



## Corso Intermedio - Lezione 9

[http://www.vbsimple.net/intermed/int\\_09.htm](http://www.vbsimple.net/intermed/int_09.htm)

- [Gestione dei pulsanti.](#)
- [Aggiunta di un nuovo brano.](#)
- [Modifica di un brano preesistente.](#)

Il progetto fino a questa lezione sviluppato si limita alle interazioni semplici con l'utente: reagire ai click sui pulsanti o alla selezione di un brano dall'elenco sulla sinistra: introdurremo in questa lezione una primaria gestione dei pulsanti ovvero stati e azioni legate ai pulsanti in funzione del contesto. In particolare il pulsante **Modifica brano** effettuerà una triplice azione:

1. Abilitarsi alla selezione di un brano e disabilitarsi alla cancellazione del brano scelto;
2. Attivazione della fase di modifica per permettere il cambiamento dei dati presenti nelle caselle di testo;
3. Accettazione dei dati immessi e quindi salvataggio degli stessi nell'archivio.

La prima delle azioni è naturalmente la più semplice; vediamo subito come è gestita:

```

1. Private blnModifica As Boolean
2.
3. Private Sub Form_Load()
4.     ... Codice trattato nelle lezioni precedenti ...
5.     cmdModificaBrano.Enabled = False
6.     cmdEliminaBrano.Enabled = False
7.     blnModifica = False
8.     ... Codice trattato nelle lezioni precedenti ...
9. End Sub
10.

```

Dichiareremo in testa al codice del form una variabile di nome **blnModifica** che verrà utilizzata successivamente per identificare la fase di modifica, compresa tra il click sul pulsante modifica ed il successivo salvataggio.

All'interno dell'evento Load del Form - senza eliminare il codice precedentemente esposto - aggiungeremo le righe 5-7; la prima disattiverà inizialmente il pulsante **cmdModifica**, mentre la seconda di queste era stata eliminata nella lezione precedente e disattiverà il pulsante di eliminazione all'avvio del programma, infine la riga 7 inizierà il valore della variabile **blnModifica**.

```

11. Private Sub lstBrani_Click()

```

```

12.   If lstBrani.ListIndex >= 0 Then
13.       ... Codice trattato nelle lezioni precedenti ...
14.       cmdEliminaBrano.Enabled = True
15.       cmdModificaBrano.Enabled = True
16.   End If
17. End Sub
18.

```

Così alla selezione di un brano dall'elenco riattiveremo i due pulsanti precedentemente disattivati.

```

18. Private Sub cmdEliminaBrano_Click()
19.   If lstBrani.ListIndex >= 0 Then
20.       If MsgBox("Sei sicuro di voler eliminare il brano" _
21.           & vbCrLf & Trim$(udtDiscoCorrente.strTitoloBrano) _
22.           & "?", vbCritical Or vbYesNo Or vbDefaultButton2, _
23.           "Elimina brano") = vbYes Then
24.           udtDiscoCorrente.blnEliminato = True
25.           Put #intFileDati, udtDiscoCorrente.intIndice, udtDiscoCorrente
26.           lstBrani.RemoveItem lstBrani.ListIndex
27.           cmdEliminaBrano.Enabled = False
28.           cmdModificaBrano.Enabled = False
29.       End If
30.   End If
31. End Sub
32.

```

L'azione del pulsante **cmdEliminaBrano** è stata già trattata nella lezione precedente; ad essa aggiungeremo due semplici righe che si occuperanno di disattivare i due pulsanti al termine della cancellazione del brano selezionato. Si occuperà di riattivarli la funzione precedentemente vista legata alla selezione di un brano.

Resta da trattare la fase di modifica che sarà richiamata in occasione del primo click sul pulsante *Aggiungi* oppure sul pulsante *Modifica*; nel primo caso saranno anche svuotate tutte le caselle di testo. Vediamo inizialmente il primo di questi due casi: l'operazione di inserimento, aggiornata con le ultime modifiche.

```

33. Private Sub cmdAggiungiBrano_Click()
34.   If blnModifica = False Then
35.       txtTitoloBrano.Text = ""
36.       txtAutoreBrano.Text = ""
37.       txtTitoloCD.Text = ""
38.       txtNumeroTraccia.Text = ""
39.       txtAnnoPubblicazione.Text = ""
40.       txtNote.Text = ""
41.       cmdAggiungiBrano.Caption = "S&alva brano"
42.       txtTitoloBrano.SetFocus
43.   Else
44.       With udtDiscoCorrente
45.           .intIndice = LOF(intFileDati) / Len(udtDiscoCorrente) + 1
46.           .strTitoloBrano = txtTitoloBrano.Text
47.           .strAutore = txtAutoreBrano.Text
48.           .strTitoloCD = txtTitoloCD.Text
49.           .intTraccia = CInt(txtNumeroTraccia.Text)
50.           .intAnnoPubblicazione = CInt(txtAnnoPubblicazione.Text)
51.           .strNote = txtNote.Text
52.           .blnEliminato = False
53.           Put #intFileDati, .intIndice, udtDiscoCorrente
54.           lstBrani.AddItem Trim$(.strTitoloBrano)
55.           lstBrani.ItemData(lstBrani.NewIndex) = .intIndice
56.       End With
57.       cmdAggiungiBrano.Caption = "&Aggiungi brano"
58.   End If
59.   blnModifica = Not blnModifica
60.   txtTitoloBrano.Locked = Not blnModifica

```

```

61.     txtAutoreBrano.Locked = Not blnModifica
62.     txtTitoloCD.Locked = Not blnModifica
63.     txtNumeroTraccia.Locked = Not blnModifica
64.     txtAnnoPubblicazione.Locked = Not blnModifica
65.     txtNote.Locked = Not blnModifica
66.     cmdEliminaBrano.Enabled = Not blnModifica
67.     cmdModificaBrano.Enabled = Not blnModifica
68.     lstBrani.Enabled = Not blnModifica
69. End Sub
70.

