

Provázané TextBoxy na formuláři potřebujeme v případě, kdy například mají sloužit jako převodník mezi fyzikálními jednotkami (tlak, GPS souřadnice, ...). Pokud je vazba jednosměrná, stačí v události Change prvního prvku měnit obsah druhého. Jestliže ale nechcete uživatele omezovat v tom, který prvek si vybere jako vstupní, pak už to s kódem VBA nebude tak jednoduché. A co když se navzájem ovlivňují tři textová pole na formuláři?

Hned zkraje je potřeba si uvědomit, že Application.EnableEvents nezahrnuje události ovládacích prvků na formuláři. V obecném případě si definujeme globální proměnnou (např. blnUdalostiZapnuty) a v událostních procedurách prvků testujeme její stav. Velmi často při spouštění formuláře (procedura UserForm_Initialize) plníme prvky výchozími hodnotami. Už změna obsahu prvního z nich přesune probíhající kód do událostní procedury Change tohoto prvku. Pokud v ní pracujeme s dalším prvkem, může dojít k řetězové reakci událostí Change. Proto je UserForm_Initialize prvním místem, kde bychom měli s naší globální proměnnou pracovat (přiřadit False na začátku a True na konci).

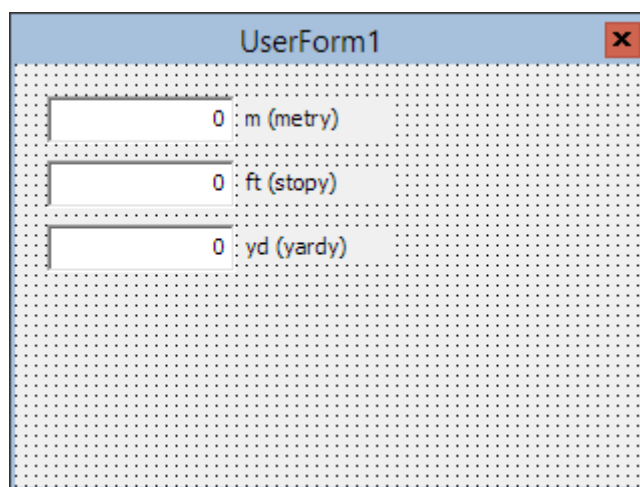
V praxi je běžné, že se ovlivňuje více prvků navzájem. Tady už si s jednou globální proměnnou nevystačíme.

Příklad ukazuje převodník mezi délkovými jednotkami metr (m) – stopa (ft) – yard (yd). Samotné výpočty se řídí prostou trojčlenkou.

1 yd = 3 ft

1 ft = 0,3048 m

1 yd = 0,9144 m



Provázané TextBoxy – návrh

Pro každý prvek založíme na úrovni modulu vlastní proměnnou (jakýsi přepínač).

1	<code>Private blnMetrStopUdalost As Boolean</code>
2	<code>Private blnStopaStopUdalost As Boolean</code>
3	<code>Private blnYardStopUdalost As Boolean</code>

Při startu formuláře “vypneme” u všech prvků kromě jednoho události a u tohoto prvku si zajistíme, aby jeho událost proběhla. O to se postará naplnění prvku (u TextBoxu přiřazení hodnoty do vlastnosti Text, u jiných prvků jako je ComboBox metoda AddItem).

1	<code>Private Sub UserForm_Initialize()</code>
2	
3	<code> 'povoleni udalosti pro prvni prvek</code>
4	<code> blnMetrStopUdalost = False</code>
5	<code> 'zakaz udalosti pro ostatni prvky</code>
6	<code> blnStopaStopUdalost = True</code>
7	<code> blnYardStopUdalost = True</code>
8	
9	<code> 'prirazeni hodnoty do prvnioho prvku</code>
10	<code> 'a vynuceni udalosti</code>
11	<code> txtMetr.Text = 1</code>
12	
13	<code>End Sub</code>

