

MỆO VẬT

VISUAL BASIC 6

VDC MEDIA – 2001

MỤC LỤC

Làm cho Form luôn luôn hiện ra một chỗ và cùng một cỡ.....	2
Dùng Event nào để update Form?	3
Làm sao ta biết một Form đã được Loaded?	3
Làm một Form giống một Form có sẵn	4
Làm cho Controls biến mất.....	5
Chọn nhiều Controls như một nhóm	5
Chọn nhiều hàng trong ListBox	5
'Sort' các hàng trong một Listbox	6
Sắp ngay cột những chữ trong các hàng khác nhau của Listbox.	6
Drag một hàng từ Listbox này bỏ vào Listbox khác	6
Làm sao biết Drag Source là control nào	7
In những hàng được chọn của một ListBox.....	8
Bắt buộc user phải click ComboBox để chọn một hàng	8
Làm một Control hiện ra trên mọi trang của Tab Control.....	8
Sắp xếp thứ tự các TextBox cho Data Entry	9
Làm sao cho data vào trong ListBox hay ComboBox.....	9
Sao không thấy hình trong CommandButton?	10
Làm sao đặt icon lên Toolbar làm shortcut cho một menu command của VBIDE?	10
Các dạng của Ngày và Thì giờ	10
Làm sao đọc từ một Text file.....	14
Làm sao chứa vào một Text file	14
Cách thức Log một dữ kiện vào một Text file ?.....	15
Cách thức đọc/viết value của một Variable từ một "ini" file ?	17
Đổi Enter key ra Tab key	18
Đổi dạng MousePointer ra đồng hồ cát	19
Kiểm Help cho các Functions.....	19
Dùng Docket Dot Matrix Printer	20
Cho Application một Icon.....	21
VB shorcut Keys	21

Làm hiện ra một Form khác

Giả sử bạn đang ở trong Form1 và muốn làm cho Form2 hiện ra, hãy sử dụng:

`Form2.Show`

Khi muốn dấu Form2, dùng:

`Form2.Hide`

Khi một Form được 'Show' lần đầu tiên, nó được 'Load' (cho thành hình) và 'Show' ngay sau đó. Khi một Form đã có rồi, tức là đã được 'Load' rồi, lúc bạn 'Hide' nó có c q ng giống như kéo một tấm màn che nó lại. Kế đó, khi bạn 'Show' nó có c q ng giống như mở màn ra, không có gì ảnh hưởng Form.

Bạn có thể cho Form biến mất bằng cách 'Unload' nó, như:

`Unload Form2`

Lần tới khi bạn 'Show' nó, Form sẽ được 'Load' trở lại. Mỗi khi Form được 'Load', nó 'execute' sub `Form_Load`. `Form_Load` là nơi bạn viết các 'statements' để 'initialise' Form, tức là những gì cần phải được sắp xếp cho sẵn sàng trước khi Form thành hình.

Do đó bạn phải quyết định khi nào dùng `Unload` hay `Hide` đi? Làm cho Form biến mất. Nhớ rằng mỗi khi Form được 'Load' thì nó 'initialise' và mất một ít thì giờ.

Nếu bạn muốn làm cho một Form hiện ra và khiến nó dành nhận tất cả 'keystrokes' và 'mouse clicks' thì dùng:

`Form2.Show vbModal`

Ngay sau khi Form2 được 'Show' dưới dạng `vbModal`, nó dành ngay CPU process, đến đổi statement nằm ngay dưới đó trong Form1 c q ng đợi cho đến khi Form2 biến mất bằng `Unload` hay `Hide` mới chạy tiếp được.

Nếu bạn muốn Form2 hiện ra trong `Modeless` (không phải `Modal`) nhưng luôn luôn nằm trên Form1, ngay cả khi Form1 trở thành active, dùng:

`Form2.Show ,Me ' Me là Form1`

Làm cho Form luôn luôn hiện ra một chỗ và cùng một cơ

trong Sub `Form_Load`, dùng

Move fLeft, fTop, fWidth, fHeight

Để nhớ thứ tự các parameters trong Sub Move hãy nhớ tọa độ có X,Y . trong Sub Move ta nói đến vị trí X,Y (fLeft, fTop) rồi cỡ X,Y (fWidth, fHeight).

Move là một method có thể được dùng cho Form hay Control và rất hiệu quả.

Nếu muốn Frame1 luôn luôn nằm ở đỉnh trái, dùng:

Frame1.move 0,0

Nếu muốn chỗ và cỡ của Form 'configurable' (user có thể thay đổi) thì chứa các dữ kiện đó trong Registry rồi đọc fLeft, fTop, fWidth, fHeight từ Registry trong Sub Form_Load. (xem Function GetSetting)

Để cho Form2 hiện ra bên phải và ngang hàng với Form1, trong Sub Form_Load của Form2 dùng:

Sub Form_Load()

Move Form1.Width, Form1.Top

End Sub

Dùng Event nào để update Form?

