** e **Form2**.

Inseriamo all'interno di Form1 un pulsante di nome **Pulsante1** ed un altro pulsante di nome **CaricaForm2**. Scriviamo queste semplici righe di codice:



```

1. Private Sub CaricaForm2_Click()
2.     Form2.Show vbModeless
3. End Sub
4.
5. Private Sub Pulsante1_Click()
6.     Beep
7.     MsgBox "Click!", vbOKOnly, "Prima finestra"
8. End Sub

```

La riga 2 serve a visualizzare il Form2 (che vedremo fra poco), dal quale chiameremo il pulsante di nome **Pulsante1**.

Alla riga 5 è dichiarata la funzione legata all'evento Click su **Pulsante1**.

Come si vede, essa è dichiarata *Private*; non potrebbe essere chiamata attraverso un altro form. All'interno d'essa abbiamo due semplici istruzioni: **Beep** che genera un effetto sonoro di avvertimento e **MsgBox** che mostra una semplice finestra con un messaggio.

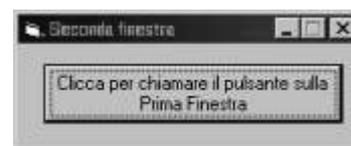
Com'è intuibile, al click sopra **Pulsante1** viene generato un effetto sonoro, seguito da una finestra di dialogo con il messaggio "*Click!*".

Passiamo ora al secondo form, dal quale chiameremo l'evento **Pulsante1\_Click**.

Il form si chiamerà **Form2** e conterrà soltanto un pulsante di nome **Pulsante2**.

Il codice di questo secondo form è semplicissimo:

```
1. Private Sub Pulsante2_Click()  
2.     Form1.Pulsante1.Value = True  
3. End Sub
```



Non abbiamo un riferimento diretto all'evento *Click* di **Pulsante1** sul primo form, perché sarebbe inaccessibile a causa del modificatore di accesso *Private*.

Abbiamo invece un'assegnazione alla proprietà *Value* di **Pulsante1**. La proprietà *Value*, subito dopo l'uscita dall'evento *Click* viene automaticamente reimpostata su *False*.

Proviamo il progetto?

Nel momento in cui cliccheremo sul primo pulsante del Form1 apparirà il messaggio di avvertimento. Se clicchiamo sopra il secondo, viene caricato e mostrato il Form2.

Quando cliccheremo sopra il pulsante del Form2 verrà invocato l'evento *Click* del primo pulsante sul Form1.

Si consiglia in ogni caso, dove non ci sono soluzioni alternative, di non cambiare i modificatori di accesso da *Private* a *Public*. Ciò va contro l'*information hiding*, base fondamentale della buona programmazione ad oggetti.

[Fibia FBI](#)

29 Novembre 2000



[Torna all'indice degli HowTo](#)

---