

WORD HACKS™

*Tips & Tools for
Taming Your Text*



O'REILLY®

Andrew Savikas

HACK
100

Google Without Leaving Word

This hack shows you how to search Google from within Word and display the results on the Task Pane.

You probably use Google a few dozen times a day. If you work in Word, this means switching over to your web browser, checking the results, and then going back to Word. Once you get Google working from a macro [Hack #99], this hack will show how to display the search results in the New Document Task Pane [Hack #9].

This hack uses a configuration file [Hack #67] to store data and some code that uses VBScript regular expressions [Hack #82]. Before you dive in, check out those hacks.

The Code

Open the *GoogleTools.dot* template you created when you installed the Web Services Toolkit, as discussed in “Use Google from Your Macros” [Hack #99]. Select Tools → Macro → Macros and insert the following code, which consists of a procedure named `GoogleToTaskPane` and a supporting function named `StripHTML`.

```

Sub GoogleToTaskPane()
    Dim vSearchResults As Variant
    Dim v As Variant
    Dim sResults As String
    Dim sEntryName As String
    Dim sEntryURL As String
    Dim sLogFile As String
    Dim sSearchDisplayTitle As String
    Dim sSearchURL As String
    Dim i As Integer

    ' Google API variables
    Dim sGoogleAPIKey As String
    Dim sSearchQuery As String
    Dim lStart As Long
    Dim lMaxResults As Long
    Dim bFilter As Boolean
    Dim sRestrict As String
    Dim bSafeSearch As Boolean
    Dim sLanguageRestrict As String
    Dim sInputEncoding As String
    Dim sOutputEncoding As String
    Dim google_search As New clsWS_GoogleSearchService

    ' Initialize variables
    sLogFile = "C:\google_taskpane.ini"
    sGoogleAPIKey = "your_key_here"

```

```

lStart = 1
lMaxResults = 10
bFilter = True
sRestrict = ""
bSafeSearch = False
sLanguageRestrict = ""
sInputEncoding = "UTF-8"
sOutputEncoding = "UTF-8"

' Hide the Task Pane
Application.CommandBars("Task Pane").Visible = False

' Remove existing items from New Document Task Pane
For i = 0 To 9
    sEntryURL = System.PrivateProfileString( _
        FileName:=sLogFile, _
        Section:="GoogleTaskPane", _
        Key:="URLName" & CStr(i))
    sEntryName = System.PrivateProfileString( _
        FileName:=sLogFile, _
        Section:="GoogleTaskPane", _
        Key:="EntryName" & CStr(i))
    If Len(sEntryURL) > 0 Then
        Application.NewDocument.Remove _
            FileName:=sEntryURL, _
            Section:=msoBottomSection, _
            DisplayName:=sEntryName, _
            Action:=msoOpenFile
    End If
Next i

' Get new search query
sSearchQuery = InputBox("Enter a Google query:")
If Len(sSearchQuery) = 0 Then Exit Sub

' Get search results
vSearchResults = google_search.wsm_doGoogleSearch( _
    str_key:=sGoogleAPIKey, _
    str_q:=sSearchQuery, _
    lng_start:=lStart, _
    lng_maxResults:=lMaxResults, _
    bln_filter:=bFilter, _
    str_restrict:=sRestrict, _
    bln_safeSearch:=bSafeSearch, _
    str_lr:=sLanguageRestrict, _
    str_ie:=sInputEncoding, _
    str_oe:=sOutputEncoding).resultElements

' Check for no results
On Error Resume Next
v = UBound(vSearchResults)
If Err.Number = 9 Then
    MsgBox "No results found"

```

```

Exit Sub
ElseIf Err.Number <> 0 Then
    MsgBox "An error has occurred: " & _
        Err.Number & vbCrLf & _
        Err.Description
Exit Sub
End If

' Add each result to the Task Pane
' and to the log file
i = 0
For Each v In vSearchResults
    sSearchURL = v.URL
    sSearchDisplayTitle = StripHTML(v.title)
    Application.NewDocument.Add _
        FileName:=sSearchURL, _
        Section:=msoBottomSection, _
        DisplayName:=sSearchDisplayTitle, _
        Action:=msoOpenFile

    System.PrivateProfileString( _
        FileName:=sLogFile, _
        Section:="GoogleTaskPane", _
        Key:="URLName" & CStr(i)) = sSearchURL
    System.PrivateProfileString( _
        FileName:=sLogFile, _
        Section:="GoogleTaskPane", _
        Key:="EntryName" & CStr(i)) = sSearchDisplayTitle
    i = i + 1
Next v

' Show the New Document Task Pane
CommandBars("Menu Bar").Controls("File").Controls("New...").Execute

End Sub

Function StripHTML(str As String) As String
Dim re As Object
Dim k As Long
On Error Resume Next
Set re = GetObject(Class:="VBScript.RegExp")
If Err.Number = 429 Then
    Set re = CreateObject(Class:="VBScript.RegExp")
    Err.Clear
ElseIf Err.Number <> 0 Then
    MsgBox Err.Number & vbCrLf & Err.Description
End If

' Check for common character entities by ASCII value
For k = 33 To 255
    re.Pattern = "&#" & k & ";"
    str = re.Replace(str, Chr$(k))
Next k

```

Google Without Leaving Word

```
' Remove common HTML tags
re.Pattern = "<[^>]+?>|&[^;]+?;"
re.Global = True
str = re.Replace(str, vbNullString)
StripHTML = str
End Function
```



Make sure you replace the value *your_key_here* with your Google developer's key.

