

„Kdyby peklo nebylo, tak by si ho musel Microsoft vymyslet.“ Roky říkám uživatelům, že mezery v datumu jsou sice správně typograficky, nicméně Excel je nesnáší a varoval jsem před změnami v nastavení systému Windows. No a nenašel se v Redmondu nějaký dobrodruh, který ve Windows 8/8.1 mezery narval do krátkého formátu datumu jako výchozí? A tak se ozval třeba včera uživatel, že mu nefunguje konverze datumu ve formátu řetězce na skutečné datum typu Date.



Windows 8 – krátký formát datumu

Pokud Windows 8 nemáte, můžete si situaci nasimulovat i ve Windows 7 a starších.



Excel – krátký formát datumu s mezerami

Nejjednodušší je samozřejmě v nastavení systému mezery odmazat. Co když si to ale nemůžete dovolit a potřebujete univerzální makro? Po několika hodinách se mi podařilo sepsat vlastní funkci, která – jak doufám – pokrývá alespoň běžné stavy. Vůbec si ale nedělám iluze o její rychlosti na tisících datumech. Pracuje zkrátka s textovými řetězci a to chvíli trvá. Třeba se vám podaří najít elegantnější řešení.

```

1 Sub DatumPrevod()
2
3 Dim strDatum As String
4
5 Dim dtDatum As Date
6
7 strDatum = "19.3.2014"
8
9 'když selhává
10 'dtDatum = CDate(strDatum)
11 'a podobně končí chybou i další
12 'dtDatum = Format(strDatum, "Short Date")
13 'dtDatum = DateValue(strDatum)
14 'dtDatum = Evaluate("=DATEVALUE(""" & strDatum & """)")
15
16 'pak je potřeba postupovat jinak...
17
18 dtDatum = CDate(epfDATUMTEXT(strDatum, "d.m.yyyy"))
19
20 End Sub
21
22 Function epfDATUMTEXT(strDatumZdroj As String, strDatumZdrojMaska As String, _
23 Optional strDatumCilMaska As String) As String
24
25 Dim lngLCID As Long
26 Dim lngRet As Long
27 Dim intDen As Integer, intMesic As Integer, intRok As Integer
28 Dim intPocetD As Integer, intPocetM As Integer, intPocetY As Integer
29
30 Dim strD As String, strM As String, strY As String, strZnak As String, _
31 strTemp As String
32 Dim strDatumCil As String, strDatumFormatKratky As String
33
34 'jaký je formát pro datum?
35 'nedostačující informace
36 'Application.International(...)
37
38 'oddělovače ze systému, tj.
39 'Application.UseSystemSeparators = True
40 If Len(strDatumCilMaska) = 0 Then
41 'krátký formát datumu ze systému
42 'LCID ... identifikátor oblasti/jazyka
43 'čeština ... 1029
44 lngLCID = GetUserDefaultLCID()
45 strDatumFormatKratky = Space$(16)
46 lngRet = GetLocaleInfo(lngLCID, LOCALE_SSHORTDATE, _
47 strDatumFormatKratky, 16)
48 strDatumCilMaska = Left$(strDatumFormatKratky, lngRet - 1)
49 End If
50
51 'pro každý znak původní masky
52 For i = 1 To Len(strDatumZdrojMaska)
53 'znak masky
54 strZnak = UCase(Mid$(strDatumZdrojMaska, i, 1))
55 'odpovídající znak ve zdrojovém datumu
56 strTemp = Mid$(strDatumZdroj, i + j, 1)
57
58 If (InStr(1, "DMY", strZnak) = 0) And (IsNumeric(strTemp)) Then
59 strZnak = strX
60 j = j + 1
61 End If
62
63 'nabíjení čísel do řetězce patřičné skupiny
64 Select Case strZnak
65 Case "D"
66 strD = strD & strTemp
67 strX = "D"
68 Case "M"
69 strM = strM & strTemp
70 strX = "M"
71 Case "Y"
72 strY = strY & strTemp
73 strX = "Y"
74 End Select
75 Next i
76
77 'převod na číselné hodnoty
78 intDen = Val(strD)
79 intMesic = Val(strM)
80 intRok = Val(strY)
81
82 strDatumCilMaskaTemp = UCase(strDatumCilMaska)
83
84 'počty zástupných znaků v cílové masce
85 intPocetD = Len(strDatumCilMaskaTemp) - Len(Replace(strDatumCilMaskaTemp, _
86 "D", ""))
87 intPocetM = Len(strDatumCilMaskaTemp) - Len(Replace(strDatumCilMaskaTemp, _
88 "M", ""))
89 intPocetY = Len(strDatumCilMaskaTemp) - Len(Replace(strDatumCilMaskaTemp, _
90 "Y", ""))
91
92 'záměna zástupných znaků v cílové masce za skutečné hodnoty
93 strDatumCil = Replace(strDatumCilMaskaTemp, String(intPocetD, "D"), _
94 Format(intDen, String(intPocetD, "0")))
95 strDatumCil = Replace(strDatumCil, String(intPocetM, "M"), Format(intMesic, _
96 String(intPocetM, "0")))
97 strDatumCil = Replace(strDatumCil, String(intPocetY, "Y"), Right(intRok, _
98 intPocetY))
99
100 epfDATUMTEXT = strDatumCil
101
102 End Function

```

Alespoň několik poznámek:

- Funkce s pomocí API zjišťuje formát krátkého datumu v systému (API funkce jsou uvedeny pouze v sešitu, zde nejsou vypsány).
- Funkce nezvládá rozdělené sekvence stejného typu (například yy-yy jako rok 20-14).
- Funkce jakžtakž zvládá fakticky nesourodé datумы a jejich masky (běžné „d.m.yyyy“ pro 19.3.2014 jistě chápete vy i Excel, ale naučte to vlastní program, kde jednomu „d“ odpovídá jedna nebo dvě číslice...).
- Je zvláštní, že funkce listu DATUMHODN (stejně jako rozpoznávací funkce při ručním vstupu do buňky) zblajzne kde co a vrátí datum, zatímco její užití v rámci VBA a Evaluate neprojde (pod WorksheetFunction není dostupná, protože VBA obsahuje vlastní funkci DateValue).
- Formát buňky pro datum obsahuje systémové formáty a značí je hvězdičkou. A i když popisek v okně tvrdí, jak jsou provázané se systémem, já tvrdím, že si „melou svou“ bez ohledu na nastavení ve Windows.
- Funkci je možné zkrátit. V momentě, kdy naplníte proměnné intDen, intMesic a intRok, aplikujte VBA funkci DateSerial a obdržíte datum typu Date. Já to dotáhnul až do řetězcové podoby odpovídající krátkému formátu datumu.



Excel - formát buňky pro datum

Další ukázky užití výše uvedené funkce:

```
1  Sub Test_epfDATUMTEXT()  
2  
3      Dim strDatum1 As String  
4      Dim strDatum2 As String  
5  
6      'zdrojové datum, zdrojová maska datumu, cílová maska datumu  
7      strDatum1 = epfDATUMTEXT("19-03|2014", "dd-MM/yyyy", "yy/mm/dd")  
8  
9      'zdrojové datum, zdrojová maska datumu, cílová maska datumu  
10     'dle krátkého formátu datumu v systému Windows  
11     strDatum2 = epfDATUMTEXT("19-03|2014", "dd-MM/yyyy")  
12  
13 End Sub
```



Excel - vlastní funkce pro datum zadané textem

Sešit ke stažení:

[excel-datum-kratky-format.zip](#)