Mỗi khi trở lại một Form sau khi làm việc ở một Form khác, ta muốn 'update' một số dữ kiện đang được 'display' để phản ánh những thay đổi vừa mới xảy ra.

Để code trong Sub Form_Activate.

Làm sao ta biết một Form đã được Loaded?

Dùng Function sau đây của Microsoft:

Function FormIsLoaded(MyFormName As String) As Boolean

' Determines if a form is loaded.

Dim i As Integer

FormIsLoaded = False

For i = 0 To Forms.Count - 1

If Forms(i).FormName = MyFormName Then

```

        FormIsLoaded = True
        Exit Function ' Quit function once form has been found.
    End If
Next i

End Function

```

Đi qua hết collection của Forms để xem có Form nào trùng tên.

Cách dùng:

```

If FormIsLoaded("Form2") then

    MsgBox "Form2 is already loaded"

End If

```

Không cho User Unload Form bằng cách click dấu X

Nhiều khi bạn muốn thu xếp mọi chuyện đàng hoàng trước khi Unload một Form nên bạn làm một Exit hay Close button cho user dùng. Để làm mất dấu X phía đỉnh phải, 'set' Property ControlBox của Form ra False. Nếu làm thế này có điểm bất lợi là khi 'minimize' Form, ta không thấy icon của Form. Có một cách vẫn để nguyên dấu X, nhưng khi user click nó thì ta check 'unloadMode' trong Sub Form_QueryUnload để biết lý do tại sao Form phải unload. Trong Sub Form_Unload có thể 'assign' Cancel ra True để 'abort' Unload.

Nếu bạn muốn không cho user minimise Form thì set Property MinButton ra False.

Làm một Form giống một Form có sẵn

Nhiều khi bạn muốn làm một Form giống giống một Form đã có sẵn trong Project. Thí dụ bạn muốn 'duplicate' Form2 để làm nên một Form3 để sau đó sửa chữa nó chút đỉnh. Bạn làm như sau:

1. Mở Form2 ra
2. Dùng Properties để đổi tên Form2 ra Form3
3. Dùng File | Save As menu command để chứa Form này dưới tên Form3.frm

Bây giờ bạn có thêm Form3 trong Project, trong khi Form2.frm vẫn còn đó nhưng không còn thuộc về Project nữa. Nếu bạn muốn cho nó vào Project trở lại thì dùng Project | Add File .. menu command để chọn Form2.frm bỏ vào Project.

Làm cho Controls biến mất

Bạn làm cho một Control như Label, Textbox, ListBox .v..v biến mất bằng cách cho Property 'visible' trở nên False, như:

```
TxtName.visible = False
```

Dùng

```
TxtName.visible = True
```

để làm Textbox txtName hiện ra trở lại. Khi muốn cho một nhóm Controls biến mất hay hiện ra cùng một lúc, chứa tất cả các Controls ấy trong một container như Frame hay PictureBox, rồi chỉ cần làm cho container ấy visible hay không.

Invisible Control rất tiện dụng khi ta dùng nó để chứa dữ kiện. Một TextBox tàn hình có thể Databound để lạng lẽ chứa data lấy từ một recordField của current Record cho bạn dùng. Bạn có thể 'assign' (cho nó bằng) một value vào một Control tàn hình. Nếu Control ấy Databound, value sẽ được viết vào database khi đúng lúc.

Chọn nhiều Controls như một nhóm

Chọn nhiều Controls rất tiện cho việc assign một property cho cả nhóm qua Properties Window. Để chọn nhiều Controls như một nhóm bạn có thể 'rubber band' (quấn dây thun) cả nhóm bằng cách 'drag' con trỏ thành một hình chữ nhật bao cả nhóm lại. Nếu các Controls ấy nằm trong một container như Frame hay Picture box, trước hết bạn phải 'click' Form rồi kế đó đè nút 'Ctrl' xuống trong khi 'rubber band' như nói trên. Một khi các Controls trong một nhóm đã được chọn rồi ta có thể dùng menu Format command để Align (ngang hàng) chúng hay làm cho chúng cùng cỡ.

Chọn nhiều hàng trong ListBox

Muốn cho user chọn nhiều hàng trong Listbox ta phải set Property Multiselect của Listbox ra Extended. Sau đó user có thể chọn nhiều hàng bằng cách bấm nút Shift hay Ctrl trong khi Click những hàng. Nhớ là nếu bấm nút Shift thì chọn những hàng liên tục giữa hai cái click, còn bấm nút Ctrl thì chọn hay không chọn thêm từng hàng một.

