

REQUIRES LBBROWSE3.DLL available here:

<http://alycesrestaurant.com/lbbrowse.htm>

Place the DLL in the same folder as the BAS, or Windows System folder.

I suggest tokenizing the code. To do this:

- Open the code in the Liberty BASIC IDE.
- In the RUN menu, select "Make TKN File"
- Save the TKN and be sure the DLL is in the same folder or the Windows System folder.
- In the SETUP menu, find the option for "External Programs"
- In the dialog to set up external programs, browse to the TKN file and select it.
- After you restart Liberty BASIC, you'll find the help search program listed at the bottom of the RUN menu. It will be very easy to access the helpfile now, and to search it.

Last update:

May 15, 2011

```
'Copyright 2011, Alyce Watson
'http://alycesrestaurant.com/
'
'REQUIRES LBBROWSE3.DLL available here:
'http://alycesrestaurant.com/lbbrowse.htm
'Place DLL in program folder, or Windows System folder.
'
'This code is free for personal use.
'You may not republish this code in its current form.
'You may modify this code for your personal use.
'You may publish a modified version of this code under
'  these conditions:
'    1. You have made major changes to the code.
'    2. You give credit to the original author
'
'Thanks to the many people in the Liberty BASIC community who have sha
red code.
'This program contains code and ideas inspired by all that I've learne
d from others.
'I'd especially like to thank Janet, Stefan, Chris and Rod for help an
d
'inspiration for this program.
'An old LBFriend called Michael Rankin once wrote to me that
'"Knowledge is a gift we receive from others."
'I used that quote at the top of many of the early newsletter issues.
'It describes the spirit of our community perfectly. Thanks to all.

'initialize variables and arrays
```

```
nomainwin
WindowWidth=800:WindowHeight=500
UpperLeftX=10:UpperLeftY=10
installDir$=GetFolder$(GetModuleFileName$())
helpDir$ = installDir$+"lb4help\LibertyBASIC_4_web\HTML\"
winTitle$ = "Liberty BASIC Help Search Engine"
helpurl$ = installDir$+"lb4help\LibertyBASIC_4.html"
selfile$ = helpurl$
'selfile$ is filename of currently displayed file

dim info$(10,10)
'get file info into array
files helpDir$, "*.htm", info$()
numFiles = val(info$(0,0))
if numFiles=0 then end
dim topics$(20)
topics$(1)="Contents";chr$(0);helpurl$

graphicbox #1.g, 170, 34, 300, 40
stylebits #1.g, 0, _WS_VISIBLE, 0, 0
statictext #1.label1, "Search term:",10,10,80,16
button #1.go, "Display",[doSearch],UL,105,4,55,26
Stylebits #1.go, _BS_DEFPUSHBUTTON, 0, 0, 0
button #1.back, "<",[back],UL,170,4,30,26
button #1.forward, ">",[forward],UL,202,4,30,26
button #1.home, "TOC",[goHome],UL,240,4,35,26
button #1.index, "Index",[getIndex],UL,275,4,35,26
button #1.font, "Text Size",[doFont],UL,310,4,70,26
button #1.print, "Print Topic",[doPrint],UL,380,4,70,26
button #1.annotate, "Annotate",[doAnnotate],UL,450,4,70,26
button #1.source, "View Source",[viewSource],UL,520,4,70,26
button #1.help, "Help",[help],UL,590,4,50,26
button #1.about, "About",[about],UL,640,4,50,26
textbox #1.search,10,30,150,26
listbox #1.topics,topics$(),[doDisplay],10,60,150,400
open winTitle$ for window as #1
    #1 "trapclose [quit]"
    #1 "resizehandler [resizeMe]"
    #1 "font ms_sans_serif 8"
