Sons of Thunder Software
SOFTWARE DEVELOPMENT AND CONSULTING SERVICES

livecode

stsFormatFileListing

Usage

```
stsFormatFileListing(origFolder[,@pathList[,includeOrigFolder[,indentString[,folderFormat]]]])
put stsFormatFileListing("/Users/kenray/MyFolder",tPathList)
put stsFormatFileListing("/Users/kenray/MyFolder",tPathList,true,"    ","[folder]")
```

Description

This function takes the multiline, full path input from stsGetFileListing and formats it for screen display.

- ***origFolder*** is the same folder path used in stsGetFileListing's ***folderName*** parameter.
- ***@pathList*** is the list that is returned from stsGetFileListing. The reason this is a pass-by-reference (PBR) variable is that *pathList* is actually changed during processing so that it matches in a one-to-one correspondence with the formatted form (see example below).
- ***includeOrigFolder*** determines whether to include the original folder name in the formatted result. If you don't pass *includeOrigFolder*, the original folder at *origFolder* is not returned in the formatted result (i.e. you only see the contents of *origFolder*). If true (or non-empty), the first line of the formatted result is the folder name in origFolder.
- ***indentString*** is what to use to indent various levels of the folder hierarchy in the formatted result. If you don't pass indentString (or pass empty), you will have no indentation at all. A good indentation level is four spaces ("   "), however whatever indent string is passed, it will be used for each level of indention.
- ***folderFormat*** is a string that corresponds to the way you want folders to be displayed in the formatted result. The string must contain te word "folder", and the actual folder name replaces the string "folder" in folderFormat. So if *folderFormat* is "[folder]", the folder names in the formatted result will be surrounded with brackets. If not passed or empty, no folders are returned at all, and the lines that would normally be taken by the folder names in the result will be removed (this allows you to get a raw file name listing across multiple folders).

Example

If you had two paths you sent in (the result of stsGetFileListing):

```
MyDisk/MyFolder1/MyFile1
MyDisk/MyFolder2/MyFile2
```

You'd get this back as a result of the function (if includeOrigFolder is false, indentString is " ", and folderFormat is "folder"):

```
MyFolder1
   MyFile1
MyFolder2
   MyFile2
```

and the pass by reference (PBR) variable would contain:

```
MyDisk/MyFolder1/
MyDisk/MyFolder1/MyFile1
MyDisk/MyFolder2/
MyDisk/MyFolder2/MyFile2
```

Code

Select the code below and copy it to the clipboard, or retrieve it from Scripter's Scrapbook Online.

```
function stsFormatFileListing pOrigFolder,@pPathList,pIncludeOrigFolder,pIndentString,pFolderFormat
  put (pIncludeOrigFolder <> "" and pIncludeOrigFolder <>"false") into pIncludeOrigFolder
  if pIndentString = "" then put tab into pIndentString
  if "folder" is not in pFolderFormat then put "folder" into pFolderFormat
  if char -1 of pOrigFolder = "/" then delete char -1 of pOrigFolder
  put "" into tRetVal
  put "" into tNewPathList
  sort pData
  put length(pOrigFolder)+1 into tLen
  set the itemDel to "/"
  put "" into tPrevLine
  if pIncludeOrigFolder then
    put replaceText(pFolderFormat,"folder",item -1 of pOrigFolder) & cr into tRetVal
    put pOrigFolder & "/" & cr into pNewPathList
    put 1 into tStartLevel
  else
    put 0 into tStartLevel
  end if
  repeat for each line tLine in pPathList
    put tStartLevel into tLevel
    delete char 1 to tLen of tLine
    put the number of items of tLine into tNumItems
    repeat with x = 1 to tNumItems
      put item x of tLine into tItem
      if tItem = item x of tPrevLine then
        add 1 to tLevel
      else
        if (x <> tNumItems) or (x = tNumItems AND char -1 of tLine = "/") then  -- folder
          put _stsLevelPad(replaceText(pFolderFormat,"folder",tItem),tLevel,pIndentString) & cr after tRetVal
          put pOrigFolder & "/" & item 1 to x of tLine & "/" & cr after pNewPathList
        else
          put _stsLevelPad(tItem,tLevel,pIndentString) & cr after tRetVal
          put pOrigFolder & "/" & item 1 to x of tLine & cr after pNewPathList
        end if
        exit repeat
      end if
    end repeat
    repeat with y = (x+1) to tNumItems
      put item y of tLine into tItem
      add 1 to tLevel
      if (y <> tNumItems) or (y = tNumItems AND char -1 of tLine = "/") then  -- folder
        put _stsLevelPad(replaceText(pFolderFormat,"folder",tItem),tLevel,pIndentString) & cr after tRetVal
        put pOrigFolder & "/" & item 1 to y of tLine & "/" & cr after pNewPathList
      else
        put _stsLevelPad(tItem,tLevel,pIndentString) & cr after tRetVal
        put pOrigFolder & "/" & item 1 to y of tLine & cr after pNewPathList
      end if
    end repeat
    put tLine into tPrevLine
  end repeat
  delete last char of tRetVal
  delete last char of pNewPathList
  put pNewPathList into pPathList
  return tRetVal
end stsFormatFileListing

function _stsLevelPad pWhat,pNum,pString
  repeat pNum times
    put pString before pWhat
  end repeat
  return pWhat
end _stsLevelPad/pre>
```

Print this code entry

News and Rumors Products Services Developer Resources Contact STS About STS

Send all comments to webmaster@sonsothunder.com.