

```

/*****
/*
/*-----
/* Task      : Creates a shortcut to the selected file on the
/*            Desktop.
/*-----
/* Authors    : Michael Tischer and Bruno Jennrich
/* developed on : 09/25/1995
/* last update  : 10/03/1995
/*****
#include <windows.h>
#include <shlobj.h>
#include "shortcut.h"

/*****
/* GetNumOfFiles : Gets the number of files returned in a
/*               multiple selection.
/*-----
/* Parameters:    pszFiles - Address of buffer filled by
/*               GetOpenFilename() with the filenames
/* Return value : Number of files in pszFiles separated by NUL
/*-----
/* Info : With multiple selections the first filename is the
/*        directory from which the files that follow were
/*        selected.
/*****
int GetNumOfFiles( LPSTR pszFiles )
{
    int iPos, iCnt, iLen;
    iPos = iCnt = 0;
    do
    {
        iLen = lstrlen( &pszFiles[ iPos ] );
        if( iLen )                // More filenames?
        {
            iCnt++;
            iPos += iLen + 1;      // Position at next name
        }
    } while( iLen );
    return iCnt;
}

/*****
/* GetFileOffsetExt : Gets the beginning of the filename and the
/*                  beginning of the file extension within a file
/*                  path.
/*-----
/* Parameter :    pszFile      - File path
/*                pFileOffset - Address of INT variables for receiving
/*                the beginning of the filename
/*                pFileExt     - Address of INT variables for receiving
/*                the beginning of the file extension
/* Return value : Length of the whole file path
/*****
int GetFileOffsetExt( LPSTR pszFile, int *pFileOffset, int *pFileExt )
{
    int iLen;
    iLen = lstrlen( pszFile );
    *pFileOffset = iLen;
    *pFileExt = iLen;

    // Look for last backslash -----
    while( ( ( *pFileOffset - 1 ) >= 0 ) &&
           ( pszFile[ *pFileOffset - 1 ] != '\\\' ) )
        (*pFileOffset)--;
    // Look for last '.' -----
    while( *pFileExt &&
           ( pszFile[ *pFileExt ] != '.' ) )
        (*pFileExt)--;
    // If no period is found, it is either a directory or
    // a file without an extension.
    if( !(*pFileExt) ) *pFileExt = iLen;
    return iLen;
}

/*****

```

```

/* CreateMultipleShortcuts : Creates multiple shortcuts */
/*-----*/
/* Parameters:      lpLocationDir - Directory in which shortcut files */
/*                  are to be placed. */
/*                  lpFileNames   - Address of buffer with filenames */
/*                  This buffer is in the format */
/*                  used by GetOpenFilename() for */
/*                  multiple selection. */
/* Return value : none */
/*-----*/
void CreateMultipleShortcuts( LPSTR lpLocationDir,
                             LPSTR lpFileNames )
{
    char szLocation[ MAX_PATH ];
    char szPath[ MAX_PATH ];
    int iPos = 0, iLen, iCnt;
    char cBackUp;

    // Get number of selected files -----
    iCnt = GetNumOfFiles( lpFileNames );

    // If more than one file was passed in the buffer, the first
    // "filename" stands for the directory from which the files that
    // follow were selected. Therefore, we need to skip this
    // directory.
    if( iCnt > 1 ) iPos = strlen( lpFileNames ) + 1;
    else          iPos = 0;

    do
    {
        iLen = strlen( &lpFileNames[ iPos ] );
        if( iLen )
        {
            int iFileOffset;
            int iFileExt;

            // Put together name of shortcut file -----
            strcpy( szLocation, lpLocationDir );
            strcat( szLocation, "\\Shortcut to " );

            // Get beginning of filename and file extension -----
            iLen = GetFileOffsetExt( &lpFileNames[ iPos ],
                                    &iFileOffset,
                                    &iFileExt );

            // Replace '.' with '\0'
            cBackUp = lpFileNames[ iPos + iFileExt ];
            lpFileNames[ iPos + iFileExt ] = '\0';

            // Attach filename to shortcut name -----
            strcat( szLocation, &lpFileNames[ iPos + iFileOffset ] );
            lpFileNames[ iPos + iFileExt ] = cBackUp;

            // Shortcut file has the LNK extension -----
            strcat( szLocation, ".lnk" );

            // Put together name of file to which the shortcut is referring
            strcpy( szPath, lpFileNames );
            if( iCnt > 1 ) // With multiple selection you also need
            {           // to add the name (without the path).
                strcat( szPath, "\\");
                strcat( szPath, &lpFileNames[ iPos ] );
            }

            // Create shortcut -----
            CreateShortcut( szPath, //pszPath
                           "",      //pszArguments
                           szLocation, //pszLocation
                           "",      //pszWorkingDir
                           SW_SHOW ); //Flags

            iPos += iLen + 1; // Position at next filename
        }
    }while( iLen );
}

/*-----*/

```

```

/* WinMain - Start function */
/*-----*/
/* Parameters:    default parameters */
/* Return value : default return value */
/*-----*/
int WINAPI WinMain( HINSTANCE hInst,
                   HINSTANCE hInstPrev,
                   LPSTR      lpCmdLine,
                   int         nCmdLine )
{
    char g_szFileNames[ MAX_PATH * 20 ];
    OPENFILENAME ofn;                                // Structure for opening file

    g_szFileNames[ 0 ] = '\0';
    ofn.lStructSize = sizeof( ofn );
    ofn.hwndOwner = NULL;
    ofn.hInstance = hInst;
    ofn.lpstrFilter = "All files (*.*)\0*.*";
    ofn.lpstrCustomFilter = NULL;
    ofn.nMaxCustFilter = 0;
    ofn.nFilterIndex = 0;
    ofn.lpstrFile = g_szFileNames;                  // Buffer for file names
    ofn.nMaxFile = sizeof( g_szFileNames );
    ofn.lpstrFileName = 0;
    ofn.nMaxFileName = 0;
    ofn.lpstrInitialDir = NULL;                      // Initial directory
    ofn.lpstrTitle = "File for Desktop Link";         // Dialog Title
    ofn.Flags = OFN_EXPLORER | OFN_ALLOWMULTISELECT; // Win95 style, mult.
    ofn.nFileOffset = 0;
    ofn.nFileExtension = 0;
    ofn.lpstrDefExt = "";
    ofn.lCustData = 0;
    ofn.lpfnHook = NULL;
    ofn.lpTemplateName = NULL;

    if( GetOpenFileName( &ofn ) )                   // Display OpenFile dialog box
    {
        LPSHELLFOLDER lpDesktop;

        CoInitialize( NULL );
        // First get the ShellFolder interface of the Desktop -----
        if( SHGetDesktopFolder( &lpDesktop ) == ERROR_SUCCESS )
        {
            LPITEMIDLIST pidl;

            // Get PIDL of desired special folder -----
            if( SHGetSpecialFolderLocation( NULL,
                                           CSIDL_DESKTOPDIRECTORY,
                                           &pidl ) == NOERROR )
            {
                char szDesktopPath[ MAX_PATH ];

                szDesktopPath[ 0 ] = '\0';
                // Get path of Desktop directory -----
                SHGetPathFromIDList( pidl, szDesktopPath );

                // Create shortcuts -----
                CreateMultipleShortcuts( szDesktopPath, g_szFileNames );

                CoTaskMemFree( pidl );                 // Release PIDL
            }
            lpDesktop->lpVtbl->Release( lpDesktop );    // Release interface
        }
        CoUninitialize();                             // Uninstall OLE
    }
    return 0;
}

```