Effettuare una pausa all'interno di un progetto

http://www.vbsimple.net/howto/ht_034.htm

Sono davvero tante le occasioni nelle quali si rende necessario effettuare una pausa all'interno di un progetto. La stragrande maggioranza dei casi riguarda la sincronizzazione di processi esterni al progetto e la pausa concede tempo e priorità agli altri programmi invece che al nostro progetto.

Esistono svariate soluzioni per effettuare questa operazione ed in questa sede ne vedremo tre, ognuna con i suoi pregi e difetti. Prima di vederle una per una inseriamo sopra un form cinque controlli.

Il primo controllo è una *Label* di nome **SecondiLabel**, con la proprietà *Caption* impostata su ***"Secondi"***.
Il secondo controllo è una *TextBox* di nome **Secondi**, con la proprietà *MaxLength* impostata a 2. Gli altri tre controlli sono tre *CommandButton* di nome **Pausa1**, **Pausa2** e **Pausa3**; essi serviranno per richiamare le tre soluzioni proposte in questo HowTo.

In questi esempi nasconderemo il form, lanceremo la funzione che esegue la pausa e solo dopo il numero specificato di secondi, il form sarà nuovamente mostrato.

Prima soluzione, senza l'utilizzo dell'API

Difficoltà: 1 / 5

La prima soluzione utilizzerà una funzione interna a Visual Basic. Vediamo subito il codice dell'evento Click sul primo pulsante.

```
1. Private Sub Pausa1_Click()
2.     Dim TempoIniziale As Long
3.     TempoIniziale = Timer
4.     Me.Hide
5.     While Timer < (TempoIniziale + Val(Secondi.Text))
6.         DoEvents
7.     Wend
8.     Me.Show
9. End Sub
```

Alla riga 2 dichiariamo una variabile di nome **TempoIniziale**. Essa servirà per memorizzare il momento in cui viene avviata la pausa. Tale momento viene memorizzato mediante lettura del valore restituito dalla funzione Timer (riga 3). Essa infatti restituisce il

numero di secondi trascorsi dalla mezzanotte.

Alla riga 4 viene nascosto il form e avviata la pausa.
Fintanto che il numero di secondi attuale è minore del momento iniziale + il numero di secondi specificato nella casella **Secondi** (riga 5), il programma sarà in pausa.
All'interno del ciclo è stato inserita l'istruzione DoEvents per abbassare l'attività del nostro progetto e concedere maggior priorità agli altri processi (riga 6).
Alla fine della pausa il form sarà nuovamente mostrato (riga 8).

Questa funzione ha un limite veramente pesante: la funzione Timer calcola il numero di secondi trascorsi dalla mezzanotte. Pertanto se, durante la pausa, si passa da un giorno al successivo per superamento della mezzanotte, il Timer viene azzerato e la pausa non terminerà più, bloccando il programma.

Seconda soluzione, con l'utilizzo dell'API

Difficoltà: 1 / 5

La seconda soluzione è molto simile alla precedente ma elimina il limite poco prima citato. Vediamo pertanto la funzione che regola il click sopra il pulsante **Pausa2**.

```
 1. Private Declare Function GetTickCount Lib "kernel32" () As Long
 2.
 3. Private Sub Pausa2_Click()
 4.     Dim TempoIniziale As Long
 5.     TempoIniziale = GetTickCount
 6.     Me.Hide
 7.     While GetTickCount < (TempoIniziale + Val(Secondi.Text) * 1000)
 8.         DoEvents
 9.     Wend
10.     Me.Show
11. End Sub
```

Alla riga 1 dichiariamo la funzione API *GetTickCount* vista in altri *HowTo* (019 - 029 - 030). Essa riporta il numero di millisecondi trascorsi dall'avvio di Windows.

La funzione è del tutto identica alla precedente. Gli unici cambiamenti riguardano la sostituzione della funzione *Timer* con la funzione *GetTickCount* e la specificazione della pausa in millisecondi anziché in secondi.

Essendo riportato il tempo in millisecondi, il valore indicato nella casella di testo **Secondi** sarà moltiplicato per 1000 (riga 7).

La funzione non risente del limite precedente e non presenta alcun difetto rilevante.

Terza soluzione, con l'utilizzo dell'API

Difficoltà: 1 / 5

L'ultima soluzione è una delle più semplici ed applicate. Consiste nell'utilizzare la funzione API Sleep. Vediamo l'evento Click dell'ultimo pulsante.

```
1. Private Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
2.
3. Private Sub Pausa3_Click()
4.     Me.Hide
5.     Sleep Val(Secondi.Text) * 1000
6.     Me.Show
7. End Sub
```

Dichiarata la funzione alla riga 1, la funzione in questione diventa semplicissima: alla riga 4 viene nascosto il form, alla riga 5 viene richiamata la funzione specificando il numero di millisecondi di pausa, dato dal numero di **Secondi** moltiplicato per 1000. Alla riga 6 viene nuovamente mostrato il form.

Il limite principale di questa funzione è che non è possibile inserire del codice nel momento di pausa. Quando viene richiamata la funzione Sleep l'intero programma viene congelato e tutti gli eventi del progetto non saranno richiamati.

Delle tre funzioni probabilmente la seconda è la più utile e personalizzabile e non presenta alcun rischio connesso.

Fibia FBI
25 Aprile 2001

Torna all'indice degli HowTo

