

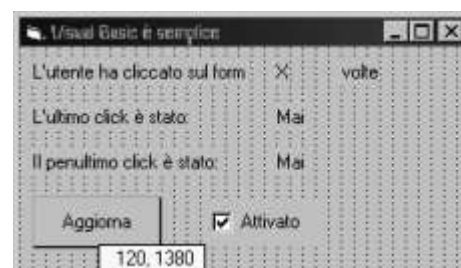


Corso base - Lezione 9

http://www.vbsimple.net/base/bas_09.htm

- [Oggetti contenitore e loro caratteristiche.](#)
- [Oggetto Frame.](#)
- [Inserimento di un controllo all'interno di un oggetto contenitore.](#)
- [Oggetto OptionButton.](#)



Come avrete sicuramente notato, nel momento in cui spostiamo un controllo sulla superficie del form, appare una casella con le sue coordinate attuali e man mano che l'oggetto viene spostato sono aggiornate le proprietà Top e Left dell'oggetto.



Ma quelle coordinate a cosa si riferiscono? Dove si trova l'origine 0,0 d'esse?

Le coordinate di ogni oggetto si riferiscono sempre al loro **oggetto contenitore**. Per esempio un pulsante avrà le coordinate relative al form dentro il quale il pulsante è inserito. L'angolo superiore sinistro del form avrà le coordinate 0,0.

Gli oggetti contenitore hanno la funzione di raggruppare più controlli all'interno d'essi. I controlli contenuti ereditano alcune caratteristiche e comportamenti dei loro contenitori; per esempio quando l'oggetto contenitore è disabilitato mediante la proprietà Enabled tutti i controlli all'interno d'esso non risponderanno agli eventi e non saranno quindi funzionanti. Altresì quando l'oggetto contenitore è nascosto mediante la proprietà Visible, con esso spariranno tutti i controlli contenuti.

Ma l'oggetto Form non è l'unico contenitore in Visual Basic. Tra i controlli standard Thunder abbiamo come contenitore anche gli oggetti **Frame**  e le **PictureBox**  che vedremo nel Corso Intermedio; intanto soffermiamoci sull'oggetto **Frame**.

Inseriamo subito un Frame sul nostro form. La prima cosa che salta all'occhio è che possiede un bordo attorno ed una scritta sulla sinistra. Spesso proprio per queste sue due caratteristiche viene utilizzato per raggruppare alcuni elementi visivi.

Vediamo come possiamo sfruttare un Frame per le sue caratteristiche di oggetto contenitore. Ingrandiamo un po' il nostro form e posizioniamo il nostro frame nel modo visibile nella figura qui a fianco. Chiamiamo il nostro frame ClickFrame.



Sorge ora il problema di portare le nostre labels all'interno del frame. Useremo il metodo più semplice: il *taglia e incolla*.

Selezioniamo la Label1 cliccando una volta sopra di essa. Ora, tenendo premuto il tasto CTRL clicchiamo una volta sopra le altre label, fino a selezionarle tutte e sette. Fatto questo, dal menu Modifica, selezioniamo la voce **Taglia**. Gli oggetti spariranno, ma rimarranno in memoria.

Clicchiamo ora una volta sopra il frame ClickFrame ed esso sarà evidenziato. In questo stato apriamo il menu Modifica e selezioniamo la voce **Incolla**. Verranno inseriti all'interno del frame tutti e 7 gli oggetti tagliati.

Si tratta ora di posizionarli per bene all'interno del frame. Aggiustare bene le coordinate spostando gli oggetti o uno per uno oppure tutti contemporaneamente mediante la selezione multipla vista prima.

Fatto questo spostiamo il frame alle coordinate 0,0 ovvero nell'angolo in alto a sinistra del form. Potremo ora rimettere il form alle dimensioni normali, ovvero come appare nella figura a fianco.

Il risultato di quest'operazione è un piacevole effetto grafico di raggruppamento di più oggetti.



Ma se non offrisse altre caratteristiche, il nostro frame si limiterebbe al solo raggruppamento grafico di oggetti.

Tuttavia gli oggetti contenitore forniscono qualche caratteristica in più come la possibilità di poter muovere contemporaneamente tutti gli oggetti contenuti all'interno del frame. Altresì possiamo nascondere o mostrare tutti gli oggetti con una sola operazione.

