

Re: TextFrame and VBA

Source: <http://www.tech-archive.net/Archive/Office/microsoft.public.powerpoint/2005-01/2084.html>

From: Steve Rindsberg (*abuse_at_localhost.com*)

Date: 01/17/05

Date: Mon, 17 Jan 2005 01:18:08 EST

In article <879mu01d832p03c1bcenr36p58foljaq9@4ax.com>, Jeff Jones wrote:

> *I expected to find this bridge as well.*
>
> *There are three lines of text and the initial characters in each line*
> *can uniquely identify each. One starts with Americas, another with*
> *Asia and the third with Europe. Each line will always be there and*
> *should always be on the last slide.*
>
> *I wandered through various code examples looking for a character*
> *string but couldn't figure out how to marry that code to the code to*
> *set a hypertext link.*

Hm. Something like this (off top of head, may require a bit of tweakage):

Function WheresWaldosTextBox(oPresentation As Presentation) As Shape
' Returns the text box you need to work with

Dim oSh As Shape

```
' Look at the last slide in the presentation's shapes
For Each oSh In oPresentation.Slides(oPresentation.Slides.Count).Shapes
    If Mid$(oSh.TextFrame.TextRange.Paragraphs(1), 1, Len("America")) = "America" Then
        If Mid$(oSh.TextFrame.TextRange.Paragraphs(2), 1, Len("Japan")) = "Japan" Then
            If Mid$(oSh.TextFrame.TextRange.Paragraphs(3), 1, Len("Korea")) = "Korea" Then
                Set WheresWaldosTextBox = oSh
                Exit Function
            End If
        End If
    End If
Next ' Shape

End Function
```

```
Sub testWaldo()
    MsgBox WheresWaldosTextBox(ActivePresentation).TextFrame.TextRange.Text
End Sub
```

>
> *There's no reason why I can't create the TextFrame myself although,*
> *I'd need to drop the existing object before creating my own.*
>
> *Jeff*
>
> *On Sun, 16 Jan 2005 19:24:56 EST, Steve Rindsberg*
> *<abuse@localhost.com> wrote:*
>
> *>In article <8fqlu0dr879oodmpv9snmlliatovdracos@4ax.com>, Jeff Jones wrote:*
> *>> Thank you Steve. It seems that I have one more hurdle to overcome.*
> *>> It turns out that the shape isn't always "Rectangle 17" but can be*
> *>> other numbers. Therefore the line of code "Set oAgenda =*
> *>> oSld.Shapes("Rectangle 17").TextFrame.TextRange" may or may not be*
> *>> correct and it seems that I can't trap the error. I'm going to have*
> *>> to figure out how to discern how to reference the text frame and*
> *>> range. Sigh.....*
>
> *>I figured we'd cross that bridge sooner or later. What's unique about this bit of*
> *>text that you can key off of? Will it always be there and if so, will it always have*
> *>some characteristic that distinguishes it from the other text on the slide?*
>
> *>Or can you create it yourself as needed?*
>
>
>
> *Jeff*
>
> *On Sun, 16 Jan 2005 14:01:21 EST, Steve Rindsberg*
> *<abuse@localhost.com> wrote:*
>
> *>In article <8c0lu01g5nqso2hb51882q3ln4mv2kmt8q@4ax.com>, Jeff Jones wrote:*
> *>>> LOL. I love to see people comrading.....*
> *>>>*
> *>>>The way Brian and I abuse one another, you'd think we're married. ;-)*
> *>>>*
> *>>>> Sub ChangeHyperLinkData()*
> *>>>>*
> *>>>> Dim oSld As Slide*
> *>>>> Dim oAgenda As TextRange*
> *>>>> 'Stop*
> *>>>> ActiveWindow.View.GotoSlide Index:=ActivePresentation.Slides.Count*
> *>>>> Set oSld =*
> *>>>> ActivePresentation.Slides(ActivePresentation.Slides.Count)*
> *>>>>*
> *>>>>*
> *>>>>You don't need to go to the slide (ie, to display it) in order to affect it, and*
> *>>>>in fact your code will run way faster if you don't.*
> *>>>>*
> *>>>>Set oSld = ActivePresentation.Slides(ActivePresentation.Slides.Count)*
> *>>>>*
> *>>>>is enough*

