



## I controlli Label, TextBox e CommandButton

### ■ Il controllo Label

Il controllo Label serve per visualizzare un testo non modificabile dall'utente e viene utilizzato per descrivere altri controlli, ed è spesso usato per descrivere un TextBox o per visualizzare delle informazioni. Questo tipo di controllo deve essere inserito in un controllo UserForm e dato che è molto utilizzato, è inutile assegnare un nome significativo a ogni sua istanza, a meno che non la si debba modificare durante l'esecuzione del programma. Si utilizzano principalmente le seguenti proprietà: *Caption* per visualizzare il testo, *Left* e *Top* per posizionarlo, *TextAlign* per allineare il testo all'interno dell'etichetta, *Font* per lo stile e il formato del testo, *BackColor* e *ForeColor* per i colori di sfondo e del testo. È possibile formattare una etichetta sia nella finestra Proprietà o mediante un'istruzione VBA come: `Label1.Caption = "Inserire la quantità richiesta"`, oppure utilizzando una dichiarazione *With* come illustrato di seguito.

*Esempio:* Cliccando sul pulsante di una Form verrà formattato il testo della Label

```
Private Sub CommandButton1_Click()  
With Label1  
.Caption = "Inserire il testo"  
'Allineamento del testo impostato al centro  
.TextAlign = fmTextAlignCenter  
.WordWrap = True  
'Impostare il tipo di carattere  
.Font.Name = "Arial"  
.Font.Size = 12  
.Font.Italic = True  
'impostare a giallo e il colore del testo e a rosso il colore di sfondo  
.ForeColor = RGB(255, 255, 0)  
.BackColor = RGB(255, 0, 0)  
End With  
End Sub
```

### ■ Il Controllo CommandButton

Un *CommandButton* viene in genere utilizzato per eseguire una macro tramite l'evento Click. Se entrate in VBE è possibile accedere all'evento Click facendo doppio clic sul CommandButton, oppure è possibile selezionare il nome del CommandButton (nel modulo di codice per il Form utente) nella casella a discesa nella finestra del codice a sinistra e poi selezionare e poi cliccare nella tendina in alto a destra. Cliccando sul CommandButton verrà eseguito il codice che viene inserito nell'evento *Click* oppure è possibile inserire il nome di una macro da eseguire

*Esempio:* Usando l'evento Click si può chiudere la Form

```
Private Sub CommandButton2_Click ()  
MsgBox "Chiusura UserForm !"  
Unload Me  
End Sub
```



*Esempio:* Cliccando sul pulsante di comando si esegue un'altra macro per svuotare tutti i controlli della Form

```
Private Sub CommandButton1_Click ()  
ClearForm  
End Sub
```

```
Private Sub ClearForm ()  
Textbox1.value = ""  
ComboBox1.Value = ""  
CheckBox1.Value = False  
OptionButton1.Value = False  
End Sub
```

## ■ Il Controllo TextBox

Un controllo TextBox accetta dati da parte dell'utente. Oltre alle proprietà comuni menzionate in precedenza, le proprietà chiave includono:

**Proprietà AutoTab:** Restituisce un valore booleano (True o False) che specifica se la scheda passa automaticamente al controllo successivo nell'ordine di tabulazione dopo il numero massimo di caratteri determinato dalla struttura *MaxLength* all'entrata nel TextBox da parte dell'utente, mentre invece il valore False (che è di default) indica che lo spostamento al controllo successivo avviene manualmente quando l'utente preme il tasto Tab.

**Proprietà EnterKeyBehavior:** Restituisce un valore booleano (True o False) che determina l'effetto quando un utente preme il tasto Invio in un TextBox. Se la proprietà *MultiLine* è impostata su True, il valore indica la creazione di una nuova linea premendo Invio mentre il valore False (Default) si passa al controllo successivo nell'ordine di tabulazione. Se *MultiLine* proprietà è impostata su False, il Focus è sempre spostato al controllo successivo nell'ordine di tabulazione ignorando la proprietà *EnterKeyBehavior*

**Proprietà MaxLength:** Specifica il numero massimo di caratteri che possono essere inseriti in una TextBox. Specificando un valore di 0 indica che non vi è alcun limite massimo.

**Proprietà MultiLine:** Restituisce un valore booleano (True o False) che determina se il testo viene visualizzato in più righe o meno, nella TextBox. True indica che il testo viene visualizzato in più righe, e questo è anche il valore di default.

**Proprietà PasswordChar:** Specifica quali caratteri si devono visualizzare nella TextBox al posto dei caratteri effettivamente inseriti o digitati dall'utente. Questa proprietà è utile per proteggere i codici di sicurezza sensibili, o per convalidare un utente prima di consentire di procedere ulteriormente.