Mỗi hàng trong Listbox List1 được cho thứ tự từ 0 đến List1.ListCount-1.

Khi hàng i được selected thì List1.Selected(i) = True.

Muốn 'deselect' (không chọn) hàng đó thì dùng List1.Selected(i) = False.

'Sort' các hàng trong một Listbox

Nếu ta muốn những hàng trong một Listbox được sorted theo mẫu tự (alphabetical order) thì khi thiết kế (at design time) set Property Sorted của Listbox ra True. Trong khi xử lý ta không thay đổi value của Property Sorted được. Nếu một Listbox đã lỡ có Property Sorted = False mà ta muốn Sort nó trong khi xử lý thì làm sao? Bạn đặt thêm lên Form một Listbox khác. Listbox này có Property Sorted = True. Khi nào muốn Sort các hàng trong Listbox thứ nhất ta làm như sau:

1. Chép tất cả mọi hàng từ Listbox thứ nhất qua Listbox thứ nhì, các hàng sẽ được sorted tự động trong Listbox thứ nhì.
2. Clear Listbox thứ nhất
3. Chép mọi hàng từ Listbox thứ nhì qua Listbox thứ nhất.
Nhớ đừng cho user thấy Listbox thứ nhì bằng cách set Property Visible của nó ra False (tàn hình).

Lưu ý Clear một Listbox trước khi cho một số hàng mới vào và không muốn giữ những hàng cũ bằng cách dùng method Clear như List1.Clear.

Sắp ngay cột những chữ trong các hàng khác nhau của Listbox
Nếu muốn tất cả các chữ cột thứ 8 của các hàng trong một Listbox đứng thẳng hàng chiều cao (vertically) bạn có thể set Font của Listbox ra Courier, một loại Fixed Font. Tất cả các chữ của Fixed Font như Courier đều có cùng chiều rộng (character width). Ngược lại trong một Proportional spacing Font như "MS Sans Serif" chiều rộng của chữ 'i' nhỏ hơn của chữ 'm'. Nếu bạn muốn dùng Tab trong Listbox thì phải gọi một API Function để làm cho Listbox chịu nhận Tab (support Tab feature).

Sắp ngay cột những chữ trong các hàng khác nhau của Listbox.

Nếu muốn tất cả các chữ cột thứ 8 của các hàng trong một Listbox đứng thẳng hàng chiều cao (vertically) bạn có thể set Font của Listbox ra Courier, một loại Fixed Font. Tất cả các chữ của Fixed Font như Courier đều có cùng chiều rộng (character width). Ngược lại trong một Proportional spacing Font như "MS Sans Serif" chiều rộng của chữ 'i' nhỏ hơn của chữ 'm'. Nếu bạn muốn dùng Tab trong Listbox thì phải gọi một API Function để làm cho Listbox chịu nhận Tab (support Tab feature).

Drag một hàng từ Listbox này bỏ vào Listbox khác

Giả sử bạn muốn Drag một hàng từ List1 bỏ vào List2 và display nó ở cuối List2.

Ta sẽ dùng Mousedown Event của List1 mà Drag. Khi bắt đầu Drag ta muốn MousePointer đổi ra hình một trang giấy. Do đó ta phải Browse directory để cho vào Property DragIcon của List1 tên một Icon File. Bạn có thể vào Folder C:\Program Files\Microsoft Visual Studio\Common\Graphics\IconsDragDrop để chọn một Icon File.

```
Private Sub List1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
List1.Drag ' Bắt đầu drag
```

```
End Sub
```

```
Private Sub List2_DragDrop(Source As Control, X As Single, Y As Single)  
' Source ở đây là List1
```

```
List2.AddItem Source.Text ' Bỏ hàng từ List1 vào cuối List2  
Source.RemoveItem Source.ListIndex ' Lấy hàng ra khỏi List1  
List2.ListIndex = List2.ListCount-1 ' Highlight hàng cuối và làm cho nó được display
```

```
End Sub
```

Làm sao biết Drag Source là control nào

Ta có thể dùng Function TypeName như:
TypeName(Source) để xem nó là "ListBox" hay "TextBox" etc..hay dùng
TypeOf như:

```
If TypeOf Source Is ListBox Then
```

```
Msgbox " Drag source is a ListBox"
```

```
End If
```

Ta cũng có thể dùng Name của Source như:

```
If Source.Name = "List1" thenMsgBox "Drag source is List1"
```

```
End If
```

Ngoài ra ta cũng có thể dùng Property Tag của Source Control để chứa một biệt danh và dùng giống như tên của Drag Source như:

```
If Source.Tag = "MYLIST1" then
```


