

Berete do ruky svou pár let starou šablonu nebo sešit klienta, jehož barevná koncepce a štábní kultura pokulhává a chystáte se ho přebarvit? Tušíte, že to bude práce na dlouhé minuty, ne-li hodiny? Zkusím vám práci trochu zpříjemnit a především zkrátit.

Ergonomie novodobých Excelů s Pásem karet nám moc nepomáhá. Zatímco dříve bylo možné patřičné panely pro barvy a ohraničení možné odtrhnout a mít je k dispozici jako plovoucí dialogy na jedno klepnutí myši, dnes je především přístup k nástrojům ohraničení otravný. Přitom ruční kreslení ohraničení patřilo k neprávem opomíjeným technikám.



### Panely pro práci s barvami a ohraničením v Excelu 2003 a 2010

S tímhle už dnes v Excelu prostě nepochodíte...

1	<b>Sub</b> Excel2003()
2	
3	'panely Barva výplně, Barva písma a Ohraničení
4	Application.CommandBars("Fill Color").Visible = <b>True</b>
5	Application.CommandBars("Font Color").Visible = <b>True</b>
6	Application.CommandBars("Borders").Visible = <b>True</b>
7	
8	<b>End Sub</b>

Můžete namítat, že Excel 2007 jako první přinesl rozšířenou paletou barev a lepší práci s motivy. Osobně to vidím spíš jako barevný povyk pro nic, který byl převzat z okrajově používaného Publisheru. Pro přebarvení nám nebude moc platná ani skupina Motivы na kartě Rozložení stránky, ani Styly na kartě Domů. Potřebujeme něco víc.

Je škoda, že uživatelé pořádně neznají možnosti dialogu Najít/Nahradit (CTRL+F, CTRL+H). Většina z nich nikdy neklepla myškou na tlačítko Možnosti...



### Najít a nahradit – Formát

Pozn. Dobrým zvykem je před uzavřením dialogu vymazat formátování (rozklepněte tlačítko Formát, volba Vymazat pole pro hledání formátu). Programové zpracování by mohlo vypadat nějak takto:

```

1  Sub ZmenaBarevnostiPozadi()
2      With Application
3          'najít barvu pozadí buňky (červená)
4          .FindFormat.Interior.ColorIndex = 3
5          'zaměnit za barvu (fialová)
6          .ReplaceFormat.Interior.ColorIndex = 13
7      End With
8      'vlastní záměna pro výběr buněk
9      Selection.Replace What:="", Replacement:="", LookAt:=xlPart, _
10         SearchOrder:=xlByRows, MatchCase:=False, SearchFormat:=True, _
11         ReplaceFormat:=True
12      'odstranění nastavení formátů pro vyhledávání
13      With Application
14          .FindFormat.Clear
15          .ReplaceFormat.Clear
16      End With
17  End Sub
18
19  Sub ZmenaBarevnostiOhraniceni()
20      With Application
21          'najít barvu ohraničení buňky (červená)
22          .FindFormat.Borders.ColorIndex = 3
23          'zaměnit za barvu (fialová)
24          .ReplaceFormat.Borders.ColorIndex = 13
25      End With
26      'vlastní záměna pro výběr buněk
27      Selection.Replace What:="", Replacement:="", LookAt:=xlPart, _
28         SearchOrder:=xlByRows, MatchCase:=False, SearchFormat:=True, _
29         ReplaceFormat:=True
30      'odstranění nastavení formátů pro vyhledávání
31      With Application
32          .FindFormat.Clear
33          .ReplaceFormat.Clear
34      End With
35  End Sub

```

Výpis VBA výše ukazuje práci s vlastností ColorIndex, která je vlastní základní barevné paletě z Excelu 2003. Na sešity novodobých verzí Excelu doporučuji aplikovat vlastnost Color.

Pozor na jednu věc. Řekněme, že existuje zelené svislé ohraničení mezi sousedícími buňkami C7 a D7. Excelu není jedno, jestli k němu fyzicky došlo obarvením pravé hrany buňky C7 nebo levé hrany buňky D7, i když pro oko je výsledek stejný. Evidentní je tento rozdíl v případě, že v daném místě dojde k zalomení stránky. Ohraničení se pak zobrazí u té buňky, na které bylo skutečně užito.

Pořád to není ale to pravé ořechové, že? Co když chceme nahradit všechny výskyty červené barvy u

všech typů ohraničení naráz? Jistě, máme tu VBA...

