

# The dBASE Reports Book – Errata Sheets

by Ken Mayer  
April, 2007

No matter how hard one tries to get everything 100% accurate, little errors slip into a book's text. These pages are designed to show the errors and the corrections ...

## Chapter 1

Page 7 – the URL for the dUFLP is bad – it should be:

<http://www.goldenstag.net/dbase/index.htm#dUFLP>

(I left out the web page ... you can also just use:

<http://www.goldenstag.net/dbase>

and click on the menu link for the dUFLP instead ...)

Page 7 - The program: *ComponentPalletteUpdate.prg* is actually *ComponentPaletteUpdate.prg* (misspelled “Palette” in the book). In addition, the table “CREG0009.DBF” may be something else, depending on the language you have installed for dBASE. For example, Antonio Campagna (who notified me of this issue), has “CREG0010.DBF” instead. In those cases, you will need to slightly modify the program “ComponentPaletteUpdate.prg”:

```
// first, get the path to the file ...
cPath = _dBWinHome+"designer\form"
cTable = "creg0009.dbf"
cReg = cPath+"\ "+cTable
```

So that it looks for “creg0010.dbf” instead.

Page 14 - When you create the DamsParentChild.dmd file, you go into the source code (as you do on Page 14 of this book), you should check at the end of the source code to see if this statement exists:

```
this.rowset = form.Dams1.rowset
```

It should appear before the ENDCLASS statement. If it does not, please add it. This will affect some of the exercises in the book.

If you have chosen to just use the copy of this file that comes with the .zip file mentioned earlier in this chapter (with the tables), you may want to see if the statement exists in there as well. A recent upload ensured that it is there.

## Chapter 8

Page 114, Tim MacAlpine notes that the path to the file “labe0009.db” in the book reflects a custom path on my computer, and not the default in dBASE. This file by default should be found at (note any language differences):

```
C:\Program Files\dBASE\PLUS\Designer\label\labe0009.db
```

(And it should also be noted that the file may have a different number based on language, as noted elsewhere in these pages.)

(Also Chapter 8)

The form discussed on the same page (114), in the dUFLP, may have the path hardcoded (depending on the version of the dUFLP you have) – if I modified this in the form designer, the designer will have streamed it out incorrectly with the path. Please read the directions in the header for the file “CustLabl.wfm” if the path is wrong (open the file in the Source Code Editor: modify command :dUFLP:CustLabl.wfm and read the comments, following the instructions).

## Chapter 9

This is not really an error, but an addition to the chapter. This is the chapter on working with the reportViewer object in dBASE. Roger Sauer posted this on the newsgroups.

While viewing a page in the reportViewer, if you move the vertical scrollbar down to see more of the page, then go to the next page, the scrollbar stays down. That is, even though the next page has been rendered, you’re not viewing the top of that page, as you might expect.

This function can be used to programmatically set the vertical and horizontal scrollbars back to the “home” (or top/left) position of the report. You would call this just before issuing a report viewer’s ref object’s *render()* method. Note that this uses the Windows API:

```
function ScrollHome( hWnd )
    Usage: for example, with a reportViewer,
           in code that navigates from page to
           page, you might use this before
           calling the reportViewer's ref.render():
           class::ScrollHome( form.reportViewer1.hwnd )
*/

#define WM_HSCROLL 0x0114
#define WM_VSCROLL 0x0115
#define SB_PAGEUP 2
#define SB_PAGELEFT 2
if empty( hWnd )
    return
endif

if type( "SendMessage" ) # "FP"
    extern cLong SendMessage( cHandle, cInt, cLong, cLong );
    user32 from "SendMessage"
endif

SendMessage( hWnd, WM_VSCROLL, SB_PAGEUP, 0 )
SendMessage( hWnd, WM_HSCROLL, SB_PAGELEFT, 0 )

return
```

## Chapter 10

This chapter is on Multiple tables and Grouped reports, and there are a couple of minor confusion points (pointed out by Tim MacAlpine).

Page 148, toward the bottom, mentions “Invoice.dmd”. This will be in the Tables folder - the one you placed the tables in. You will want to use this data module, either copy it to a “Chapter 10” folder, or simply use it for working with the tutorial.

Page 157 discusses opening the (above) data module in the Source Code Editor, and adding a line that should be boldfaced (but is not – an error on my part when formatting this chapter of the book):

```
this.LINEITEMS1 = new QUERY()
this.LINEITEMS1.parent = this
with (this.LINEITEMS1)
    OnOpen = class::LineItems_OnOpen
    ...
endwith
```

(As a side note, the statement in the function on the same page that wraps – beginning with “f.beforeGetValue” should be a single line in the source code, with no break – do not press the Enter key when entering it.)

## Chapter 17

Page 274 – The second paragraph – the problem discussed here may also be the text object’s *variableHeight* property, which defaults to *false*. If you set this to *true*, then the band may adjust properly. You might try this before trying to alter the reportGroup headerBand’s *expandable* property as the book discusses.