MsgBox "Tag of Drag Source is: " & Source.Tag

End If

In những hàng được chọn của một ListBox

Ta dùng Printer Object. Ráp tất cả những hàng được chọn lại thành một text string dài, thêm vào cuối mỗi hàng là Carriage Return / LineFeed (vbCrLf) .

```
Dim i
strOut = "" ' Initialise Output string không có gì cả
For i=0 to List1.Listcount-1
If List1.Selected(i) then
strOut = strOut & List1.List(i) & vbCrLf
End if
Next
Printer.Print strOut ' Gởi text String ra Printer
Printer.EndDoc ' Bắt đầu in ra
Nếu bạn muốn in từng hàng thì dùng:
Printer.Print List1.List(i) ' Không cần phải dùng vbCrLf
Muốn in chữ đậm thì mở BOLD lên bằng:
Printer.FontBold = True
và khi nào muốn tắt chữ đậm thì dùng:
Printer.FontBold = False
```

Bắt buộc user phải click ComboBox để chọn một hàng

Khi Form mới hiện ra, nếu đã có sẵn một hàng chữ nằm trong ComboBox, mà đó đúng là hàng user muốn dùng, ta không có dịp để dùng Event Click của ComboBox để làm chuyện gì tùy thuộc vào hàng mà user đã chọn.

Một mảnh lời là trước khi cho các hàng vào ComboBox ta để một hàng trống, thí dụ như:

```
Combo1.AddItem " "
```

ComboBox sẽ như trống không khi Form mới hiện ra, bắt buộc user phải chọn một hàng chứ không dùng hàng đầu 'by default' (tự động có nếu không làm gì).

Làm một Control hiện ra trên mọi trang của Tab Control

Mỗi trang của Tab Control chứa riêng những Control của nó. Nếu bạn muốn một Textbox, Listbox hay CommandButton hiện ra trên mọi trang của Tab, bạn tạo control đó bên ngoài Tab và kéo nó vào đặt lên trên Tab. Nế Control

nằm phía dưới Tab, dùng VBIDE menu Command "Format | Order | Bring to Front" để mang Control ra phía trước.

Sắp xếp thứ tự các TextBox cho Data Entry

Khi user điền các dữ kiện vào một Form (Data Entry), user rời mỗi TextBox bằng cách bấm nút Tab. Kế đó MouseCursor nhảy đến TextBox có Property TabIndex value theo thứ tự sau TabIndex value của TextBox vừa rời khỏi. Thường thường ta cho TabIndex value theo thứ tự từng cặp LblName, txtName rồi lblAddress, txtAddress, v.v. có TabIndex values thí dụ như 1,2,3,4. Khi txtName (TabIndex=2) mất Focus, mouseCursor dời đến lblAddress (TabIndex=3) , nhưng vì lblAddress là một Label nên không chứa mouseCursor được, do đó mouseCursor nhảy đến txtAddress (TabIndex=4). Nếu Caption của lblAddress có một character underlined, như

```
lblAddress.Caption = "Add&ess" ' character "e" underlined
```

khi user bấm Alt+e bất cứ lúc nào mousePointer sẽ nhảy đến lblAddress nhưng rồi cuộc rút vào TextBox kế đó, tức là txtAddress trong trường hợp này.

Nếu bạn không muốn mousePointer đáp vào một TextBox nào set Property TabStop của nó ra False. Bạn cũng có thể không cho user thay đổi dữ kiện ở một TextBox bằng cách set Property Enabled ra False hay Locked ra True.

Làm sao cho data vào trong ListBox hay ComboBox

Khi Design ta có thể dùng Property List của ListBox hay ComboBox để đánh vào các dòng data. Mỗi khi xuống hàng phải nhớ dùng Ctrl+Enter chứ không phải Enter, nếu không VBIDE tưởng là bạn đã chấm dứt data entry.

Một cách khác để cho data vào ListBox hay ComboBox là dùng Method AddItem trong Sub Form_Load, như:

```
Sub Form_Load()  
  
Dim i  
  
For i=1 to 10  
  
    List1.AddItem "Line " & CStr(i)  
  
Next  
  
lstState.AddItem "NSW"
```

```
lstState.AddItem "VIC"
```

```
lstState.AddItem "QLD"
```

```
lstState.AddItem "SA"
```

```
lstState.AddItem "WA"
```

```
lstState.AddItem "TAS"
```

```
End Sub
```

Sao không thấy hình trong CommandButton?

Bạn dùng Properties Window để đặt một Picture vào một CommandButton bằng cách Browse và chọn một Image File. Nhưng bạn không thấy Picture đâu cả. Bạn còn cần phải set Property Style của CommandButton ra "Graphical", thay vì "Standard".

Làm sao đặt icon lên Toolbar làm shortcut cho một menu command của VBIDE?