**Proprietà ScrollBars:** Specifica se un TextBox ha barre di scorrimento verticali e/o orizzontali, o nessuna. Ci sono 4 impostazioni "auto-esplicativo":

- *fmScrollBarsNone* (valore 0) - questa è l'impostazione di default
- *fmScrollBarsHorizontal* (valore 1)
- *fmScrollBarsVertical* (valore 2)
- *fmScrollBarsBoth* (valore 3).

L'impostazione *fmScrollBarsNone* non visualizza nessuna barra di scorrimento e se la proprietà *AutoSize* è impostata su True, non comparirà nessuna barra di scorrimento perché il TextBox si allarga per accogliere il testo o di dati aggiuntivi.



Se *WordWrap* è impostata su True, non comparirà nessuna barra di scorrimento orizzontale nel TextBox. Le impostazioni della barra di scorrimento *fmScrollBarHorizontal* o *fmScrollBarBoth*, visualizzano una barra di scorrimento orizzontale in una TextBox a linea singola se il testo è più lungo rispetto alla capienza della casella. Le impostazioni della barra di scorrimento *fmScrollBarVertical* o *fmScrollBarBoth*, visualizzano una barra di scorrimento verticale in una TextBox a più righe se il testo è più lungo rispetto alla capienza della casella e *WordWrap* è impostata su True.

Per visualizzare una barra di scorrimento orizzontale in un TextBox multilinea, l'impostazione della barra di scorrimento dovrebbe essere *fmScrollBarHorizontal*, e *WordWrap* dovrebbe essere impostata su False e il testo dovrebbe essere più lungo rispetto alla capienza della casella.



*Una barra di scorrimento orizzontale (o verticale) è visibile solo se il controllo ha spazio sufficiente per includere la barra di scorrimento sotto o al bordo destro della sua casella.*

**Proprietà Text:** Il testo in un TextBox viene restituito o impostato da questa proprietà. Un valore assegnato alla proprietà Text viene assegnato automaticamente al valore della proprietà, e viceversa.

**Esempio:** Aggiungere un TextBox in un Form e formattarlo utilizzando il codice VBA

```
Private Sub CommandButton1_Click()  
Dim box_1 As MSForms.TextBox  
Set box_1 = Controls.Add("Forms.TextBox.1", "Esempio 1")  
With box_1  
    .Font.Name = "Times New Roman"  
    .Font.Size = 10  
    .TextAlign = fmTextAlignLeft  
    .Width = 100  
    .Height = 50  
    .Left = 50  
    .Top = 75  
    .MultiLine = True  
    .WordWrap = True  
    .AutoSize = False  
    .ScrollBars = 2  
    .SetFocus  
End With  
End Sub
```

**Esempio:** Impostare la proprietà Enabled di una TextBox per impedire all'utente di digitare direttamente nella casella di testo che deve essere riempito solo per l'opzione selezionata dall'utente nel ListBox

```
Private Sub UserForm_Initialize()  
With ListBox1  
    For i = 1 To 10  
        .AddItem i  
    Next i  
    .ControlTipText = "Selezionare un numero dal Listbox per inserirlo nel TextBox."  
End With  
Me.TextBox1.Enabled = False  
End Sub  
Private Sub ListBox1_Click()  
    TextBox1.Text = ListBox1.Value  
End Sub
```



*Esempio:* Utilizzare una TextBox per impostare una password, in questo esempio usiamo la proprietà *PasswordChar* per controllare se il nome utente e la password, sono autorizzati a procedere e caricare UserForm1 viene caricato.

```
Private Sub UserForm_Initialize()  
    TextBox2.MaxLength = 5  
    TextBox2.PasswordChar = "*"   
    TextBox2.BackColor = RGB(255, 255, 0)  
End Sub  
  
Private Sub CommandButton1_Click()  
    Dim password As String  
    If TextBox1.Text = "Pippo" And TextBox2.Text = "123456" Then  
        password = "True"  
    ElseIf TextBox1.Text = "Topolino" And TextBox2.Text = "987654" Then  
        password = "True"  
    ElseIf TextBox1.Text = "Paperino" And TextBox2.Text = "369852" Then  
        password = "True"  
    End If  
    If password = "True" Then  
        MsgBox "Password Esatta Puoi Continuare"  
        Unload Me  
        UserForm1.Show  
    Else  
        MsgBox "Paasword o Username errato. Riprova"  
        TextBox1.Text = vbNullString  
        TextBox2.Text = vbNullString  
        TextBox1.SetFocus  
    End If  
End Sub
```