

## Re: Public variables and separate library files

**Source:** <http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2004-09/9476.html>

---

**From:** Tom Ogilvy (*twogilvy\_at\_msn.com*)

**Date:** 09/28/04

Date: Tue, 28 Sep 2004 07:05:06 -0400

You would need to create a reference from the file needing to use the variable to the file containing the variable. If you do create the reference, then if you open the file containing the reference, the file referenced will also open and you can not close the file referenced until the file with the reference is closed.

An alternative is to have a function in the file containing the variable return the value of the variable, then call that function using Application.Run. This would be a one way path, however.

Essentially, variables are local to the workbook in which they are contained.

```
--
Regards,
Tom Ogilvy
"Richard" <ra3@sbcglobal.net> wrote in message
news:3a1fe86a.0409272157.2411c159@posting.google.com...
> How does one make a variable available to other routines stored in a
> separate library-type file? I thought using "Public", but in the
> simple example below the variable "temp" is null within the called
> subroutine.
>
> Calling file.
> Public temp As Integer
> Sub Master()
> Workbooks.Open ("\\Temp\\Library1.xls")
> temp = 0
> Again:
> MsgBox ("Before:" & temp)
> Application.Run "Library1.xls!TestSub"
> MsgBox ("After:" & temp)
> temp = temp + 1
> GoTo Again
> End Sub
> File "\\Temp\\Library1.xls".
> Sub TestSub()
> MsgBox ("Within:" & temp)
> End Sub
> Output produced.
> Before:0
> Within:
> After:0
> Before:1
```

microsoft.public.excel.programming: Re: Public variables and separate library files

```
> Within:
> After:1
> Before:2
> Within:
> After:2
> etc.
>
> If I add a "Public" statement at the top of the Library1.xls module, I
> get zeros instead of nulls within the called routine. (It works
> correctly if I concatenate the two routines into the same file, but
> this of course defeats the purpose of having separate libraries of
> code).
```