Dùng Menu command View | Toolbars | Customize để display Dialog Customize. Kế đó chọn Tag Commands, bên trái sẽ là một Listbox chứa danh sách các Categories và bên phải là những Commands thuộc về Category đang được highlighted. Thử select từng Category để tìm trong Listbox bên phải cái Command có icon mà bạn muốn. Drag icon đó lên Toolbar. Nhân dịp này bạn có thể bỏ một icon có sẵn trên Toolbar bằng cách Drag nó ra chỗ khác.

Các dạng của Ngày và Thời giờ

Dùng Function Format để lấy ngày giờ theo dạng bạn muốn. Thí dụ:

```
strToday = Format (Now(), "ddd dd-mmm-yyyy hh:nn:ss")
```

- Now() cho ta hiện giờ (realtime lấy từ đồng hồ của CPU)
- ddd cho Mon, Tue .vv..
- mmm cho Jul, Feb .vv..

Nhớ dùng "nn" cho phút.

Hai cách dùng Timer

1. Timer có Interval tính bằng milliseconds. Khi Timer1 được Enabled (tức là Timer1.Enabled=True) nó sẽ chạy Sub Timer1_Timer mỗi Interval msec.

Thí dụ ta muốn Label1 chớp tắt (blink):

```
Timer1.Interval = 500 ' Nửa giây
```

```
Timer1.Enabled = True ' Bắt đầu chạy Timer1 cho hết interval
```

```
Sub Timer1_Timer()
```

```
Label1.Visible = NOT Label1.Visible
```

```
End Sub
```

Cứ mỗi nửa giây hể Label1 đang hiện ra thì cho nó tàn hình và ngược lại.

Khi nào muốn ngưng chớp tắt thì dùng:

```
Timer1.Enabled = False
```

```
Label1.Visible = True
```

2. Để làm một việc trong tương lai, ta có thể dùng Timer nhưng ngay trong

Sub Timer1_Timer ta để câu Timer1.Enabled = False

để tắt Timer1 sau khi làm xong công chuyện. Thí dụ:

```
Sub Timer1_Timer()
```

```
Timer1.Enabled = False ' Ngưng Timer1
```

```
' Để code làm công chuyện tại đây
```

```
End Sub
```

Đợi một chút

Có khi ngay ở giữa code bạn muốn đợi (wait) một chút. Bạn có thể làm theo bài mẫu của Microsoft như sau:

```
Dim PauseTime, Start, Finish, TotalTime
```

```
' Code đang làm việc nữa chừng
```

```
PauseTime = 5 ' Set duration in seconds.
```

```
Start = Timer 'Set start time as now
```

```
Finish = Start + PauseTime ' set finish time
```

```

Do While timer < Finish
    DoEvents ' Yield to other processes.
Loop
' Code tiếp tục công việc

```

Timer dùng ở đây là một Function có sẵn (in-built) của VB chứ không phải Control Timer. Timer cho ta số giây kể từ nửa đêm.

Ta dùng DoEvents để khỏi dành hết CPU time trong một closed Do While Loop. Nó nhường cho các process khác chạy trong khi chờ đợi Finish.

Khi bạn chạy một Closed Loop giống như trên và assign cho Label1 một Caption với value biến đổi để Feedback cho user biết tình hình diễn tiến đến đâu, nếu bạn không dùng DoEvents giống như nói trên, Label1 sẽ không có dịp 'refresh' để 'display' value mới của Label1.Caption.

Không biết bạn có để ý thấy cách chờ đợi như trên có thể gặp trở ngại nếu lúc bấy giờ là gần đến nửa đêm ít hơn 5 giây. Khi đó Finish sẽ lớn hơn 3600*24 giây và Timer sẽ không bao giờ đạt đến con số ấy để nhảy ra khỏi Loop. Giống như mình lập gia đình vậy, sẽ làm mọi suốt đời, không có lối thoát.

Một cách khác để giải quyết là ta nhờ một Timer1 chạy độc lập để cho biết khi nào thời gian đã trải qua (elapsed) bằng cách set một 'variable' tên blnElapsed ra True. blnElapsed phải được để trong phần General Declaration để cả code chính và Sub Timer1_Timer đều thấy nó.

```

Dim blnElapsed as Boolean ' Declare in General section

```

```

Sub Timer1_Timer()

```

```

    Timer1.Enabled = False

```

```

    blnElapse = True

```

```

End Sub

```

```

' Code đang làm việc nữa chừng

```

```

    blnElapsed = False

```

```

    Timer1.Interval = 5000 ' Set duration in msec.

```

```

    Timer1.Enabled = True 'Start Timer1

```

```

Do

```

```

    DoEvents ' Yield to other processes.

