

- [fqvarfort](#)

Here's an updated version of the code [first posted here](#). It uses a portion of the code from the previous version but much of it is replaced with faster and easier-to-follow solutions. This version is approx 13x faster than the previous version. No new features has been added though.

```
leng=90                                'length of texteditor line
Global leng ', tag$, comment$

WindowWidth=760
WindowHeight=590
UpperLeftX = Int((DisplayWidth - WindowWidth)/2)
UpperLeftY = Int((DisplayHeight - WindowHeight)/2)

Nomainwin
Menu #main, "&File", "&Open", getTutorial, "&Save", saveText, "&Quit", quitMenu
Texteditor #main.tb1, 5, 5, 740, 500
Statictext #main.status, "", 680, 510, 60, 25
Open "Parse Liberty Basic 4 Tutorial" for Window as #main

Print #main, "Trapclose quit"
Print #main.tb1, "!autoresize";
Print #main, "Font Courier_New 10"
Print #main.status, "!Font Courier_New 11 Bold"
Print #main.status, "!Hide" 'otherwise will be visible if textedit
or is maximized

Call getTutorial
Wait

! ***** SUBS AND FUNCTIONS *****

Sub quitMenu
Call quit handle$
End Sub

Sub quit handle$
Close #main
End Sub

Sub getTutorial

oldtimer = time$("ms")
```

```
FileDialog "Open Liberty Basic lesson file", "*.lsn;*.txt", fileName$
If Right$(fileName$,3) = "txt" Then Call getText fileName$
If fileName$<>" " Then
Print #main.status, "!Show"
Open fileName$ For Input As #1
Open fileName$; ".txt" For Output As #2

ProgressLOF = LOF(#1)

Do
'Read next line
Line Input #1, comment$

'Get the tag of current line "<TAG>"
startPos=Instr(comment$,"<",marker)
endPos=Instr(comment$,">",marker)
tag$ = Left$(comment$,endPos-startPos+1)

'Update progressbar
Gosub [ProgressUpdate]

'Print 2 blank lines in
'front of lesson titles
Select Case tag$
Case "<lesson-title>"
Call printBlankLines 2
End Select

'Print value (code, comments or title)
Select Case tag$
case "<lesson-code>", "<section-code>", "<example-code>"
Line Input #1, comment$
Gosub [ProgressUpdate]
Call printCode comment$
Case "<lesson-comment>", "<chapter-comment>", "<section-
comment>", "<example-comment>"
Line Input #1, comment$
Gosub [ProgressUpdate]
Call printComments comment$
Case "<lesson-title>", "<section-title>", "<chapter-
title>", "<example-title>"
startPos=Instr(comment$,">",1)+1
endPos=Instr(comment$,"</",1)-1
comment$=Mid$(comment$,startPos,endPos-(startPos-1))
```

```
        call printTitle comment$
    Case Else
        comment$ = ""
    End Select

    'Print 2 blank lines
    'below the value
    if (comment$ <> "") then
        call printBlankLines 2
    End if
    Loop While eof(#1) = 0
    Print #main.status, ""
    Close #1
    Close #2

    text$ = GetFileContents$(fileName$, ".txt")
    Print #main.tb1, "!contents text$"
    Print #main.status, "!Hide"
    Print #main.tb1, "!origin 1 1" ;
End if

Print "Time: "; time$("ms") - oldtimer
Exit Sub

[ProgressUpdate]
'See how far we have read in the
'lsn file and calculate the new
'progress %, only update the
'status when the progress %
'has actually changed
ProgressLOC = LOC(#1)
ProgressNew = Int((ProgressLOC / ProgressLOF) * 100)
If (ProgressNew <> ProgressOld) Then
    ProgressOld = ProgressNew
    Print #main.status, ProgressNew; " %"
End If
Return

End Sub

Sub getText fileName$
    Open fileName$ For Input As #1
    Print #main.tb1, "!contents #1";
    Close #1
End Sub
```

```
Sub saveText
    Filedialog "Save As... (TEXT ONLY)", "*.txt", fileName2$
    If fileName2$<> "" Then
        Open fileName2$ For Output As #1
        Print #main.tb1, "!contents? textTutorial$";
        Print #1, textTutorial$
        Close #1
    End If
End Sub

Sub printTitle title$

    Print #2, Space$(int(leng/2)-(int(len(title$)/2)));
    Print #2, title$

End Sub

Sub printCode code$

    If Right$(code$, 32) = "no code example for this section" Then Exit
Sub
    If Right$(code$, 28) = "place your example code here" Then Exit Sub
    Call ConvHTML2Text code$
    Call WordWrapCode code$, leng
    Print #2, code$

End Sub

Sub printComments comment$

    Call ConvHTML2Text comment$
    Call WordWrapCode comment$, leng
    Print #2, comment$

End Sub

Sub printBlankLines num

    For x=1 To num
        Print #2, ""
    Next x

End Sub

Sub ConvHTML2Text ByRef Text$
```

```
Call ReplaceText Text$, "&#9;", "    "  
Call ReplaceText Text$, "&quot;", Chr$(34)  
Call ReplaceText Text$, "&lt;", "<"  
Call ReplaceText Text$, "&gt;", ">"  
Call ReplaceText Text$, "&amp;", "&"  
Call ReplaceText Text$, "&#13;&#10;", Chr$(13) + Chr$(10)
```

End Sub

Sub ReplaceText ByRef Text\$, ReplaceFrom\$, ReplaceTo\$

```
    Pos = InStr(Text$, ReplaceFrom$)  
    Do While (Pos > 0)  
        Text$ = Left$(Text$, Pos - 1) + ReplaceTo$ + Mid$(Text$, Pos + Len  
(ReplaceFrom$))  
        Pos = InStr(Text$, ReplaceFrom$)  
    Loop
```

End Sub

Sub WordWrapCode ByRef Text\$, length

```
    CurrStart = 1  
    Do  
        CurrEnd = InStr(Text$, Chr$(13) + Chr$(10), CurrStart)  
        If (CurrEnd = 0) Then CurrEnd = (Len(Text$) + 1)  
        CurrLength = (CurrEnd - CurrStart - 1)  
        If (CurrLength < length) Then  
            CurrStart = (CurrEnd + 2)  
        Else  
            CurrEnd = CurrStart  
            Do  
                LastEnd = CurrEnd  
                CurrEnd = InStr(Text$, " ", CurrEnd + 1)  
                If (CurrEnd = 0) Then  
                    CurrEnd = (Len(Text$) + 1)  
                    Exit Do  
                End If  
                CurrLength = (CurrEnd - CurrStart - 1)  
            Loop While (CurrLength < length)  
            Text$ = Left$(Text$, LastEnd) + Chr$(13) + Chr$(10) + Mid$(Text$  
, LastEnd + 1)  
            CurrStart = (LastEnd + 3)  
        End If  
    Loop While (CurrStart < Len(Text$))
```

End Sub

Function GetFileContents\$(Filename\$)

Open Filename\$ For Input As #2

GetFileContents\$ = Input\$(#2, LOF(#2))

Close #2

End Function