

Activate a Select jsou sesterské metody, se kterými se setkáváme pod VBA. Jejich cílem je zjednodušeně řečeno vybrat objekt. My se budeme zabývat aktivací (výběrem) oblasti buněk a (ouška) listu v Excelu.

Na metodu Select narazíme už ve výstupním kódu ze Záznamníku maker (Excel 2010). Příklad ukazuje výběr Listu1, buňky B5, a následné obarvení jejího pozadí na žluto.

1	Sub Makro1()
2	'
3	' Makro1 Makro
4	'
5	Sheets("List1"). Select
6	Range("B5"). Select
7	With Selection.Interior
8	.Pattern = xlSolid
9	.PatternColorIndex = xlAutomatic
10	.Color = 65535
11	.TintAndShade = 0
12	.PatternTintAndShade = 0
13	End With
14	
15	End Sub

Nahrané makro tedy nejprve vybírá List1 (Select, klepli jsme na ouško listu), buňku B5 (Select, klepli jsme na buňku myší), a následně pracuje s výběrem (Selection, konstrukce With..End With). Jak si ukážeme, celá stavba makra je ale zbytečně komplikovaná a lehce nebezpečná. Navíc, v případě výběru listu bychom se měli bavit spíše o jeho aktivaci (metoda Activate).

Začínající programátor, který příklad odkouká ze Záznamníku maker, pak použije něco v tomto duchu:

1	Sub KlasickyPriklad()
2	
3	'aktivace třetího listu
4	Worksheets(3).Activate
5	
6	'výběr oblasti B2:D6 aktivního listu
7	Range("B2:D6"). Select
8	
9	'práce s pozadím (vybrané oblasti)
10	Selection.Interior.ColorIndex = 6
11	Selection.Interior.ColorIndex = xlNone
12	
13	End Sub

Nevýhody

Jestliže zapomenete na aktivaci požadovaného listu a u objektu Range neuvédete rodiče, dosadí si za něj Excel aktivní list.

Pokud nevyberete oblast buněk, může se stát, že pod objektem Selection bude třeba obrázek.

Neustálé přepínání se mezi listy a výběry buněk velmi výrazně zpomaluje vaši proceduru (dochází k překreslování obrazovky a musíme sáhnout po známé syntaxi `Application.ScreenUpdating = False`).

Pozn. Výběr nelze provádět na neaktivním listu.

1	Sub SelectNeaktivniList()
2	
3	'aktivace prvního listu
4	Worksheets(1).Activate
5	
6	'výběr nelze provádět na neaktivním listu
7	'chyba
8	Worksheets(3).Range("B2:D6"). Select
9	
10	End Sub

Rozdíl mezi Activate a Select se dobře demonstruje na oblasti buněk.

1	Sub RozdilActivateSelect()
2	
3	'aktivace prvního listu
4	Worksheets(1).Activate
5	
6	'výběr oblasti B2:D6 aktivního listu
7	Range("B2:D6"). Select
8	'lze, nepoužívá se
9	'Range("B2:D6").Activate
10	
11	'aktivace buňky B5 (se zachováním výběru)
12	Range("B5").Activate
13	
14	End Sub



Metoda Select aplikovaná na výběru a Activate na buňce tohoto výběru

Ručně lze změny aktivní buňky ve výběru docílit klávesou TAB nebo například klávesovou zkratkou CTRL + . (tečka), která prochází rohové buňky výběru.

A teď průšvih, na který mě upozornil klient v nefunkčním kódu. Představme si, že druhý list ze tří je skryt.



