Corso base - Lezione 10

http://www.vbsimple.net/base/bas_10.htm

- L'oggetto TextBox (casella di testo).
- Sue proprietà.

Un controllo fondamentale per comunicare con l'utente di un programma, sia in input che in output sono le ***TextBox*** ovvero le caselle di testo. La loro funzione principale è quella di permettere all'utente l'immissione di un valore che verrà elaborato in fase di esecuzione.

Le caselle di testo, per loro natura, non posseggono alcuna formula di controllo del loro contenuto; ciò significa che non è possibile definire se una casella di testo debba ricevere soltanto numeri, lettere, date o altri tipologie di dati. Ogni controllo di tal genere deve essere fatto da parte del programmatore mediante una funzione legata ad alcuni eventi dell'oggetto.

Le TextBox possiedono tante proprietà e solo pochi metodi e la maggior parte di questi ultimi riguarda funzionalità avanzate quale il trascinamento. In questo corso saranno soltanto accennate le proprietà principali di questo controllo e saranno tralasciate tutte le altre che, tuttavia saranno approfondite nella sezione Thunder.

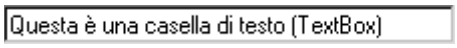

Figura 1

Se inseriamo sopra un form qualunque una *TextBox* ci apparirà quasi del tutto identica a quella mostrata nella figura 1, eccetto per il fatto che il testo contenuto in essa sarà differente. A differenza di tutti gli altri controlli che abbiamo visto nelle lezioni precedenti, il controllo *TextBox* non possiede una proprietà di nome **Caption** che determina il testo contenuto in esso. Però la soluzione sta subito dietro l'angolo: una proprietà quasi del tutto analoga alla citata Caption è la proprietà **Text**. Essa determina il testo presente all'interno della casella di testo.

Come quasi tutti gli altri controlli Thunder anche questo possiede le proprietà **Enabled** che determina l'attivazione dell'oggetto, **Top**, **Left**, **Width** e **Height** che regolano posizione e grandezza dell'oggetto.

Tuttavia, esistono alcune proprietà che sono particolari per il controllo TextBox; tra queste diamo uno sguardo alle più utili:

- **HideSelection**
  Nasconde la selezione quando l'oggetto perde lo stato attivo ovvero termina di essere l'oggetto attivo con il focus impostato.

 Copyright Visual Basic Simple 2002

- **Locked**
  Determina se il testo sia immodificabile.
  Si differenzia dalla proprietà Enabled per il fatto che la *TextBox* può ricevere il focus ed il cursore si può spostare attraverso il testo pur impedendone la modifica.

- **MaxLength**
  Determina la lunghezza massima in caratteri del testo che la casella di testo può contenere al suo interno.

- **MultiLine**
  Stabilisce se la casella di testo possa contenere più righe di testo.
  Questa proprietà è impostabile soltanto in fase di progettazione, mentre è in sola lettura in fase di esecuzione.

- **PasswordChar**
  Questa proprietà di tipo String*1 (ovvero può ricevere come valore una stringa lunga al massimo 1 carattere) quando impostata, sostituisce **solo visibilmente** tutti i caratteri del testo con il carattere specificato in essa.
  È utile nei casi si voglia consentire l'inserimento di una password, che non sia visibile ad altre persone; ogni volta che l'utente preme un tasto ed inserisce un carattere nella TextBox esso viene rimpiazzato con il carattere impostato in questa proprietà.
  Tuttavia i cambiamenti detti operano solo esteticamente, mentre la proprietà *Text* conterrà il testo in maniera corretta, come è stato inserito dall'utente.

- **SelStart**
  Imposta la posizione, a partire dal primo carattere, del cursore all'interno della casella di testo.

- **SelLength**
  Valore numerico che indica la lunghezza, in caratteri, della selezione a partire dalla posizione indicata dalla proprietà *SelStart*. Se tale valore è 0, nessuna parte del testo sarà selezionata, mentre per ogni valore maggiore di 0, saranno selezionati tanti caratteri a partire da *SelStart*.
  La visibilità della selezione è determinata comunque anche dalla proprietà *HideSelection*.

