

## Re: Listbox to delimited text file

---

*Source:* <http://www.tech-archive.net/Archive/VB/microsoft.public.vb.controls/2006-04/msg00224.html>

---

- *From:* Greg from Kentucky <[GregfromKentucky@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:GregfromKentucky@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Sat, 29 Apr 2006 12:02:02 -0700
- 

Thank you, Bob! Unfortunately, I don't know how to implement your suggestion. It looks logical now that you went to the trouble of responding with your solution but I don't know where it is I should put in the code.. Maybe 'amateur' was too strong a word – beginner might have been more appropriate.

I should make clear that I don't necessarily want to remove what is in the Listbox, although that would be okay, I just want the user to fill up maybe as many as 75 rows with his data, then click a button to either copy or cut the data from the Listbox using your suggested code to the semicolon delimited text file.

I have a sample of the output from a working Excel Workbook – the SpreadSheets are just so cumbersome for the user with the paste special, different worksheets in the book, and dealing with it being read only, and even removing the formulas from cells he shouldn't be in. Let me paste three lines of the output text file below and many thank you's for any additional assistance.

---

```
ABCDEFGHIJ-16062;ABCDEFGHIJ-16062;ABCDEFGHIJ;123456789;S;04/13/2006
06:20;2006-D00206;114860;A
ABCDEFGHIJ-18388;ABCDEFGHIJ-18388;ABCDEFGHIJ;123456789;S;04/13/2006
06:35;2006-D00210;118440;A
ABCDEFGHIJ-5332;ABCDEFGHIJ-5332;ABCDEFGHIJ;123456789;S;04/13/2006
07:00;2006-D00202;120240;A
```

---

"Bob O`Bob" wrote:

Greg from Kentucky wrote:

I am mostly an amateur with VB6 but I would like some advice about two problems. I have run out of help materials and have no other choice but to ask for help.

First:

I have several single-line textboxes, a Combo Box, and some calculated labels that the user enters his data into, I send the contents of each to a Listbox with a CommandButton like this:

## Re: Listbox to delimited text file

```
lst1.AddItem Format(Out, txtOut.text) & vbTab & Format(Vehicle,  
comboVehicle) & vbTab & Format(Number, txtNumber.text) & vbTab &  
Format(txtSerial.text, Serial) . . .
```

This works fine and is a perfect preview for the user. My wish is to export all of this data into a plain text file and change the tab spaces (Listbox columns) into semicolons, leaving the rows intact and putting a carriage return/break to represent each row. I need it in semicolon delimited format so it can be imported by another program that I have no control over.

like this?

```
For i = 0 To lst1.ListCount - 1  
s = s & Replace(lst1.List(i), vbTab, ";") & vbCrLf  
Next
```

something pretty similar to that should get everything you want into a string.

The CommandButton resets all the TextBoxes, ComboBox, and Labels at the same time it populates the Listbox and moves the cursor back to the first Textbox for the user to enter a new record.

I am not partial to the Listbox – I only used it because I knew how and it gives immediate feedback to the user that his entry was made correctly and gives him an option to remove the entire line/row if it is wrong.

I am open to using a MSFlexGrid, ListBox, RichTextBox, or even a DataBase of some sort, I just don't know how. I am open to all suggestions!

Second:

I would like some advice on doing the actual writing and saving of the Text file, assuming someone out there can help me with the first part.

I'm partial to wrapping text file stuff into all-at-once procedures:

```
Public Function PutFile(sFn As String, sData As String) As Long  
'write file from string  
'don't forget – this just writes the first so many bytes –  
' if file already existed and was longer, other bytes will remain.  
Dim fh As Integer 'file I/O handle
```

Re: Listbox to delimited text file

```
fh = FreeFile
Open sFn For Binary Access Write Lock Read Write As fh
Put #fh, , sData
Close #fh
PutFile = Len(sData)
End Function
```

```
Public Function GetFile(sFn As String) As String
'read entire file into a string
Dim fh As Integer 'file I/O handle
fh = FreeFile
Open sFn For Input Access Read Lock Write As fh
GetFile = Input$(LOF(fh), fh)
Close #fh
End Function
```

I find that until file sizes get well up into the dozens of megabytes,  
all-at-once outperforms line-by-line I/O pretty dramatically.

Bob

—