

Re: Macros

Source: <http://www.tech-archive.net/Archive/Word/microsoft.public.word.vba.beginners/2004-08/0132.html>

From: Jay Freedman (jay.freedman_at_verizon.net)

Date: 08/10/04

Date: Tue, 10 Aug 2004 15:05:57 -0400

Hi Tim,

It looks like you're in a good position to use the AutoText-and-macros setup I described before. There shouldn't be any complications.

For each of the two includable pages, with tables containing form fields, select the whole page and use the Insert > AutoText > AutoText dialog to save it as an entry in the template. Pay careful attention to the Look In box at the bottom of the dialog -- make sure it shows your template and not "All active templates". See <http://www.word.mvps.org/faqs/customization/AutoText.htm> for some useful background.

I'm not sure whether to recommend MacroButton fields directly in the table, or buttons on a custom toolbar, to launch the macros. Either should work.

You'll want two macros in the template, one for each type of insert. The general idea of the code is

```
Sub AddPage2Table()  
    Dim oRg As Range  
  
    If ActiveDocument.ProtectionType <> _  
        wdNoProtection Then  
        ActiveDocument.Unprotect  
    End If  
  
    Set oRg = ActiveDocument.Range  
    oRg.Collapse wdCollapseEnd  
  
    ActiveDocument.AttachedTemplate _  
        .AutoTextEntries("page2").Insert _  
        Where:=oRg, RichText:=True  
  
    ActiveDocument.Protect _  
        Type:=wdAllowOnlyFormFields, _  
        NoReset:=True  
End Sub
```

microsoft.public.word.vba.beginners: Re: Macros

This will insert the AutoText entry named "page2" at the end of the document. You'd need a second macro that's the same except for the name of the entry it inserts. You could add code to make sure the document doesn't get more than four copies of one entry and one copy of the other -- but often it's better just to trust the users to do it right.

There are probably some gotchas lurking in there, but they shouldn't be too hard to work around.

--

Regards,

Jay Freedman

Microsoft Word MVP

FAQ: <http://word.mvps.org>

Tim Craig wrote:

> Thank you Jay,

>

> I apologize for making your responses more difficult, it
> certainly is not my intention to do so! The information
> that you are providing is invaluable to me in my attempt
> to get my document/form working for our internal customers.

>

> In response to your questions:

> - What version of Word are you using?

> Word 2002 (10.5815.4219)SP-2

>

> - Do all of the prospective users have the same version of
> Word, or will some of them have other versions?

> YES, Sprint pushes updates to each machine nightly

> to insure that we have the same version of most

> applications.

>

> - What do you do with the form after it's filled in?

> The completed form is saved under a unique file

> name to a document control system that is available to

> users requiring access to the documents.

>

> - Do you just save and/or print the document, or do you

> try to extract data from the fields into a spreadsheet or

> database?

> As far as I know there is no attempt to extract

> data into a spreadsheet or database.

>

> On occasion it is necessary to retrieve the stored file,

> make changes and or additions then resave the document

> back to the document control system.

>

>

> Tim Craig

>

>> -----Original Message-----

>> Hi Tim,

>>

>> Gee, I love pulling teeth... A document with form fields is

>> distinctly different from a document that just contains a table. Do

>> you have this template "protected for forms" when you use it?

>>

>> Most of my previous advice still stands: store each

> table, or piece of

>> table, as an AutoText entry in the main template (which becomes the

>> only template). As part of that AutoText, include a MacroButton in

microsoft.public.word.vba.beginners: Re: Macros

```
>> the last cell that invokes a macro, also in the same template.
>>
>> The difference is that the macro will need to unprotect the document
>> first; then insert the appropriate AutoText entry; and finally
>> reprotect the document, using the NoReset:=True argument to avoid
>> clearing all the fields that have already been filled in.
>>
>> As long as we're still discovering important bits of the problem:
>>
>> - What version of Word are you using?
>> - Do all of the prospective users have the same version
> of Word, or
>> will some of them have other versions?
>> - What do you do with the form after it's filled in? Do you just save
>> and/or print the document, or do you try to extract data from the
>> fields into a spreadsheet or database?
>>
>> The last item determines whether your macro needs to
> worry about the
>> names of the fields. Inserting the same AutoText containing fields
>> several times leads to fields that don't have names,
> which causes the
>> kind of problem discussed at
>> http://word.mvps.org/FAQs/MacrosVBA/AssignNameToFmFld.htm.
>>
>> --
>> Regards,
>> Jay Freedman http://aspnet2.com/mvp.ashx? JayFreedman
>> Microsoft Word MVP FAQ: http://word.mvps.org
>>
>> "Tim Craig" <anonymous@discussions.microsoft.com> wrote:
>>
>>> Thank you for your quick response.
>>>
>>> Since I'm a Super Novice trying to do operations well
>>> beyond my level of expertise, I would have to say that I'm
>>> the one that's confused.
>>>
>>> 1. Following is the "Insert" snippet from the macro.
> This
>>> is an exit macro from a field on page two of the main
>>> template.
>>>
>>> Selection.InsertFile FileName:=ActiveDocument.Path
>>> & "\Manage_Report.dot", Range:="", _
>>> ConfirmConversions:=False, Link:=False,
>>> Attachment:=False
>>>
>>> So, your assumption is correct.
>>>
>>> The first document is currently a (.dot file). This
> where
>>> I originally had all four of my macros.
>>>
>>> There are actually three separate documents that
> comprise
>>> this "Word application".
>>> The second page is attached when needed may be "attached
>>> up to 4 times", each constituting a new and different
>>> completed form when attached.
>>> The third page is attached only once when needed.
>>>
```