```

Loop Until blnElapsed

' Code tiếp tục công việc

Cách này có vẻ được nhưng nó vẫn bắt chẹt (hold up) CPU và làm cản trở các processes khác (đây là giới hạn của VB).

Cách hoàn hảo nhất là ta đợi bằng cách dùng Timer1, nhưng đặt phần code còn lại để tiếp tục công việc ngay trong Sub Timer1_Timer.

' Code đang làm việc nữa chừng

Timer1.Interval = 5000 ' Set duration in milliseconds.

Timer1.Enabled = True ' Start Timer1

Sub Timer1_Timer()

Timer1.Enabled = False

' Code tiếp tục công việc

End Sub

Làm thùng rác cháy lửa nửa giây

Trong khi xử lý, ta có thể biến hình của một Image bằng cách cho nó một Picture mới như:

Image1.Picture = LoadPicture("PictureName.bmp")

rồi bắt đầu một Timer với Interval bằng 500 milliseconds.

Đầu tiên đặt một Timer tên Timer1 lên Form. Vì lúc đầu Timer1.Interval bằng 0 nên Timer1 chưa chạy được (giống như Enabled = False)

' Cho hình thùng rác cháy lửa

Image1.Picture = LoadPicture("TrashBinOnFire.bmp")

Timer1.Interval = 500

Timer1.Enabled = True ' Nửa giây sau sẽ chạy Timer1_Timer

Sub Timer1_Timer()

Timer1.Enabled = False ' Ngưng Timer1

' Cho hình thùng rác thường

Image1.Picture = LoadPicture("TrashBin.bmp")

End Sub

Làm sao đọc từ một Text file

Sau đây là một thí dụ đọc data từ một Textfile tên "Friends.txt" nằm trong cùng folder với chương trình đang xử lý.

```
Sub Form_Load()

    Dim strALine as string

    Dim strLocalFolder as string

    Dim strFullPathFileName as string

    strLocalFolder = App.path

    If Right(strLocalFolder,1) <> "\" then

        strLocalFolder = strLocalFolder & "\"

    End If

    strFullPathFileName = strLocalFolder & "Friends.txt"

    Open strFullPathFileName for input as #1

    lstFriend.Clear ' Clear ListBox lstFriend

    Do While Not EOF(1) ' Read till End-Of-File

        Line Input #1, strALine ' Read a line

        lstFriend.AddItem strALine 'Add that line to ListBox

    Loop

    Close #1 ' Close the file

End Sub
```

Làm sao chứa vào một Text file

Sau đây là một thí dụ đọc data từ một Textfile tên "Friends.txt" nằm trong cùng folder với chương trình đang xử lý.

```
Sub Form_Load()
```

```

Dim i

Dim strLocalFolder as string

Dim strFullPathFileName as string

strLocalFolder = App.path

If Right(strLocalFolder,1) <> "\" then

    strLocalFolder = strLocalFolder & "\"

End If

strFullPathFileName = strLocalFolder & "Friends.txt"

Open strFullPathFileName for output as #2

For i=0 to lstFriend.Listcount-1

    Print #2, lstFriend.List(i)

Next

Close #2 ' Close the file

End Sub

```

Cách thức Log một dữ kiện vào một Text file ?

Nhiều lúc ta cần Log vào trong một LogFile các biến cố xảy ra trong khi nhu liệu đang chạy để sau này kiểm lại tình hình.

```

Sub LogEvent(ByVal GivenFileName, ByVal msg As String, HasFolder As Boolean, IncludeTimeDate As Integer)

```

```

' Append event message Msg to a text Logfile GivenFileName

```

```

' If GivenFileName is fullPathName then HasFolder is true

```

```

' IncludeTimeDate = 0 : No Time or Date

```

```

'           = 1 : Prefix with Time

```

```

'           = 2 : Prefix with Time and Date

```



```

Dim FileNo, LogFileName, theFolder

If HasFolder Then

    LogFileName = GivenFileName

Else

    If Right(App.Path, 1) <> "\" Then

        theFolder = App.Path & "\"

    Else

        theFolder = App.Path

    End If

    LogFileName = theFolder & GivenFileName

End If

FileNo = FreeFile

If Dir(LogFileName) <> "" Then

    Open LogFileName For Append As FileNo

Else

    Open LogFileName For Output As FileNo

End If

Select Case IncludeTimeDate

Case 0 ' No Time or Date

    Print #FileNo, Msg

Case 1 ' Time only

    Print #FileNo, Format(Now, "hh:nn:ss ") & Msg

Case 2 ' Date & Time

```

```

Print #FileNo, Format(Now, "dd/mm/yyyy hh:nn:ss ") & Msg

End Select

Close FileNo

End Sub

```

Coi chừng trường hợp LogEvent được gọi bởi hai Sub khác nhau cùng một lúc. Khi Sub thứ nhì muốn mở LogFileName thì bị error vì LogFileName đã bị Sub thứ nhất mở rồi. Trong trường hợp đó ta có thể sửa LogEvent lại cho nó AddItem Msg vào một Listbox rồi giao nhiệm vụ viết xuống File cho một Process chuyên viên thứ ba.