> > > >
> > > >
> > > > *ActiveWindow.Selection.SlideRange.Shapes("Rectangle 17").Select*
> > > > *ActiveWindow.Selection.ShapeRange.TextFrame.TextRange.Select*
> > > >
> > > > *With ActivePresentation.Slides(ActivePresentation.Slides.Count)*
> > > > *Set oAgenda =*
> > > > *ActiveWindow.Selection.ShapeRange.TextFrame.TextRange*
> > > > *End With*
> > > >
> > > >
> > > *Likewise, you don't need to select anything either. Just this will do it instead*
> > > *of all the above:*
> > > >
> > > *Set oAgenda = oSld.Shapes("Rectangle 17").TextFrame.TextRange*
> > > >
> > > >
> > > > *With oAgenda.Sentences(1) _*
> > > > *.ActionSettings(ppMouseClick).Hyperlink*
> > > > *.Address = "<http://www.gsms-am.eds.com/>*
> > > > *<<http://www.gsms-am.eds.com/>> "*
> > > >
> > > *and so on ...*
> > > >
> > > > *It works fine as long as I open the blank.pot file that contains the*
> > > > *macro. I've added a button to a new toolbar that executes the macro*
> > > > *to the blank.pot file and it is available when I open PowerPoint but*
> > > > *it's not effective until the blank.pot file itself has been opened.*
> > > >
> > > > *I'm playing with an add-in with the macro in it and can load the*
> > > > *add-in with no problems but the button is still not effective until*
> > > > *blank.pot is opened. Sigh.....*
> > > >
> > > > *Right. The button is associated with the macro in the original pot file in PPT's*
> > > > *mind. Add an Auto_Open subroutine to the add-in and include code in it to create*
> > > > *any necessary buttons/bars/menus.*
> > > >
> > > > *Create an ADD-IN with TOOLBARS that run macros*
> > > > *<http://www.rdpslides.com/pptfaq/FAQ00031.htm>*
> > > >
> > > > *Where I want to end up is to have a macro that will open every*
> > > > *presentation, in turn, in a folder, change the links on the last*
> > > > *slide, and save the presentation.*
> > > >
> > > > *Do something to every file in a folder*
> > > > *<http://www.rdpslides.com/pptfaq/FAQ00536.htm>*
> > > >
> > > >
> > > > *Do you suppose that I'm sneaking up on my objective? I'm FAR too lazy*
> > > > *to want to make all these changes manually. Besides, it's FAR, FAR*
> > > > *more fun to build a macro to automate tedious tasks.*

>>>>
>>>> *Thank you all for your help thus far. My beloved is certain that I'm*
>>>> *truly a computer geek and the fact that this wonderfulness if fun*
>>>> *supports my geekness. <giggle>*
>>>>
>>>> *Jeff*
>>>>
>>>> *On Sat, 15 Jan 2005 11:49:42 EST, Steve Rindsberg*
>>>> *<abuse@localhost.com> wrote:*
>>>>
>>>> *>In article <3cdiu01u3tss69q6dhp5pp7787l1rkkhj8@4ax.com>, MS MVP Brian Reilly*
>>>> *>wrote:*
>>>> *>> Jeff,*
>>>> *>> Here's another piece of code in addition to Steve's that I use all the*
>>>> *>> time as a wrapper to iterate through all shapes on all slides. Note,*
>>>> *>> I've commented out the section that you can change exactly what you*
>>>> *>> do, but the top and bottom parts are the iteration code.*
>>>> *>*
>>>> *>Brian reformats his hard drive quite often. Chances are he lost his mind the*
>>>> *>last time he did it. It's one of those pesky hidden files, ya know?*
>>>> *>*
>>>> *>Sub DiddleAllTheShapes()*
>>>> *>*
>>>> *>Dim oSh as Shape*
>>>> *>Dim oSl as Slide*
>>>> *>*
>>>> *>' Look at each slide*
>>>> *>For Each oSl in ActivePresentation.Slides*
>>>> *>' Look at each shape on the slide*
>>>> *> For each oSh in oSl.Shapes*
>>>> *>' Do whatever you need to with the shape*
>>>> *> With oSh*
>>>> *> Debug.Print .Name*
>>>> *>.Left = .Left + 10*
>>>> *>' .Whatever*
>>>> *> End With ' The shape*
>>>> *> Next ' Shape*
>>>> *>Next ' Slide*
>>>> *>*
>>>> *>End Sub*
>>>> *>*
>>>> *>-----*
>>>> *>Steve Rindsberg, PPT MVP*
>>>> *>PPT FAQ: www.pptfaq.com*
>>>> *>PPTools: www.pptools.com*
>>>> *>=====*
>>>> *>*
>>>> *>*
>>>> *>*
>>>> *>-----*
>>>> *>Steve Rindsberg, PPT MVP*

> >> >PPT FAQ: www.pptfaq.com
> >> >PPTools: www.pptools.com
> >> >=====

> >> >
> >>
> >
> >-----

> >Steve Rindsberg, PPT MVP
> >PPT FAQ: www.pptfaq.com
> >PPTools: www.pptools.com
> >=====

> >
>

Steve Rindsberg, PPT MVP
PPT FAQ: www.pptfaq.com
PPTools: www.pptools.com
