

Inhalt

1	Aufgabe	1
2	VBA-Prozeduren	2
3	Quellcode der VBA-Prozeduren	2

Listings

Listing 1: Treiber	2
Listing 2: Subprozedur	4
Listing 3: Funktionsprozedur	5

Tabellen

Tabelle 1: Typ, Name und Aufgabe der jeweiligen VBA-Prozedur	2
--	---

1 Aufgabe

In diesem Beitrag wird beschrieben, wie Leerstellen bzw. Leerräume (engl. 'white spaces') am Anfang und/oder am Ende jeden Absatzes im aktuellen Dokument mit geeigneten VBA-Prozeduren entfernt werden können.

2 VBA-Prozeduren

Die oben genannte Aufgabe lässt sich mit folgenden drei Prozeduren durchführen (s. Tabelle 1):

Prozedur		
Typ	Name	Aufgabe
Subprozedur	Driver	Hauptprozedur
Subprozedur	RemoveLeadingAndTrailingSpaces	Unerwünschte Leerstellen am Anfang und/oder Ende jeden Absatzes des aktuellen Dokuments entfernen
Funktionsprozedur	IsWhiteSpace	Boolesche Rückgabe, ob das an die Funktion übergebene Zeichen ein Leerzeichen bzw. einen Leerraum darstellt.

Tabelle 1: Typ, Name und Aufgabe der jeweiligen VBA-Prozedur

3 Quellcode der VBA-Prozeduren

Der Quellcode der drei oben genannten Sub- und Funktionsprozeduren wird im Folgenden dokumentiert:

```
1 Sub Driver()  
2     Call RemoveLeadingAndTrailingSpaces(ActiveDocument)  
3 End Sub
```

Listing 1: Treiber

```

1 Sub RemoveLeadingAndTrailingSpaces(ByVal docTgt As Document, _
2   Optional bolNonStop As Boolean = False)
3   ' Argumente: docTgt: aktuelles Word-Dokument
4   '           bolNonStop: Schalter f. 'ohne Unterbrechnung' (=> Ja/Nein)
5   ' Benötigt: Funktion "IsWhiteSpace"
6   Const conTitle As String = "Remove White Spaces"
7   Dim objPara As Paragraph ' Absatz im aktuellen Dokument
8   Dim lngAsk As Long ' Antwort auf Abfrage
9   Dim lngSpaceCnt As Long ' Zähler f. LZ (engl. 'white spaces')
10  Dim lngTotalDel As Long ' Insgesamt entfernte Leerzeichen
11  Dim rngChar As Range ' Suchbereich
12  Dim rngSpaceDel As Range ' Löchbereich
13  Dim lngParaCnt As Long ' Anzahl Absätze im akt. Dokument
14  On Error GoTo Err_Point ' Fehlerbehandlung
15  docTgt.Activate
16  With Selection
17    .HomeKey unit:=wdStory, Extend:=wdMove
18    Set objPara = .Range.Paragraphs(1) ' 1. Absatz im akt. Dokument
19  End With
20  Do
21    objPara.Range.Select
22    ' Absätze in Tabellen ignorieren
23    If Not Selection.Information(wdWithInTable) Then
24      ' LZ am Anfang eines Absatzes entfernen, falls vorhanden.
25      lngSpaceCnt = 0
26      Set rngChar = objPara.Range.Characters.First
27      If rngChar.Text <> vbCr Then
28        Do While IsWhiteSpace(rngChar.Text)
29          lngSpaceCnt = lngSpaceCnt + 1
30          Set rngChar = rngChar.Next
31          If rngChar Is Nothing Or rngChar.Text = vbCr Then
32            Exit Do
33          End If
34        Loop
35      End If
36      If lngSpaceCnt > 0 Then
37        ' Leerzeichen am Absatzanfang markieren
38        Set rngSpaceDel = ActiveDocument.Range( _
39          Start:=objPara.Range.Start, _
40          End:=objPara.Range.Start + lngSpaceCnt)
41        rngSpaceDel.Select
42        rngSpaceDel.HighlightColorIndex = wdYellow
43        If bolNonStop Then
44          lngAsk = vbYes
45        Else
46          lngAsk = MsgBox("Diese" & IIf(lngSpaceCnt < 2, "s ", " ") & _
47            CStr(lngSpaceCnt) & " Leerzeichen entfernen?", _
48            vbYesNoCancel + vbQuestion, conTitle)
49        End If
50        If lngAsk = vbYes Then
51          rngSpaceDel.Delete
52          lngTotalDel = lngTotalDel + lngSpaceCnt
53        ElseIf lngAsk = vbCancel Then
54          GoTo Err_Point
55        End If
56      End If
57      ' Leerzeichen am Ende eines Absatzes entfernen, falls vorhanden.
58      If objPara.Range.Characters.Count > 1 Then
59        ' Gehe vor die Absatzmarke
60        Set rngChar = objPara.Range.Characters.Last.Previous
61      Else
62        Set rngChar = Nothing
63      End If
64      lngSpaceCnt = 0
65      If Not rngChar Is Nothing Then
66        Do While IsWhiteSpace(rngChar.Text)
67          lngSpaceCnt = lngSpaceCnt + 1