Následující proceduru jsem sestavil co nejuniverzálnější. Očekává, že uživatel vybere oblast pro záměnu barevnosti, následně referenční buňky s původními a novými barvami, a zeptá se ještě na typ objektu pro přebarvení (0 ... vše, 1 ... písmo, 2 ... pozadí, 3 ... ohraničení). Zde už pracuji s vlastností Color.

```

1 Sub ZmenaBarevnosti()
2
3 Dim strRev As String
4 Dim mgBunka As Range
5 Dim mgRefOblast1 As Range
6 Dim mgRefOblast2 As Range
7 Dim iCil As Byte
8 Dim iPocetBarev As Integer
9 Dim aPoleBarvy()
10
11 On Error Resume Next
12
13 'oblast pro přebarvení
14 Set mgOblast = Application.InputBox("Vyberte oblast pro přebarvení.", _
15 "Zpracovávaná oblast", Selection.Address(0, 0), , , , 8)
16 If Err <> 0 Then Exit Sub
17
18 'referenční buňka s původní (hedanou) barvou
19 Set mgRefOblast1 =
20 Application.InputBox("Vyberte referenční buňky s původními barvami.", _
21 "Referenční oblast 1", Selection.Address(0, 0), , , , 8)
22 If Err = 0 Then
23
24 iPocetBarev = mgRefOblast1.Cells.Count
25
26 ReDim aPoleBarvy(1 To iPocetBarev, 1 To iPocetBarev)
27
28 For i = 1 To iPocetBarev
29 'původní barvy
30 aPoleBarvy(i, 1) = mgRefOblast1.Cells(i).Interior.Color
31 Next i
32
33 Else
34 Exit Sub
35 End If
36
37 'referenční buňka s nově aplikovanou barvou
38 Set mgRefOblast2 =
39 Application.InputBox("Vyberte referenční buňky s novými barvami.", _
40 "Referenční oblast 2", mgRefOblast1.Offset(0, 1).Address(0, 0), , , , 8)
41 If Err = 0 Then
42
43 For i = 1 To iPocetBarev
44 'nové barvy
45 aPoleBarvy(i, 2) = mgRefOblast2.Cells(i).Interior.Color
46 Next i
47
48 Else
49 Exit Sub
50 End If
51
52 'cil přebarvení (0 ... vše, 1 ... písmo, 2 ... pozadí, 3 ... ohraničení)
53 strRev = Application.InputBox("Cil přebarvení",
54 Prompt:="0 ... vše, 1 ... písmo, 2 ... pozadí, 3 ... ohraničení", _
55 Default:="0", Type:=2)
56
57 If strRev = "False" Then
58 Exit Sub
59 Else
60 iCil = Val(strRev)
61 End If
62
63 Application.ScreenUpdating = False
64
65 For i = 1 To iPocetBarev
66 For Each mgBunka In mgOblast.Cells
67 With mgBunka
68 If iCil = 1 Or iCil = 0 Then
69 If mgBunka.Font.Color = aPoleBarvy(i, 1) Then
70 'barva písma
71 mgBunka.Font.Color = aPoleBarvy(i, 2)
72 End If
73 End If
74
75 If iCil = 2 Or iCil = 0 Then
76 If mgBunka.Interior.Color = aPoleBarvy(i, 1) Then
77 'barva pozadí
78 mgBunka.Interior.Color = aPoleBarvy(i, 2)
79 End If
80 End If
81
82 'Borders.Color vrací Null,
83 'pokud všechna ohraničení nemají stejnou barvu
84
85 If iCil = 3 Or iCil = 0 Then
86 'barva horního ohraničení
87 With Borders(xlEdgeTop)
88 If .Color = aPoleBarvy(i, 1) Then
89 .Color = aPoleBarvy(i, 2)
90 End If
91 End With
92
93 'barva pravého ohraničení
94 With Borders(xlEdgeRight)
95 If .Color = aPoleBarvy(i, 1) Then
96 .Color = aPoleBarvy(i, 2)
97 End If
98 End With
99
100 'barva dolního ohraničení
101 With Borders(xlEdgeBottom)
102 If .Color = aPoleBarvy(i, 1) Then
103 .Color = aPoleBarvy(i, 2)
104 End If
105 End With
106
107 'barva levého ohraničení
108 With Borders(xlEdgeLeft)
109 If .Color = aPoleBarvy(i, 1) Then
110 .Color = aPoleBarvy(i, 2)
111 End If
112 End With
113
114 'barva ohraničení hlavní diagonály
115 With Borders(xlDiagonalDown)
116 If .Color = aPoleBarvy(i, 1) Then
117 .Color = aPoleBarvy(i, 2)
118 End If
119 End With
120
121 'barva ohraničení vedlejší diagonály
122 With Borders(xlDiagonalUp)
123 If .Color = aPoleBarvy(i, 1) Then
124 .Color = aPoleBarvy(i, 2)
125 End If
126 End With
127
128 End If
129 End With
130
131 End If
132
133 Next mgBunka
134
135 Next i
136
137 Application.ScreenUpdating = True
138
139 End Sub

```



Změna barevnosti s pomocí VBA

Příloha:

[prebarveni.zip](#)