Zpracování událostních procedur jednotlivých textových polí pak bude mít následující podobu:

```
1 Private Sub txtMetr_Change()  
2  
3     'je povolena udalost pro tento prvek?  
4     If blnMetrStopUdalost = False Then  
5  
6         'zakaz udalosti pro ostatni prvky  
7         blnStopaStopUdalost = True  
8         blnYardStopUdalost = True  
9  
10        'prepocet ostatnich prvku  
11        txtStopa.Text = Format(txtMetr.Text / 0.3048, "0.00000")  
12        txtYard.Text = Format(txtMetr.Text / 0.9144, "0.00000")  
13  
14        'povoleni udalosti pro ostatni prvky  
15        blnStopaStopUdalost = False  
16        blnYardStopUdalost = False  
17  
18    End If  
19  
20 End Sub  
21  
22 Private Sub txtStopa_Change()  
23  
24     'je povolena udalost pro tento prvek?  
25     If blnStopaStopUdalost = False Then  
26  
27         'zakaz udalosti pro ostatni prvky  
28         blnMetrStopUdalost = True  
29         blnYardStopUdalost = True  
30  
31        'prepocet ostatnich prvku  
32        txtMetr.Text = Format(0.3048 * txtStopa.Text, "0.00000")  
33        txtYard.Text = Format(txtStopa.Text / 3, "0.00000")  
34  
35        'povoleni udalosti pro ostatni prvky  
36        blnMetrStopUdalost = False  
37        blnYardStopUdalost = False  
38  
39    End If  
40  
41 End Sub  
42  
43 Private Sub txtYard_Change()  
44  
45     'je povolena udalost pro tento prvek?  
46     If blnYardStopUdalost = False Then  
47  
48         'zakaz udalosti pro ostatni prvky  
49         blnMetrStopUdalost = True  
50         blnStopaStopUdalost = True  
51  
52        'prepocet ostatnich prvku  
53        txtMetr.Text = Format(0.9144 * txtYard.Text, "0.00000")  
54        txtStopa.Text = Format(3 * txtYard.Text, "0.00000")  
55  
56        'povoleni udalosti pro ostatni prvky  
57        blnMetrStopUdalost = False  
58        blnStopaStopUdalost = False  
59  
60    End If  
61  
62 End Sub
```

A screenshot of a VBA UserForm titled "UserForm1". It contains three text boxes stacked vertically. The first text box contains the number "1" and is followed by the text "m (metry)". The second text box contains "3,28084" and is followed by "ft (stopy)". The third text box contains "1,09361" and is followed by "yd (yardy)".

Provázané TextBoxy - vstupní TextBox č. 1

A screenshot of a VBA UserForm titled "UserForm1". It contains three text boxes stacked vertically. The first text box contains "0,30480" and is followed by "m (metry)". The second text box contains "1" and is followed by "ft (stopy)". The third text box contains "0,33333" and is followed by "yd (yardy)".

Provázané TextBoxy - vstupní TextBox č. 2

A screenshot of a VBA UserForm titled "UserForm1". It contains three text boxes stacked vertically. The first text box contains "0,91440" and is followed by "m (metry)". The second text box contains "3,00000" and is followed by "ft (stopy)". The third text box contains "1" and is followed by "yd (yardy)".

Provázané TextBoxy – vstupní TextBox č. 3

Funguje to vlastně velmi jednoduše. Procedura zjistí, je-li událost v rámci daného prvku povolena. Pokud ano, zakáže události všem ostatním, vykoná na nich, co potřebuje, a na konci blokaci odstraní.

Proč u TextBoxu pracujeme s vlastností Text a ne Value? Je to proto, že vlastnost Value nezohledňuje české prostředí a například desetinnou čárku. Čísla zde zaokrouhlujeme na pět desetinných míst.

Nyní už by pro vás nemělo být těžké nechávat kupříkladu uživatele vyplnit libovolná tři textová pole pro výpočet charakteristik trojúhelníku a dopočítávat zbylé parametry.

Příloha:

[provazane_textboxy.zip](#)