

Re: String too long...

Source:

<http://www.tech-archive.net/Archive/Word/microsoft.public.word.vba.general/2007-01/msg00828.html>

- *From:* "mike" <m@xxxxx>
 - *Date:* Tue, 30 Jan 2007 11:23:39 -0500
-

While I don't think Jezebel would agree. If you now go back to my very first posting, the question remains...

"Greg Maxey" <gmaxey@xxxxxxxx> wrote in message
<news:1170169167.182826.23810@xx>

My understanding is you had a problem finding a tag and replacing it with a long sting (>255 characters).

I think we agree that this will solve that problem without using the users clipboard:

```
Sub Test1()  
Dim oRng As Word.Range  
Set oRng = ActiveDocument.Range  
With oRng.Find  
.Text = "<Additional_Info>"  
While .Execute  
oRng.Text = String(256, "1")  
oRng.Collapse wdCollapseEnd  
Wend  
End With  
End Sub
```

Other than theoretical debate about greater or lesser solutions, what answer or other information are you looking for in this discussion?

On Jan 30, 9:41 am, "mike" <m...@xxxxx> wrote:

Hi Tony, thanks for your interest,

- > Please post code that is failing and indicate where.
- > There is not a general 255-character limit; it is only in certain
- > circumstances. The KB article you refer to is saying

Re: String too long...

that there is a
> 255-character limit putting text in formfields and shows
one way to get
> round it. If you're not using Form Fields to start with,
the article does
> not apply to you.
> What exactly are you finding does not work? Can you post
actual code?

There is a reply with two separate posts to your message with sample code (formatted similar to a Microsoft KB) I am not sure if you don't see them for some reason. If your News Reader goes directly to microsoft.public.word.vba.general you should.

> First of all, when you said in an earlier post that it's not
the ticket what
> is your problem with RangeObject.Select? Are you saying it does
not select
> the Range?

Yes, see my sample code that was sent replying to you. In the second posting, the one that implements Range.Find has two lines that have "ineffective" commented beside them.

> Secondly, I'll ask again, why are you so intent on using the
Selection? Greg
> has demonstrated how to use a Range. There are two (related) issues
with
> using the selection – it reflects what you are doing on screen in
full view
> of the user and this can be annoying because the screen can flicker
and also
> it is inefficient.

I didn't really wake up one day saying that "I need to use Selection."
We're experiencing Run-time error '5854' "String parameter is too long"
when placing long text into both Selection.Find.Replacement.Text and
Range.Find.Replacement.Text. So I began an investigation.

One stop, of many, was at a MS KB article recommending Selection, because
it offers TypeText. I found this solution "also" works splended after a
Selection.Find, see the first sample code posting that employs
Selection.Find.

While Greg has kindly provided a solution that implements the Clipboard
for the Range.Find.Replacement, it does require the overhead of copying
the long string to the Clipboard. Additionally it has the associated
vulnerabilities and impositions of using the user's Clipboard.

When compared to a Microsoft workaround that employs Selection.TypeText, a
routine that basically takes the actual buffer as input, and was proven

Re: String too long...

to work with Selection.Find, the Microsoft workaround offers a definite architectural advantage.

And finally with this application's implementation, while the document is being generated, it is not yet ".Visible" to the user.

So that brought me to the question...

I am in another section of code and am not been able to obtain the benefit of Selection.TypeText the code happens to be working with the Range object. How might one go about implementing this if working with a Range.Find & Execute?– Hide quoted text -- Show quoted text –