This hack uses two parts of the Google search results: the URLs and titles. Google formats the search result title as HTML, but you can put only plain text in the Task Pane. The StripHTML function uses a few simple VBScript regular expressions to strip out common HTML tags (such as) and replace character entities (such as @) with their ASCII character equivalents (“Use Character Codes to Find or Insert Special Characters” [Hack #30]).



The StripHTML function uses *late binding*, as discussed in “Automate Word from Other Applications” [Hack #84].

It can be tricky to remove files from the Task Pane using VBA unless you know their exact names, as discussed in “Tweak the New Document Task Pane” [Hack #9]. This macro, however, stores the search results in a .ini file. Thus, the next time you do a search, you can easily remove the previous results. The macro uses a file named *C:\google_taskpane.ini*, which is defined in the GoogleToTaskPane procedure.

Running the Hack

After you insert the code, switch back to Word. Next, select Tools → Macro → Macros, choose GoogleToTaskPane, and click the Run button to display the dialog shown in Figure 10-21.

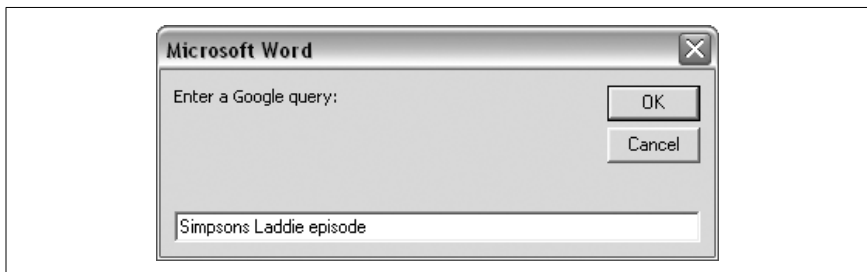


Figure 10-21. Entering a Google search that will display in the Task Pane

Enter your search terms and click the OK button. The New Document Task Pane will appear and display the search results, as shown in Figure 10-22. Hover your mouse over any of the entries to display the URL. Click a URL to open the site in your web browser.

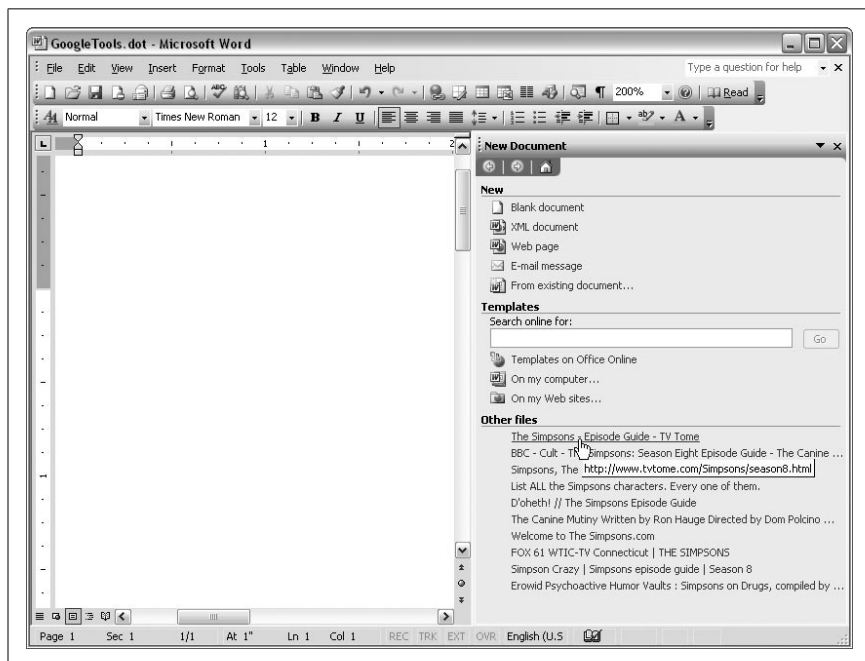


Figure 10-22. Google results displayed in the Task Pane

Every time you run a search, the macro removes the previous results from the Task Pane. If you want to remove the previous results without displaying new ones, click the Cancel button in the dialog box shown in Figure 10-21.

To make sure this handy macro loads automatically when Word starts [Hack #50], put *GoogleTools.dot* into your *STARTUP* folder.