Cách thức đọc/viết value của một Variable từ một "ini" file ?

Có khi ta muốn đọc value của một variable chứa trong một file có extension là "ini" (còn gọi là configuration file) nằm trong folder C:Windows hay c:WinNT. Trong file này những cặp variable=value nằm trong các sections, mỗi Section header có dạng như [System Var]. Thí dụ "protocol.ini" chứa những dòng sau:

```

[ndishlp$] <= section header

DriverName=ndishlp$ <= cặp variable=value

[protman$] <= section header

DriverName=protman$ <= cặp variable=value

priority=ndishlp$ <= cặp variable=value

[data] <= section header

version=v4.10.1998 <= cặp variable=value

```

Dùng Function ReadPrivateProfileString sau đây để đọc value của VarName\$ dưới section header SectionHeader\$ từ file Filename\$:

```

Function ReadPrivateProfileString( SectionHeader$, VarName$, Filename$)
As String

' Read data from an Ini file in default Windows directory

' Filename$ : The Ini file name eg: myProg.ini

' SectionHeader$ : The Section name in Ini file eg: [System Var]

```

' VarName\$: Variable name whose data value we want to get

Dim RetStr As String

RetStr = String(255, Chr(0)) ' Prepare a string of 255 zeros (number).

'Get Requested Information

ReadPrivateProfileString = Left(RetStr,
GetPrivateProfileString(SectionHeader\$, ByVal

VarName\$, "", RetStr, Len(RetStr), Filename\$))

End Function

Tương tự như vậy, ta có thể viết value mới của một variable vào trong một "ini" file như sau:

Dim ErrCode

ErrCode = WritePrivateProfileString(SectionHeader\$, VarName\$,newValue,
Filename\$)

Nhớ để hai câu tuyên bố cho các API Function ta cần vào một Basic Module:

Public Declare Function GetPrivateProfileString Lib "kernel32" Alias
"GetPrivateProfileStringA" (ByVal lpApplicationName As String, ByVal
lpKeyName As Any, ByVal lpDefault As String, ByVal lpReturnedString As
String, ByVal nSize As Long, ByVal lpFileName As String) As Long

Public Declare Function WritePrivateProfileString Lib "kernel32" Alias
"WritePrivateProfileStringA" (ByVal AppName\$, ByVal KeyName\$, ByVal
keydefault\$, ByVal Filename\$)

Đổi Enter key ra Tab key

Khi user điền các dữ kiện vào một Form (Data Entry), thường thường user rời mỗi TextBox bằng cách bấm nút Tab. Có khi user 'prefer' (chuộng hơn) dùng Enter key thay vì Tab key. Bạn có thể dùng Event Keypress của TextBox để đổi Enter ra Tab. Thí dụ:

Private Sub TxtName_KeyPress(KeyAscii As Integer)

If KeyAscii = vbCr then

KeyAscii = 0 ' Nuốt trọn Enter key để không bị side effect

' Emulate (làm giống như) bấm nút Tab bằng cách dùng
SendKeys

Sendkeys "{TAB}"

End if

End Sub

Event Keypress của TextBox được dùng như trên rất tốt. Nhưng nếu Form có rất nhiều TextBox ta có thể tránh lặp lại việc dùng KeyPress Event của từng TextBox bằng cách dùng Sub Form_Keypress, như:

Private Sub Form_KeyPress(KeyAscii As Integer)

If KeyAscii = vbCr then

KeyAscii = 0 ' Nuốt trọn Enter key để không bị side effect

Sendkeys "{TAB}"

End if

End Sub

Để nhắc Form chụp lấy Keystroke trước TextBoxes, ta phải set Form Property 'KeyPreview' ra True, nếu không Sub Form_Keypress không bao giờ được gọi.

Đổi dạng MousePointer ra đồng hồ cát

Khi program sắp làm chuyện gì hơi lâu, bạn muốn cho user biết là program đang làm việc chứ không phải chết đứng (hang) như khi mình gặp người đẹp lần đầu, dùng:

Screen.MousePointer = vbHourglass ' Biến MousePointer ra hình đồng hồ cát

' Bỏ Codes vô đây . . .