Estendiamo il nostro semplice programmino inserendo qualche controllo aggiuntivo:

1. Una *Label* A di nome **Label8** contenente il testo ***"Seleziona Testo -->"***.

2. Un *CommandButton* di nome **PulsanteSeleziona** con testo ***"Seleziona"*** posizionato subito accanto alla Label8.

3. Una *TextBox* ab| di nome **TestoLocked** con le proprietà così impostate:

 Copyright Visual Basic Simple 2002

- *Enabled*: **True**
- *Text*: **Prova a modificare questo testo**
- *Locked*: **True**
- *HideSelection*: **False**

4. Un'*OptionButton* di nome **OptionAltro** contenente il testo ***Altro***, posizionato all'interno del *Frame* **FrameTitolo**.

5. Una *TextBox* di nome **TestoTitolo**, con testo ***(testo)*** e proprietà *Enabled* **False**, posizionata anch'essa all'interno di **FrameTitolo**.

Vediamo a cosa dovrebbero servire questi controlli aggiunti?
Nel momento in cui l'utente cliccherà il pulsante ***Seleziona*** vedrà il testo della casella di testo sottostante in parte selezionato, sebbene il focus non sia su quell'oggetto, ma sul pulsante premuto. Questo per dimostrare la funzione delle proprietà *SelText, SelLength e HideSelection*.

L'altra funzione farà in modo che lo stato di selezione del pulsante ***Altro*** attivi e disattivi la casella ***(altro)***, rendendola accessibile o meno.
Nel momento in cui la casella è accessibile, qualunque cosa l'utente scriva all'interno d'essa apparirà immediatamente sulla barra del titolo del form. Questo per dimostrare la funzione dell'evento *Change*.

Apriamo la finestra del codice ed aggiungiamo le seguenti righe:

```
 1. Private Sub PulsanteSeleziona_Click()
 2.     TestoLocked.SelStart = 6
 3.     TestoLocked.SelLength = 7
 4. End Sub
 5.
 6. Private Sub TestoTitolo_Change()
 7.     Me.Caption = TestoTitolo.Text
 8. End Sub
 9.
10. Private Sub OptionAltro_Click()
11.     TestoTitolo.Enabled = True
12. End Sub
13.
```

Alla riga 1 definiamo la funzione che gestirà l'evento *Click* su **PulsanteSeleziona**.
All'interno d'essa (riga 2) abbiamo un'istruzione che porta il cursore del testo alla posizione 6 ovvero al sesto carattere. La riga seguente (3) imposta la lunghezza della selezione a 7 caratteri. Ciò significa che avremo selezionato i caratteri che vanno dal 6 al 13, ovvero il testo ***"a modif"***.

Alla riga 6 definiamo l'evento *Change* per la casella **TestoTitolo** ed all'interno della funzione abbiamo una semplice funzione che scrive sulla barra del titolo del form il contenuto della proprietà *Text* dell'oggetto TestoTitolo.

La riga 10 invece regola il *Click* sopra il pulsante **OptionAltro**.
L'esecuzione di questo evento imposta la proprietà *Enabled* della casella **TestoTitolo** a True, attivando di fatto la casella stessa.

 Copyright Visual Basic Simple 2002

Mentre, invece, andiamo a modificare le due funzioni *OptionOrario_Click* e *OptionPIGreco_Caption* come segue:

```
14. Private Sub OptionOrario_Click()
15.     TestoTitolo.Enabled = False
16.     Me.Caption = Now
17. End Sub
18.
19. Private Sub OptionPIGreco_Click()
20.     TestoTitolo.Enabled = False
21.     Me.Caption = PIGreco
22. End Sub
```

Nel momento in cui l'utente clicca sui pulsanti **OptionOrario** o **OptionPIGreco** la proprietà *Enabled* del controllo **TestoTitolo** sarà impostata su False (righe 15 e 20), disabilitando la casella di testo.

Quindi, nel momento in cui selezioniamo uno dei primi due pulsanti di opzione la casella di testo **TestoTitolo** sarà disattivata; quando clicchiamo il terzo pulsante essa sarà attivata.

Con la casella attivata l'utente potrà scrivere all'interno d'essa, ed ogni volta che viene digitato un carattere scatta l'evento *Change* e quindi le righe 6, 7 e 8 che cambiano il contenuto della *Caption* del form.

Proviamo il nostro progetto e vediamone in pratica gli effetti. Il suo utilizzo è molto semplice.

Figura 3

L'utilizzo delle caselle di testo si rende necessario quasi in ogni applicazione creata con Visual Basic poichè esse sono il mezzo più semplice per permettere all'utente l'inserimento di dati all'interno di un programma.

Fibia FBI
29 Dicembre 2000

Torna alla nona lezione Vai all'undicesima lezione

 Copyright Visual Basic Simple 2002