microsoft.public.word.vba.beginners: Re: Macros

```
>>> Each of these files is an entire page and attached to
> the
>>> first "if-and-only-if" it is needed for completing a
>>> request.
>>>
>>> 2. Each "attached page" consists of a table that the user
>>> can extend as needed. So, in my inexperienced mode I
>>> created a macro and need to attach it to the last column
>>> in the last row of the table. The reason that I created
>>> the macro is: "I discovered that I could indeed add a
> new
>>> row to the table at the end of the table simply by
>>> pressing the "tab" key, but I do NOT get the fields that
>>> the user must enter their data into when that is all
> that
>>> I do. So this macro creates the necessary data entry
>>> fields for the user.
>>>
>>>
>>>> -----Original Message-----
>>>> Hi Tim,
>>>>
>>>> One or both of us is mightily confused. It may be just
> a
>>> matter of
>>>> terminology, but somehow I don't think so...
>>>>
>>>> 1. Exactly what are you doing to "attach" the second document to
>>>> the first? In Word and VBA, the only meaning of "attach" that
> sounds
>>> similar to your
>>>> task is setting a new value for the
>>> ActiveDocument.AttachTemplate property.
>>>> But this doesn't put any text (or tables or whatever) into an
>>>> existing document, it just makes things like macros and AutoText
>>>> entries available. So I suspect you're using something else, such
>>>> as Selection.InsertFile. That does insert text from the second
>>>> file, but it doesn't bring in any macros.
>>>>
>>>> I suspect that what you really want is this: Resave the first
>>>> document as a template (.dot file) and put both macros in it.
>>>> Also, save the piece of table from the second document as an
>>>> AutoText entry in the first document (now a template). See
>>>> http://www.word.mvps.org/faqs/customization/AutoText.htm .
>>>>
>>>> Use File> New to create Throw away the second
> document,
>>> which is no longer
>>>> necessary. Change the macros so that, instead of using InsertFile,
>>>> they use ActiveDocument.AttachedTemplate _
>>>> .AutoTextEntries("TablePiece").Insert (... )
>>>> to paste in a new chunk of table (although you should
> ask
>>> yourself "why not
>>>> let the user just hit Tab at the end of the row to make
> a
>>> new row?") Use
>>>> File> New to create a document based on the template,
> and
>>> away you go.
>>>>
>>>> 2. What do you mean by "in the last row and column is a macro"? Do
```

microsoft.public.word.vba.beginners: Re: Macros

>>>> you mean there's a command button or a MacroButton field there?
>>>> That can be part of the AutoText entry. If you need to get rid of
>>>> the one
> the
>>> user just clicked,
>>>> that can be part of the macro. Exactly how the code should look
>>>> depends on what gizmo you're using.
>>>>
>>>> --
>>>> Regards,
>>>> Jay Freedman
>>>> Microsoft Word MVP FAQ: <http://word.mvps.org>
>>>>
>>>> Tim.Craig wrote:
>>>>> I have two problems that that I need help solving.
>>>>>
>>>>> 1. I have two documents with macros. The first document
>>>>> attaches the second document - wich I have working
>>>>> correctly. The problem is that the macro in the second
>>>>> document does not remain available when I attach the
>>>>> document to the first. How do I fix this problem??
>>>>>
>>>>> 2. The second document is basicly filled with a two
>>>>> column table in the last row and column is a macro that
>>>>> contatinates a new line to the end of the page for as many
>>>>> times as the user needs. How do I add the macro to each
>>>>> new line as it is added???
>>>>>
>>>>> If anyone can help I would be extremely grateful.
>>>>>
>>>>> Tim Craig
>>>>
>>>>
>>>> .
>>>>
>>
>> .