```

```

68         Set rngChar = rngChar.Previous
69     Loop
70 End If
71 If lngSpaceCnt > 0 Then
72     ' Leerzeichen am Absatzende markieren
73     Set rngSpaceDel = ActiveDocument.Range( _
74         Start:=objPara.Range.End - lngSpaceCnt - 1, _
75         End:=objPara.Range.End - 1)
76     rngSpaceDel.Select
77     rngSpaceDel.HighlightColorIndex = wdYellow
78     If bolNonStop Then
79         lngAsk = vbYes
80     Else
81         lngAsk = MsgBox("Diese" & IIf(lngSpaceCnt < 2, "s ", " ") & _
82             CStr(lngSpaceCnt) & " Leerzeichen entfernen?", _
83             vbYesNoCancel + vbQuestion, conTitle)
84     End If
85     If lngAsk = vbYes Then
86         rngSpaceDel.Delete
87         ' Gehe vor die Absatzmarke
88         Set rngSpaceDel = objPara.Range.Characters.Last.Previous
89         rngSpaceDel.Select
90         If rngSpaceDel.Text = Space(1) Then
91             rngSpaceDel.Delete
92             rngSpaceDel.Delete ' 2-mal wg. Bug in Word
93         End If
94         lngTotalDel = lngTotalDel + lngSpaceCnt
95     ElseIf lngAsk = vbCancel Then
96         GoTo Exit_Point
97     End If
98 End If
99 End If
100 Set objPara = objPara.Next ' Nächster Absatz
101 Loop Until objPara Is Nothing ' Kein weiterer Absatz
102 Selection.EndKey unit:=wdStory, Extend:=wdMove
103 MsgBox Str(lngTotalDel) & _
104     " Leerstellen (engl. 'white spaces') am Anfang/Ende eines Absatzes " & _
105     IIf(bolNonStop, "automatisch", "interaktiv") & " entfernt.", _
106     vbInformation, conTitle
107 Exit_Point:
108 On Error Resume Next
109 Set objPara = Nothing
110 Exit Sub
111 Err_Point:
112 MsgBox "Laufzeitfehler: " & Err.Number & ", " & vbCrLf & _
113     Err.Description & ", " & Err.Source, vbCritical, conTitle
114 Resume Exit_Point
115 End Sub

```

Listing 2: Subprozedur

```

1  Function IsWhiteSpace(ByVal strChar As String) As Boolean
2  ' Aufgabe: Gibt 'true' zurück, wenn das übergebene Zeichen (strChar) ein Leerzeichen
3  ' ist, sonst 'false'.
4  ' Wird aufgerufen von Subprozedur: RemoveLeadingAndTrailingSpaces.
5  ' Quelle: o. V., Whitespace character
6  ' URL: https://en.wikipedia.org/wiki/Whitespace_character#Unicode
7  ' -----
8  ' Die 24 Hex-Zahlen in der unten stehenden Select-Case Anweisung bedeuten:
9  ' Horizontal Tab, HT, vbKeyTab
10 ' LineFeed, LF
11 ' Vertical Tab, VT
12 ' Form Feed, FF, vbKeyClear
13 ' Carriage Return, CR, vbKeyReturn
14 ' Space, vbKeySpace
15 ' Non-Breaking Space, Hard Space
16 ' OGHAM Space Mark
17 ' En Quad
18 ' Em Quard (Mutton Quad)
19 ' En Space (Nut)
20 ' Em Space (Mutton)
21 ' Tree-Per-Em-Space (Thick Space)
22 ' Four-Per-Em-Space (Mid Space)
23 ' Six-Per-Em-Space
24 ' Figure Space
25 ' Punctuation Space
26 ' Thin Space
27 ' Hair Space
28 ' Zero-Width Space
29 ' Narrow No-Break Space
30 ' Medium Mathematical Space
31 ' Ideographic Space
32 ' Zero-Width Non-Breaking Space
33 ' -----
34  Select Case AscW(strChar)
35      Case &H9, &HA, &HB, &HC, &HD, &H20, &HA0, &H1680, _
36           &H2000, &H2001, &H2002, &H2003, &H2004, &H2005, &H2006, &H2007, _
37           &H2008, &H2009, &H200A, &H200B, &H202F, &H205F, &H3000, &HFEFF
38          IsWhiteSpace = True
39      Case Else
40          IsWhiteSpace = False
41  End Select
42 End Function

```

Listing 3: Funktionsprozedur