

Re: Deploying custom toolbar and custom icons

Source:

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2008-07/msg03702.html>

- *From:* "ker_01" <ker_01@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Fri, 25 Jul 2008 08:08:24 -0400
-

Thanks again Dave! I've adapted this code, and pasted my icons on a hidden sheet. Everything is working (almost) perfectly; the only downside is that the pasted icons don't seem to have a transparent background anymore, and the background gray used is a little different from the toolbar color. This is an internal project so I'm not worried, and if it bugs me too much I'll just figure out which color matches the toolbar grey and go backfill the icons manually.

Thanks!!

Keith

"Dave Peterson" <petersod@xxxxxxxxxxxxxxxxxxxx> wrote in message <news:4888C6A8.DFCC6F41@xxxxxxxxxxxxxxxxxxxx>

Here's one that uses pictures on a worksheet -- nicely named!

```
Option Explicit
Public Const ToolBarName As String = "MyToolBarName"
Sub Auto_Open()
Call CreateMenubar
End Sub
Sub Auto_Close()
Call RemoveMenubar
End Sub
Sub RemoveMenubar()
On Error Resume Next
Application.CommandBars(ToolBarName).Delete
On Error GoTo 0
End Sub
Sub CreateMenubar()

Dim iCtr As Long

Dim MacNames As Variant
Dim CapNames As Variant
Dim TipText As Variant
Dim PictNames As Variant
Dim PictWks As Worksheet
```

Re: Deploying custom toolbar and custom icons

Call RemoveMenubar

```
MacNames = Array("aaa", _  
"bbb")
```

```
CapNames = Array("AAA Caption", _  
"BBB Caption")
```

```
TipText = Array("AAA tip", _  
"BBB tip")
```

```
PictNames = Array("Pic1", "Pic2")
```

```
Set PictWks = ThisWorkbook.Worksheets("Pictures")
```

```
With Application.CommandBars.Add  
.Name = ToolBarName  
.Left = 200  
.Top = 200  
.Protection = msoBarNoProtection  
.Visible = True  
.Position = msoBarFloating
```

```
For iCtr = LBound(MacNames) To UBound(MacNames)  
PictWks.Pictures(PictNames(iCtr)).Copy  
With .Controls.Add(Type:=msoControlButton)  
.OnAction = "" & ThisWorkbook.Name & "!" & MacNames(iCtr)  
.Caption = CapNames(iCtr)  
.Style = msoButtonIconAndCaption  
.PasteFace  
.TooltipText = tip_text(iCtr)  
End With  
Next iCtr
```

```
End With  
End Sub  
Sub AAA()  
MsgBox "aaa"  
End Sub  
Sub BBB()  
MsgBox "bbb"  
End Sub
```

The AAA and BBB subs are just stubs. You can put your macro code that does the real work there—or call your macros from them.

ker_01 wrote:

Thank you Dave— it looks like an add-in will be the best way to deploy my

Re: Deploying custom toolbar and custom icons

toolbar. I'll use code to install and load (and unload) the add-in from my actual workbook.

The one piece I'm still struggling with is whether there is any way to keep the custom button icons that I've already developed. I edited my icons within the Excel icon editor, so I don't have them saved anywhere as a separate file. Is my only option to recreate each 16x16 (bitmap) icon in an image editing program, or is there any way to grab a button image (perhaps by VBA) and save that image to the worksheet so I can copy it into my new add-in?

Thanks for any advice,
Keith

"Dave Peterson" <petersod@xxxxxxxxxxxxxxxxxxxx> wrote in message news:4887B5D5.32865334@xxxxxxxxxxxxxxxxxxxx

For additions to the worksheet menu bar, I really like the way John Walkenbach does it in his menumaker workbook: <http://j-walk.com/ss/excel/tips/tip53.htm>

Here's how I do it when I want a toolbar: <http://www.contextures.com/xlToolbar02.html> (from Debra Dalglish's site)

And if you use xl2007:

If you want to learn about modifying the ribbon, you can start at Ron de Bruin's site: <http://www.rondebruin.nl/ribbon.htm> <http://www.rondebruin.nl/qat.htm> -- For macros for all workbooks (saved as an addin) or <http://www.rondebruin.nl/2007addin.htm>

In xl2007, those toolbars and menu modifications will show up under the addins.

Re: Deploying custom toolbar and custom icons

ker_01 wrote:

I'm working on a custom toolbar for a workbook I'm building (plenty of buttons, including custom icons). We've decided to stick the workbook on the network and have other folks access it as well. What is the preferred method for ensuring the toolbar is available for all users when they open the workbook?

It's been years since I had to deploy a custom toolbar, and that was in Word97 via a template– whereas this is in Excel2003, and it is an active workbook (not a template). I googled and found a lot about building a custom toolbar via code, but there were references to issues with custom icons, and I'm hoping there is a way to just keep the toolbar connected to the workbook without having to generate it via code for each user.

Thanks for any advice,
Keith

--

Dave Peterson

--

Dave Peterson