```

Un primo click sopra il pulsante **Aggiungi** farà scattare la fase di modifica; basti ricordare che inizialmente la variabile **blnModifica** ha assunto valore Falso (riga 7) mentre cambia di valore nel momento in cui viene avviata la fase di modifica. Così, all'avvio della fase di modifica tutti i campi saranno svuotati (righe 35-40) e lo stesso pulsante cambierà la sua etichetta in *Salva brano*.

Nel caso in cui fossimo già nella fase di modifica, il click sul pulsante Aggiungi farà scattare l'operazione di salvataggio vera e propria già trattata nelle lezioni precedenti. In maniera analoga, sarà modificata l'etichetta del pulsante in *Aggiunti brano*. Le righe 54 e 55 sono state aggiunte al codice preesistente per tenere l'elenco **lstBrani** sempre aggiornato, senza la necessità di chiudere il programma e riavviarlo come nelle lezioni precedenti.

Ogni click sul pulsante Aggiungi invertirà il valore della variabile **blnModifica** (riga 59) ed i controlli si adatteranno di conseguenza: le caselle di testo saranno sbloccate (righe 60-65), mentre i pulsanti Elimina e Modifica e l'elenco con i brani verranno bloccati all'avvio dell'operazione di inserimento. Al termine lo stato di questi controlli verrà invertito, in funzione del cambiamento del valore di **blnModifica**.

La gestione del pulsante Modifica è molto simile alla precedente:

```

71. Private Sub cmdModificaBrano_Click()
72.     If blnModifica = False Then
73.         cmdModificaBrano.Caption = "Salva &Modifiche"
74.         txtTitoloBrano.SetFocus
75.     Else
76.         With udtDiscoCorrente
77.             .strTitoloBrano = txtTitoloBrano.Text
78.             .strAutore = txtAutoreBrano.Text
79.             .strTitoloCD = txtTitoloCD.Text
80.             .intTraccia = Int(Val(txtNumeroTraccia.Text))
81.             .intAnnoPubblicazione = Int(Val(txtAnnoPubblicazione.Text))
82.             .strNote = txtNote.Text
83.             .blnEliminato = False
84.             Put #intFileDati, .intIndice, udtDiscoCorrente
85.             lstBrani.List(lstBrani.ListIndex) = Trim$(.strTitoloBrano)
86.         End With
87.         cmdModificaBrano.Caption = "&Modifica brano"
88.     End If
89.     blnModifica = Not blnModifica
90.     txtTitoloBrano.Locked = Not blnModifica
91.     txtAutoreBrano.Locked = Not blnModifica
92.     txtTitoloCD.Locked = Not blnModifica
93.     txtNumeroTraccia.Locked = Not blnModifica
94.     txtAnnoPubblicazione.Locked = Not blnModifica
95.     txtNote.Locked = Not blnModifica
96.     cmdEliminaBrano.Enabled = Not blnModifica
97.     cmdAggiungiBrano.Enabled = Not blnModifica
98.     lstBrani.Enabled = Not blnModifica
99. End Sub

```

All'avvio della fase di modifica (riga 72) i vari controlli con le informazioni saranno già stati aggiornati, durante la selezione del brano dall'elenco; resta quindi solo da cambiare l'etichetta del pulsante in *Salva Modifiche*. Al termine dell'operazione di modifica i dati delle varie caselle saranno ricopiati nella variabile **udtDiscoCorrente** (righe 76-83) che, lo ricordiamo, contiene le informazioni recuperate dal file con l'ultima estrazione. La stessa sarà salvata all'interno del file con una semplice istruzione Put, accompagnata dal numero del record da scrivere (riga 84). Sarà anche aggiornato l'elenco sulla sinistra con il nuovo titolo del brano. Il pulsante di modifica riprenderà la sua etichetta iniziale *Modifica brano*.

Come già visto nell'evento precedente, in funzione della fase in cui ci troviamo (modifica o salvataggio) i controlli saranno bloccati o sbloccati.

Poiché sarà compito dei pulsanti gestire lo stato di blocco delle varie caselle di testo in funzione della fase in cui ci si trova, sarà opportuno inizializzare lo stato di blocco di tutte le caselle di testo impostando la proprietà **Locked** su False durante la fase di progettazione.

Questa lezione si conclude qui; il codice sin qui presentato è davvero molto semplice e si limita a pochissime interazioni, tutte legate agli eventi click dei controlli. Con le prossime lezioni cercheremo di affrontare argomenti un po' più complessi, iniziando dalla gestione dei dati in input e dalla gestione degli errori.

[Fibia.FBI](#)

26 Novembre 2003



[Torna all'ottava lezione](#)

[Vai alla decima lezione](#)