Skrytý list a chování metod Select a Activate

1	Sub BugPrikladA()
2	
3	'tři listy, druhý skrytý
4	
5	'aktivace prvního listu
6	Worksheets(1).Activate
7	Worksheets(1). Select
8	
9	'aktivace druhého viditelného, tedy třetího listu!
10	Worksheets(2).Activate
11	'chyba:
12	Worksheets(2). Select
13	
14	'aktivace třetího listu
15	Worksheets(3).Activate
16	Worksheets(3). Select
17	
18	End Sub

A vyložený bug:

1	Sub BugPrikladB()
2	
3	'tři listy, druhý skrytý
4	
5	'aktivace prvního listu
6	Worksheets(1).Activate
7	
8	'aktivace druhého viditelného, tedy třetího listu!
9	Worksheets(2).Activate
10	
11	'při krokování hodnota zapsána do třetího listu!
12	'při spuštění celé procedury zapsána hodnota do druhého listu!
13	Cells(1) = 10
14	
15	End Sub

Abyste měli vše pod kontrolou, zvykněte si používat objektové proměnné (pro přiřazení nutný příkaz Set).

1	Sub KlasickyPrikladLepe()
2	
3	Dim wshList As Worksheet
4	Dim rngOblast As Range
5	
6	'aktivace prvního listu
7	'zde pouze pro ukázkou!
8	Worksheets(1).Activate
9	
10	'přiřazení třetího listu do objektové proměnné
11	Set wshList = Worksheets(3)
12	
13	'přiřazení oblasti buněk do objektové proměnné
14	Set rngOblast = wshList.Range("B2:D6")
15	
16	'žádný Activate pro třetí list...
17	'žádný Select pro oblast buněk...
18	
19	'práce s pozadím (vybrané oblasti)
20	'konstrukce With..End With
21	With rngOblast
22	.Interior.ColorIndex = 6
23	.Interior.ColorIndex = xlNone
24	End With
25	
26	End Sub

Uvedenou techniku používejte i v rámci přenosu dat mezi dvěma listy a oblastmi (zpravidla zdroj a cíl). Na aktivní list se sice můžete odvolat jako na ActiveSheet, nicméně tento postup doporučuji jen v případě jistoty volání procedury z daného listu (makro přiřazené tlačítku) nebo u obecných maker. V případě obsluhy dvou různých sešitů definujte objektovou proměnnou i pro ně a nespolehejte se pokud možno na odkaz ActiveWorkbook (aktivní sešit). Naopak vhodný je odkaz na ThisWorkbook (sešit nebo doplněk, z něhož je makro voláno). K aktivní buňce (aktivního listu) můžete přistoupit přes ActiveCell. Objektovou proměnnou na úrovni aplikace se zde nebudeme zabývat.

Pozn. Pro objekt ActiveSheet nefunguje pod VBA automatický seznam členů (nabídka po zápisu tečky).

1	Sub AktivniObjekty()
2	
3	Dim wkbSesit As Workbook
4	Dim wshList As Worksheet
5	Dim rngBunka As Range
6	
7	'přiřazení aktivního sešitu do objektové proměnné
8	Set wkbSesit = ActiveWorkbook
9	'přiřazení sešitu, v němž leží tato procedura
10	'do objektové proměnné
11	'Set wkbSesit = ThisWorkbook
12	
13	'přiřazení aktivního listu do objektové proměnné
14	Set wshList = ActiveSheet
15	
16	'přiřazení aktivního buňky do objektové proměnné
17	Set rngBunka = ActiveCell
18	
19	End Sub

Ačkoliv to v proceduře uvedeno není, nezapomínejte na konci práce korektně objektové proměnné „resetovat“ (uvolňovat pro ně vyhrazenou paměť) příkazem Set objPromenna = Nothing.

Tipy pro odkazy na sešit a oblast buněk:

1	Sub TipyNaOdkazy()
2	
3	Dim wshList As Worksheet
4	Dim rngOblast As Range
5	
6	'není vhodné spoléhat se na pořadí
7	'nepřehledné
8	Set wshList = Worksheets(3)
9	
10	'lépe je použít název
11	Set wshList = Worksheets("List3")
12	
13	'nejlepší je odkazovat se na kódové jméno listu
14	'kdy není potřeba ani objektová proměnná
15	'viz VBA, okno Properties
16	'vlastnost Name pro vybraný modul listu
17	wshMujList.Range("B10") = "mrkev"
18	
19	'odkaz na pojmenovanou buňku
20	'viz Definované názvy na listu
21	wshMujList.Range("MojeBunka") = "celer"
22	
23	End Sub



Kódové jméno listu

V případě kódového jména listu se tak nemusíte bát, že vám někdo listu přejmenuje. Odkaz na pojmenovanou buňku zajistí platnost odkazu i v případě posunu buňky.

