

File Library - [lbjoseph](#)

Below you can find some simple functions for working with files and file paths.

The functions are pretty self-explanatory, and the comments detail the use of them.

```
Print File.Path$("C:\Some Path\In\Some\Obscure Folder\some file.txt")
' Prints: C:\Some Path\In\Some\Obscure Folder
Print File.Name$("some setting file.txt")
' Prints: some setting file.txt
Print File.Name$("C:\Some Path\some file.txt")
' Prints: some file.txt
Print File.Exists("some_file.txt")
' Prints 1 if the file some_file.txt exists in the current directory.
Print File.Exists("C:\some_file.txt")
' Prints 1 if the file some_file.txt exists in C:\.
```

```
Function File.Exists(str$)
    ' Returns non-zero if the specified file exists.
    Dim File.Lib$(1,1)
    folder$ = File.Path$(str$)
    If folder$="" Then folder$ = DefaultDir$
    Files File.Path$(str$), File.Name$(str$), File.Lib$()
    File.Exists = Val(File.Lib$(0,0)) > 0
End Function
```

```
Function File.Path$(str$)
    ' Returns the folder portion of the path given.
    ' No trailing back-slash.
    original$ = str$
    While Right$(str$,1) <> "\" And str$ <> ""
        If Right$(str$,1) = "." Then foundFile = 1
        str$ = Left$(str$, Len(str$)-1)
    Wend
    File.Path$ = original$
    If foundFile Then File.Path$ = Mid$(str$,1,Len(str$)-1)
End Function
```

```
Function File.Name$(str$)
    ' Returns the filename portion of the path given.
    pathLength = Len(File.Path$(str$))
    str$ = Right$(str$, Len(str$)-pathLength)
```

```
File.Name$ = str$  
If Left$(str$,1) = "\" Then File.Name$ = Mid$(str$,2)  
End Function
```