Hacking the Hack

To take this hack one step further, you can modify it to use the currently selected text as the search text, rather than displaying an input box for you to enter text in.

The following macro, named *GoogleSelectionToTaskPane*, does a Google search of the currently selected text and displays the results in the Task Pane. The modified code is shown in bold.

```

Sub GoogleSelectionToTaskPane()
Dim vSearchResults As Variant
Dim v As Variant
Dim sResults As String
Dim sEntryName As String
Dim sEntryURL As String
Dim sLogFile As String
Dim sSearchDisplayTitle As String
Dim sSearchURL As String
Dim i As Integer

' Google API variables
Dim sGoogleAPIKey As String
Dim sSearchQuery As String
Dim lStart As Long
Dim lMaxResults As Long
Dim bFilter As Boolean
Dim sRestrict As String
Dim bSafeSearch As Boolean
Dim sLanguageRestrict As String
Dim sInputEncoding As String
Dim sOutputEncoding As String
Dim google_search As New clsws_GoogleSearchService

' Initialize variables
sLogFile = "C:\google_taskpane.ini"
sGoogleAPIKey = your_key_here
lStart = 1
lMaxResults = 10
bFilter = True
sRestrict = ""
bSafeSearch = False
sLanguageRestrict = ""
sInputEncoding = "UTF-8"
sOutputEncoding = "UTF-8"

' Hide the Task Pane
Application.CommandBars("Task Pane").Visible = False

' Remove existing items from New Document Task Pane
For i = 0 To 9
    sEntryURL = System.PrivateProfileString( _
        FileName:=sLogFile, _
        Section:="GoogleTaskPane", _
        Key:="URLName" & CStr(i))
    sEntryName = System.PrivateProfileString( _
        FileName:=sLogFile, _
        Section:="GoogleTaskPane", _
        Key:="EntryName" & CStr(i))
    If Len(sEntryURL) > 0 Then
        Application.NewDocument.Remove _
            FileName:=sEntryURL, _
            Section:=msoBottomSection, _

```

```

        DisplayName:=sEntryName, _
        Action:=msoOpenFile
    End If
Next i

' Move ends of selection to exclude spaces
' and paragraph marks
Selection.MoveStartWhile cset:=Chr (32) & Chr (19), _
    Count:=Selection.Characters.Count
Selection.MoveEndWhile cset:=Chr (32) & Chr (19), _
    Count:=-Selection.Characters.Count

' Get selection text for search
sSearchQuery = Selection.Text
If Len(sSearchQuery) = 0 Then Exit Sub

' Get search results
vSearchResults = google_search.wsm_doGoogleSearch( _
    str_key:=sGoogleAPIKey, _
    str_q:=sSearchQuery, _
    lng_start:=lStart, _
    lng_maxResults:=lMaxResults, _
    bln_filter:=bFilter, _
    str_restrict:=sRestrict, _
    bln_safeSearch:=bSafeSearch, _
    str_lr:=sLanguageRestrict, _
    str_ie:=sInputEncoding, _
    str_oe:=sOutputEncoding).resultElements

' Check for no results
On Error Resume Next
v = UBound(vSearchResults)
If Err.Number = 9 Then
    MsgBox "No results found"
    Exit Sub
ElseIf Err.Number <> 0 Then
    MsgBox "An error has occurred: " & _
        Err.Number & vbCrLf & _
        Err.Description
    Exit Sub
End If

' Add each result to the Task Pane
' and to the log file
i = 0
For Each v In vSearchResults
    sSearchURL = v.URL
    sSearchDisplayTitle = StripHTML(v.title)
    Application.NewDocument.Add _
        FileName:=sSearchURL, _
        Section:=msoBottomSection, _
        DisplayName:=sSearchDisplayTitle, _
        Action:=msoOpenFile

```

```
System.PrivateProfileString( _  
    FileName:=sLogFile, _  
    Section:="GoogleTaskPane", _  
    Key:="URLName" & CStr(i)) = sSearchURL  
System.PrivateProfileString( _  
    FileName:=sLogFile, _  
    Section:="GoogleTaskPane", _  
    Key:="EntryName" & CStr(i)) = _  
    sSearchDisplayTitle  
    i = i + 1  
Next v  
  
' Show the New Document Task Pane  
CommandBars("Menu Bar").Controls("File").Controls("New...").Execute  
  
End Sub
```

To help ensure a good Google search, the following two lines collapse two ends of the selection if they contain spaces or a paragraph mark:

```
Selection.MoveStartWhile cset:=Chr (32) & Chr (19), _  
    Count:=Selection.Characters.Count  
Selection.MoveEndWhile cset:=Chr (32) & Chr (19), _  
    Count:=-Selection.Characters.Count
```

Add the `GoogleSelectionToTaskPane` macro to your `Text` shortcut menu [Hack #3], and Google results will be just a right-click away.