Per spostare uno o più controlli esistenti all'interno di un oggetto contenitore abbiamo usato il metodo del *Taglia/Incolla*. Per inserire un nuovo controllo all'interno di un oggetto contenitore possiamo semplicemente cliccare sull'icona del controllo dalla casella degli strumenti e tracciare un rettangolo con il mouse **sopra la superficie del frame**. In questo modo l'oggetto viene automaticamente inserito all'interno del contenitore.

Se volessimo, invece, *uscire* un controllo da un oggetto contenitore possiamo usare la stessa tecnica del *Taglia/Incolla*, ovvero selezioniamo i singoli controlli da portare fuori e cliccare sull'opzione Taglia del menu Modifica. Fatto questo clicchiamo **sulla superficie del form** e scegliamo la voce *Incolla* dal menu Modifica.

Ci si potrebbe aspettare che gli oggetti contenuti all'interno di un frame diventino [membri](#) d'esso. Invece via codice gli oggetti contenuti sono accessibili in maniera normale, del tutto estranea al loro contenitore.

Per esempio nel nostro caso ci si potrebbe aspettare che la Label1 contenuta all'interno del frame ClickFrame sia accessibile con un'istruzione del genere:

```
ClickFrame.Label1.Caption = "Ciao" ' Sintassi errata
```

Stranamente, invece non avviene così. La nostra Label1 sarà comunque membro dell'oggetto Form. Per cambiare la sua caption dobbiamo usare l'istruzione:

```
Me.Label1.Caption = "Ciao" ' Sintassi corretta
```

oppure, con il form implicito:

```
Label1.Caption = "Ciao" ' Anche questa è corretta
```

Gli oggetti contenitore *comandano* gli oggetti contenuti soltanto graficamente e non nella struttura del codice.

La caratteristica più importante degli oggetti contenitore si rivela con l'utilizzo degli **OptionButton** la cui funzione principale è quella di permettere la scelta di **una sola** opzione tra una serie di queste. Alla selezione di un'opzione tutte le altre presenti all'interno dello stesso contenitore saranno deselezionate automaticamente. Gli OptionButton sono talvolta denominati **pulsanti Radio**.

Vediamoli all'azione!

Inseriamo sul nostro form un altro Frame di nome FrameTitolo.

All'interno d'esso inseriamo due OptionButton: il primo di nome OptionOrario ed il secondo di nome OptionPIGreco. Impostiamo le Caption d'essi come appare nella figura. Il primo conterrà il testo **Orario** mentre il secondo avrà **PI Greco**.



La proprietà che identifica se un OptionButton è selezionato è la proprietà *Value*. Il valore True indica che il pulsante è selezionato.

Possiamo provare a impostare la proprietà Value di tutti e due gli OptionButton a True per vederne il funzionamento. Ci accorgeremo che non è possibile. Non appena un OptionButton riceve il valore True, automaticamente l'altro OptionButton riceverà il valore False deselezionandosi.

Apriamo la finestra del codice e prepariamo la routine di gestione dell'evento Click per entrambi i pulsanti di opzione. Inseriamo il seguente codice e vediamo il funzionamento:

```
1. Private Sub OptionOrario_Click()  
2.     Me.Caption = Now  
3. End Sub  
4.  
5. Private Sub OptionPIGreco_Click()  
6.     Me.Caption = PIGreco  
7. End Sub
```

Le istruzioni sono semplicissime: il click sul pulsante OptionOrario provoca il cambiamento della barra del titolo del form nell'ora corrente (istruzione Now). Il click sull'altro pulsante scrive sulla barra del titolo il valore della **costante** PIGreco.

Lanciamo il programma e vediamo gli effetti di quanto detto.

Al click di un pulsante di opzione verrà deselezionato l'altro.

Sebbene questo esempio funzioni nel modo previsto, si consiglia di raggruppare gli `OptionButton` in [matrici](#). La creazione di una matrice di controlli sarà affrontata nel corso Intermedio.

Tuttavia, se dovessimo inserire più serie di `OptionButton`, dovremmo utilizzare obbligatoriamente due contenitori diversi. Infatti il corretto funzionamento di tali pulsanti è determinato dalla posizione d'essi all'interno di un contenitore.

Potremmo inserire una serie di pulsanti di opzione sulla superficie del form e l'altra all'interno di un frame, oppure potremmo creare due frames e inserire all'interno di ognuno d'essi una serie di pulsanti di opzione.

Se inserissimo entrambe le serie di pulsanti all'interno di uno stesso frame, sarebbe impossibile selezionare contemporaneamente due diversi pulsanti.

[Fibia FBI](#)

14 Novembre 2000



[Torna all'ottava lezione](#)

[Vai alla decima lezione](#)