    #1.back "!font ms_sans_serif 14 bold"
    #1.forward "!font ms_sans_serif 14 bold"
    #1.search "Type search term."
    #1.g
"down;fill lightgray;backcolor lightgray;color red; font arial 14 bold
"
    #1.g "\\          Working... please wait."
```

```
#l.g "flush"

open "lbbrowse3.dll" for dll as #b
hMain = hwnd(#1)

calldll #b, "CreateBrowser", hMain as ulong, 170 as long, _
    34 as long, 610 as long, 426 as long, _
    helpurl$ as ptr, 0 as long, result as ulong

if result<>0 then
    notice "Unable to load browser control."
    goto [abort]
end if
calldll #b, "ShowStatusbar", 0 as long, re as void

#l.search "!setfocus"

'setfocus so that user keystrokes are directed to textbox when program
opens
'select all text in textbox so user will replace it when he types
hEdit=hwnd(#l.search)
CallDLL #user32, "SendMessageA",_
    hEdit As ulong, _      'handle of EDIT control
    _EM_SETSEL As long,_   'message
    0 As long,_            'start position of highlighting
    -1 As long,_           'last char of highlighting
    result As long

'invoke resizehandler
hMain=hwnd(#1)
calldll #user32, "MoveWindow",_
    hMain as ulong, _      'window handle
    10 as long,_           'x location of window
    10 as long,_           'y location of window
    790 as long,_          'desired width of window
    500 as long,_          'desired height of window
    1 as long,_            'repaint flag,0=false,1=true
    ret as long             'nonzero=success
gosub [enableButtons]
wait

[quit] calldll #b, "DestroyBrowser", re as void
[abort] close #b:close #1:end

[doDisplay]
'retrieve text after the null character - this is the filename
```

```
#1.topics, "selection? sel$"  
selfile$=word$(sel$,2,chr$(0))  
calldll #b, "Navigate", selfile$ as ptr, ret as void  
gosub [enableButtons]  
wait
```

```
[goHome]  
calldll #b, "Navigate", helpurl$ as ptr, ret as void  
gosub [enableButtons]  
wait
```

```
[getIndex]  
#1.g "show"  
redim topics$(numFiles)  
for i = 1 to numFiles  
    scan  
    open helpDir$ + info$(i,0) for input as #file  
    lenFile=lof(#file)  
    txt$=INPUT$(#file, lenFile)  
    topics$(i)=GetTitle$(txt$)+ chr$(0)+ helpDir$ + info$(i,0)  
    close #file  
next  
sort topics$( ),1,numFiles  
#1.topics "reload"  
#1.g "hide"  
wait
```

```
[back]  
calldll #b, "CanGoBack", goback as long  
if goback=1 then calldll #b, "Back", re as void  
gosub [enableButtons]  
wait
```

```
[forward]  
calldll #b, "CanGoForward", goforward as long  
if goforward=1 then calldll #b, "Forward", re as void  
gosub [enableButtons]  
wait
```

```
[enableButtons]  
ptitle$=space$(100)  
calldll #b, "GetTitle",_ 'get page title  
    ptitle$ as ptr,_ 'string buffer will hold page title  
    re as void 'no return  
ptitle$=trim$(ptitle$)  
if ptitle$="" then ptitle$="Liberty BASIC Help"
```

```
pCaption$=winTitle$;" --- ";ptitle$
```

```
callDll #user32, "SetWindowTextA",_
    hMain as ulong,_ 'handle of window
    pCaption$ as ptr,_ 'new titlebar text
    re as long
```

```
callDll #b, "CanGoBack", goback as long
if goback=1 then
    #1.back "!enable"
else
    #1.back "!disable"
end if
```

```
callDll #b, "CanGoForward", goforward as long
if goforward=1 then
    #1.forward "!enable"
else
    #1.forward "!disable"
end if
return
```