Screen.MousePointer = vbDefault ' Biến MousePointer trở lại dạng bình thường

Kiểm Help cho các Functions

Trong khi đang viết program, thí dụ bạn muốn đọc về cách dùng 'Select'. Đánh chữ 'Select' đoạn 'highlight' nó rồi bấm nút F1. Nhớ là bạn có thể 'highlight'

một chữ bằng cách để con trỏ (mousepointer) lên trên chữ đó đoạn 'doubleClick'.

Nếu bạn không biết chắc một tên Function đánh vần như thế nào, thí dụ bạn không nhớ rằng Function 'Sendkeys' có chữ 's' ở cuối không. Thử đánh 'Sendk' rồi bấm 'Ctrl+Space' (nhấn nút 'Ctrl' trong khi bấm Space bar), VBIDE (Visual Basic Integrated Development Environment) sẽ đánh những chữ còn lại của chữ 'SendKeys'. Nếu bạn chưa đánh gì hết mà bấm 'Ctrl+Space', VBIDE sẽ 'pop-up' danh sách của tất cả Functions có thể dùng được lúc này, của VB và của chính bạn đang viết. Do đó bạn có thể đặt tên Constant hay Function, Sub của mình bắt đầu bằng những 'characters' để nhớ, và dùng tên dài cho dễ đọc. Thí dụ bạn dùng 'my' làm hai characters bắt đầu:

```
Const myTableSize = 20
```

```
Sub mySortRoutineForHTML
```

Sau đó chỉ cần đánh 'mySo' Ctrl+Space sẽ ra trọn chữ mySortRoutineForHTML.

Dùng Docket Dot Matrix Printer

Nếu bạn dùng Docket Dot Matrix Printer thì không thể xài một Windows Printer Driver bình thường được, vì nó in hình ảnh của cả một trang trong khi ta muốn in từ dòng.

Ta phải vô Start | Settings | Printers để Add một Generic / Text Only printer driver.

Kế đó nếu cần bạn cho vào những Control Command Strings để in BOLD, ITALIC, v.v. lấy từ Cẩm nang của printer.

Thí dụ số 28 để mở BOLD, 29 để tắt BOLD. Nếu Docket Printer có thể in màu đỏ và bạn không cần dùng ITALIC, bạn có thể cho Control Command Strings của mở tắt màu đỏ vào chỗ mở tắt ITALIC. Sau đó khi trong code bạn mở ITALIC thì thật ra driver gửi Control Command String cho mở màu đỏ.

Thí dụ:

```
Printer.FontItalic = True ' mở màu đỏ  
Printer.Print " ** SUPER DISCOUNT STORE **"  
Printer.FontItalic = False ' tắt màu đỏ  
Printer.Print "1 T Shirt $24.50"  
Printer.FontIBold = True ' mở chữ đậm  
Printer.Print " Total: $24.50"  
Printer.FontIBold = True ' tắt chữ đậm
```

```
Printer.Print "Thank you"  
Printer.EndDoc
```

Nếu Docket Printer không xuống hàng hàng ngang, bạn gởi trọn 40 characters ra cho mỗi hàng, (40 characters là chiều rộng của Docket) Muốn gởi trọn 40 characters bạn phải ghép thêm blank spaces cho mỗi dòng dài đủ 40 characters. Thí dụ:
strLine = Space(40) ' Initialise 40 blank spaces
Mid(strLine,15,9) = "Thank you" ' Nhét hai chữ Thank you vào, bắt đầu từ cột 15
Printer.Print strLine

Cho Application một Icon

Bạn có thể cho mỗi Form một icon bằng cách dùng Property Icon trong Properties Window của Form để chọn một icon file. Mỗi khi một Form được Minimised ta sẽ thấy Icon của nó trên Desktop. Bây giờ muốn dùng icon của một Form làm icon của Application (nhu liệu), bạn dùng menu command Project | Project? Properties. Khi Dialog hiện ra chọn Tab 'Make' và click Icon combo box để chọn Form bạn muốn chỉ định.

VB shortcut Keys

Find Ctrl+F
Replace Ctrl+H
Find Next F3
Copy Ctrl+C
Cut Ctrl+X
Paste Ctrl+V
Undo Ctrl+Z
Delete Current Line Ctrl+Y
Delete to End of Word Ctrl+Delete
View Code Window F7
View Definition of Sub/Function Shift+F2
Go to last position Ctrl+Shift+F2
Debug Step Into F8
Debug Step Over Sub/Function Shift+F8
View Object Browser F2
View Properties Window F4

Comment (biến câu code thành câu chú thích) / Indent (cho xít vô) một số dòng code

Nếu không thấy Edit Toolbar thì dùng menu command View | Toolbars | Edit để làm Edit Toolbar hiện ra. Chọn một số dòng code bạn muốn làm việc đoạn dùng Comment/Uncomment, Indent/Unindent buttons để xử lý.