

VB.net

July 13, 2007 / Gavin Jackson

programming

vb

windows

Had a bit of fun (?) at work writing a VB console application that simply opens up a word document and checks all headers and footers in the document for a specified list of keywords.

The only real gotcha is that you need to install the Primary Interop Assemblies for your version of office (and add them to your project references). Microsoft have a very strong com API for scripting their applications (Word, Excel, Powerpoint, Outlook etc).

Not much has changed since I last used it several years ago (VB.net) - I did not look into advancements in C#.

```

Module Module1
    'Print debug messages
    Const debug = False
    'If no parameters are specified, scan this document
    Const default_file = "c:\mydoc.doc"
    'The keywords we want to match on
    Const keywords = "c:\keywords.txt"
    'The logfile we want to write our results to
    Const logfile = "c:\headerfooter.log"
    Dim alKeywords As New ArrayList()

    Sub Main()
        loadKeywords()
        If Command().Length = 0 Then
            Console.WriteLine("No Params Passed")
            scanDocument("c:\mydoc.doc")
        Else
            Console.WriteLine(Command())
            scanDocument(Command())
        End If
        Console.ReadLine()
    End Sub

    Sub loadKeywords()
        'Read keywords into global string arraylist

        Dim fs As New FileStream("c:\keywords.txt", FileMode.Open, FileAccess.Read)
        'declaring a FileStream to open the file named file.doc with access mode of reading
        Dim d As New StreamReader(fs)
        'creating a new StreamReader and passing the filestream object fs as argument
        d.BaseStream.Seek(0, SeekOrigin.Begin)
        'Seek method is used to move the cursor to different positions in a file, in this code, to
        'the beginning
        While d.Peek() > -1
            'peek method of StreamReader object tells how much more data is left in the file
            Dim strKeyword As String
            strKeyword = d.ReadLine()

            If debug Then
                Console.WriteLine(strKeyword)
            End If

            alKeywords.Add(strKeyword)
            'displaying text from doc file in the RichTextBox
        End While
        d.Close()
    End Sub

    Sub scanDocument(ByVal sFile As String)
        Dim oApp As Word.Application
        Dim oDoc As Word.Document
        Dim oSec As Word.Section
        Dim oPageStart As Word.Range
        Dim iPage As Integer, iTotPages As Integer, iSection As Integer
    
```

```

Dim sHeader As String, sFooter As String

Const wdStatisticPages = 2
Const wdGotoPage = 1
Const wdGotoAbsolute = 1
Const wdHeaderFooterFirstPage = 2
Const wdHeaderFooterPrimary = 1

'Open the document
oApp = New Word.Application
oDoc = oApp.Documents.Open(sFile)
iTotalPages = oDoc.ComputeStatistics(wdStatisticPages)

'Retrieve the headers and footers on each page
With oDoc

iSection = 0

For iPage = 1 To iTotalPages

'Go to the page represented by the page number iPage and
'retrieve its section
oPageStart = oDoc.GoTo(What:=wdGotoPage, _
Which:=wdGotoAbsolute, Count:=iPage)
oSec = oPageStart.Sections(1)

'If this is a different section than the one in the previous
'iteration and it has a first page header/.footer, then
'retrieve the first page header/footer for this section.
'Otherwise, retrieve the primary header/footer for this section
If (iSection < oSec.Index) And _
(oSec.PageSetup.DifferentFirstPageHeaderFooter) Then
sHeader = oSec.Headers(wdHeaderFooterFirstPage).Range.Text
sFooter = oSec.Footers(wdHeaderFooterFirstPage).Range.Text
Else
sHeader = oSec.Headers(wdHeaderFooterPrimary).Range.Text
sFooter = oSec.Footers(wdHeaderFooterPrimary).Range.Text
End If

iSection = oSec.Index
If debug Then
Console.WriteLine("Page " & iPage & ", Section " & iSection & _
":" & vbCrLf)
Console.WriteLine(" Header: " & sHeader)
Console.WriteLine(" Footer: " & sFooter)
End If

checkText(sHeader, sHeader, sFooter, iPage, iSection)
checkText(sFooter, sHeader, sFooter, iPage, iSection)
Next

End With

```

```
'Make Word visible to compare the document with the results in the
'debug window
'oApp.Visible = True
oDoc.Close()
oApp.Quit()
End Sub
Sub checkText(ByVal sText As String, ByVal sHeader As String, ByVal sFooter As String, ByVal iPage
As Integer, ByVal iSection As Integer)

Dim i As Integer

For i = 0 To (alKeywords.Count() - 1)
If sText.ToUpper.Contains(alKeywords.Item(i).ToString().ToUpper()) Then
log("Match in document " & Command() & " Text: " & sText)
log(" Match Keyword: " & alKeywords.Item(i).ToString() & " Page: " & iPage & " Section: " &
iSection)
End If
Next
End Sub

Sub log(ByVal sTest As String)
Console.WriteLine(sTest)
End Sub
End Module
```

Downloaded from <https://www.gavinj.net/post/vbnet>

Generated July 9, 2026. Copyright Gavin Jackson. All rights reserved.