[doFont]

```
prompt "Enter desired font size. Valid numbers are 1 - 5. ";fsize$
if fsize$="" then wait
size=int(val(fsize$))
if size<1 then size=1:if size>5 then size=5
callDll #b, "DoFontSize",_ 'set font size, 1-5
    size as long,_ '1-5 1=small, 5=large
    re as void 'no return
wait
```

[doPrint]

```
callDll #b, "BrowserPrint",_ 'print hard copy of page
    re as void 'no return
wait
```

[about]

```
notice
```

```
"Liberty BASIC Help Search Engine ";chr$(169);" Alyce Watson 2011"
wait
```

[help]

```
instruc$=
```

```
"<html><head><title>Instructions</title></head><body><h1>Instructions<
/h1><p>"
```

```
    instruc$=instruc$+
"<p>Type a search term, then hit enter or the Display button. Double c
lick on a topic from the listbox to display it in the browser control.
"
    instruc$=instruc$+
"<p>The TOC button causes the help system's main page to be displayed.
"
    instruc$=instruc$+
"<p>The Index button causes an alphabetical index of pages to be displ
ayed in the listbox."
    instruc$=instruc$+
"<p>Annotations are added by typing a note, which will be added to the
bottom of the current topic in red text."
    instruc$=instruc$+
"<p>View Source will cause the source code for the current topic to be
displayed in the default HTML editor."
    instruc$=instruc$+
"<p>Print Topic will offer the option to print a hard copy of the curr
ent topic."
    instruc$=instruc$+
"<p>The topic text size can be changed by hitting the Text Size button
. Select sizes 1-5"
    instruc$=instruc$+
"<p><p><p><i>Liberty BASIC Help Search Engine copyright 2011, Alyce Wa
tson</i></body></html>"
    calldll #b, "BrowserString",_    'load html page from memory string
        instruc$ as ptr,_    'string containing html code
        ret as void          'no return
    wait

[doAnnotate]
    if selffile$="" then wait
    prompt "Type notes here.";notes$
    if notes$="" then wait
    open selffile$ for append as #g
        print #g,
"<p><!--annotation--><html><font color=red>";notes$;"</font></html>"
    close #g
    calldll #b, "Navigate", selffile$ as ptr, ret as void
    wait

[viewSource]
    calldll #b, "ViewSource",_    'view source in user's default program
        re as void              'no return
    wait
```

```
[doSearch]
    #1.search "!contents? term$"
    if term$="" then
        notice "Please type a search term."
        wait
    end if
    #1.g "show"
    term$=lower$(term$)
    redim topics$(numFiles) 'redim array to clear it
    count=0 : bestmatch$="" 'reset page to show after search

'In a loop, open each file and search with instr()
'If search term is found, extract title from HTML page.
'Concatenate title, null character, and path/filename and add to array
.
'null chr$(0) keeps filename out of user's view,
'but allows it to be retrieved for display.
    for i = 1 to numFiles
        scan
        open helpDir$ + info$(i,0) for input as #file
        lenFile=lof(#file)
        txt$=INPUT$(#file, lenFile)
        start=instr(lower$(txt$),term$)
        if start >0 then
            count=count+1
            topics$(count)=GetTitle$(txt$)+ chr$(0)+
helpDir$ + info$(i,0)
            if lower$(GetTitle$(txt$))=term$ then
'if search term is a topic title, show it
                bestmatch$=helpDir$ + info$(i,0)
            end if
        end if
        close #file
    next
    sort topics$( ),1,count
    #1.topics "reload"
    #1.g "hide"
    if count=0 then wait

'if no topic title exactly matches search term, see if one contains it
    if bestmatch$="" then
        for i = 1 to count
            if instr(topics$(i),term$) then
                bestmatch$=word$(topics$(i),2,chr$(0))
            exit for
        next i
    end if
```

```
        end if
    next
end if

'if there is still no bestmatch$, then display first topic in list
if bestmatch$="" then
    bestmatch$=word$(topics$(1),2,chr$(0))
end if
calldll #b, "Navigate", bestmatch$ as ptr, ret as void
gosub [enableButtons]
wait

[resizeMe]
width=WindowWidth-178:height=WindowHeight-42
calldll #b, "MoveBrowser",_
'relocate and resize browser control,
    170 as long,_    'new x location
    34 as long,_    'new y location
    width as long,_ 'new width
    height as long,_ 'new height
    ret as void      'no return
#1.topics "locate 10 60 150 ";WindowHeight-68
#1 "refresh"
wait

'extract title from html text
function GetTitle$(s$)
    marker1=instr(upper$(s$), "<A NAME=")
    marker1=instr(upper$(s$), "<B>",marker1)+3
    marker2=instr(upper$(s$), "</B>", marker1)
    lenTitle=marker2-marker1
    GetTitle$=mid$(s$,marker1,lenTitle)
end function

'borrowed from Stefan Pendl
function GetModuleFileName$()
    nSize = _MAX_PATH + 1
    lpFilename$ = space$(nSize); CHR$(0)
    calldll #kernel32, "GetModuleFileNameA",_
        hModule      as uLong,_
        lpFilename$ as ptr,_
        nSize        as uLong,_
        result       as uLong
    if result > 0 then GetModuleFileName$ = trim$(lpFilename$)
end function
function GetFolder$(Path$)
```



```
pos = 1
GetFolder$ = Path$
while pos > 0
    pos = instr(Path$, "\", pos)
    if pos > 0 then
        GetFolder$ = left$(Path$, pos)
        pos = pos + 1
    end if
wend
end function
```