Zlovyk číslo jedna máte tedy za sebou a nyní k dalšímu - kopírování hodnot.

1	Sub PrikladKopirovaniHodnoty()
2	
3	'aktivace prvního listu
4	Worksheets(1).Activate
5	
6	'zkopírování buňky B5
7	Range("B5").Copy
8	
9	'aktivace třetího listu
10	Worksheets(3).Activate
11	
12	'vložení hodnoty do buňky B10
13	Range("B10").PasteSpecial (xlPasteValues)
14	
15	'zrušení režimu kopírování
16	Application.CutCopyMode = False
17	
18	End Sub

Pokud kopírujeme něco jiného, než čistě hodnotu, pak se práci s Copy/Paste nevyhneme. Nicméně pro přenos hodnot/vzorců (Value, Formula) se nabízí mnohem efektivnější způsob.

1	Sub PrikladKopirovaniHodnotyLepe()
2	
3	'přímé převzetí obsahu buňky
4	Worksheets(3).Range("B10").Value = Worksheets(1).Range("B5").Value
5	
6	End Sub

Jak prosté. I zde bychom měli lépe řídit zdroj a cíl přes objektovou proměnnou. Podotýkám, že takto je možné pracovat i se skrytým listem.

Pozn. Metoda Copy umožňuje přímé zadání cíle.

```

1  Sub PrikladKopirovaniBunky()
2
3      'překopírování buňky
4      Worksheets(1).Range("B5").Copy Worksheets(3).Range("B10")
5
6  End Sub

```

Nyní příklady na vytváření nového sešitu.

```

1  Sub NovySesit()
2
3      'vytvoření nového sešitu
4      'stane se aktivním
5      Workbooks.Add
6
7      'práce s novým sešitem...
8      ActiveWorkbook.Worksheets(1).Cells(1) = 10
9
10 End Sub
11
12 Sub NovySesitLepe()
13
14     Dim wkbNovySesit As Workbook
15
16     'vytvoření nového sešitu
17     'a jeho přiřazení do objektové proměnné
18     Set wkbNovySesit = Workbooks.Add
19
20     'práce s novým sešitem...
21     wkbNovySesit.Worksheets(1).Cells(1) = 10
22
23 End Sub

```

```

1  Sub NovySesitZKopieListu()
2
3      'vytvoření nového sešitu z kopie stávajícího listu
4      Worksheets(1).Copy
5
6      'práce s listem...
7      ActiveSheet.Cells(1) = 10
8
9  End Sub
10
11 Sub NovySesitZKopieListuLepe()
12
13     Dim wshSesit As Workbook
14     Dim wshList As Worksheet
15
16     'vytvoření nového sešitu z kopie stávajícího listu
17     'a jeho přiřazení do objektové proměnné
18     'nelze uskutečnit, metoda Copy nevrací objekt
19     'Set wshList = Worksheets(1).Copy
20
21     'vytvoření nového sešitu z kopie stávajícího listu
22     Worksheets(1).Copy
23
24     'přiřazení aktivního listu do objektové proměnné
25     Set wshList = ActiveSheet
26
27     'přiřazení sešitu do objektové proměnné (odkaz na rodiče)
28     Set wshSesit = wshList.Parent
29     'alternativně
30     'Set wshSesit = ActiveWorkbook
31
32     'práce s listem...
33     wshList.Cells(1) = 10
34
35 End Sub

```

Doufám tedy, že tímto článkem už jste si „vybrali“ tu správnou cestu a že daný režim zůstane „aktivován“. A nezapomínejte, že Záznamník maker je dobrý sluha i pro pokročilé programátory, ale špatný pán.