



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

## DUM 08 téma: Rozhodování v kódu VBA

<b>ze sady:</b>	<b>03</b>	<b>tematický okruh sady:</b>	<b>Tvorba skript a maker</b>
<b>ze šablony:</b>	10 Algoritmizace a programování	<b>určeno pro:</b>	4. ročník
<b>vzdělávací obor:</b>	18-20-M/01 Informační technologie - Aplikace osobních počítačů		
<b>vzdělávací oblast:</b>	odborné vzdělávání		
<b>číslo projektu:</b>	CZ.1.07/1.5.00/34.0066		
<b>anotace:</b>	DUM seznamuje s strukturami pro rozhodování v prostředí VBA. Je zde vše popsáno teoreticky a ukázáno pro názornost na příkladech.		
<b>metodika:</b>	viz metodický list VY_32_INOVACE_10308ml.pdf		
<b>datum tvorby:</b>	1.4.2013		

## Rozhodování v kódu VBA

### *If-Then*

Jde o větvení běhu programu do dvou různých alternativ. Program se dělí dle splnění nebo nesplnění nějakého výrazu. Vždy je dojde k vykonání pouze jedné alternativy v závislosti na vyhodnocení výrazu. Příkaz pro podmíněný příkaz je If-Then. Je důležité říct, že jde o základní příkaz, který umožňuje dělení kódu do více alternativ. Syntaxe tohoto podmíněného příkazu vypadá následovně:

```
If výraz Then příkaz
```

Jak to celé pracuje? Je-li výraz splněn, vykoná se příkaz za klíčovým slovem Then, pak se pokračuje vykonáváním kódu pod řádkem s rozhodováním. Při nesplnění výrazu se rovnou začne vykonávat pak kód pod řádkem s rozhodováním.

Další variantou příkazu If je použití rozhodovacího bloku. Syntaxe vypadá pak takto:

```
If výraz Then
    Příkaz1
    Příkaz2
Else
    Příkaz3
    Příkaz4
End If
```

V případě splnění výrazu se vykonají příkazy v prvním bloku příkazů tzn. Příkaz 1 a 2, poté se kód pokračuje během pod End If. Pokud nedojde ke splnění výrazu, tak se vykonají příkazy v druhém bloku příkazů tzn. Příkaz 3 a 4, poté se kód pokračuje během pod End If. Klíčové slovo End If ukončuje celý rozhodovací blok.

### ***Praktický příklad pro použití příkazu If-Then***

Nejprve si vytvoříme nové makro, ale pomocí záznamu nového makra, který již znáte s předchozích DUMů. Jednu libovolnou buňku jako měnu a ve druhé se budou veapsané hodnoty zarovnávat na střed. Po té zastavíme záznam makra. Otevřeme si dialogové okno maker, vybereme námi vytvořené makro a zvolíme upravit. Nyní se zobrazí kód podobný tomuto:

```

Sub moje_makro()
'
' moje_makro Makro
'
'
'
Selection.NumberFormat = "#,##0.00 $"
Range("D3").Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
End Sub

```

Kód se bude lišit podle toho do jakých buněk jsme klikali. Nyní si z něho vypreparujeme tyto příkazy:

```

Sub moje_makro()
'
' moje_makro Makro
'
'
'
Selection.NumberFormat = "#,##0.00 $"
'
'
Selection.HorizontalAlignment = xlCenter
'
'
End Sub

```

Zachovali jsme si z kódu pouze dva řádky. První formátuje buňku jako měnu a druhý v buňce zarovnává vepsané hodnoty na střed. Kód upravíme tak, že pokud v buňce A1 bude číslo větší než 100, tak se buňka naformátuje jako měna. V opačném případě tedy pokud buňka menší než 100, její obsah bude zarovnán na střed.

```

Sub Makro1()
'
' Makro1 Makro
'
'
'
Range("A1").Select
If ActiveCell.FormulaR1C1 > 100 Then
'
'
Selection.NumberFormat = "#,##0.00 $"
Else
'
'
Selection.HorizontalAlignment = xlCenter
End If

```

Kombinací generovaného kódu a ručně psaného kódu si mnohdy můžeme ulehčit práci.

## Select-Case

Nyní si ukážeme poslední možnost pro rozhodování v prostředí VBA. Klasický příkaz pro rozhodování má nevýhodu, že se pomocí něho lze vybrat jen mezi dvěma alternativami. Jak to uděláme, když potřebujeme vybrat z více alternativ vykonávaného kódu? Jde to úplně jednoduše, stejně jako ve všech moderních programovacích jazycích existuje rozhodovací blok Select Case, který se někdy označuje jako přepínač. Jeho syntaxe vypadá takto:

```
Select Case výraz
    Case nějaká hodnota výrazu
        blok příkazů 1
    Case nějaká hodnota výrazu
        blok příkazů 2
    Case nějaká hodnota výrazu
        blok příkazů 3
    Case Else
```

Zde bude proveden vždy ten příkaz nebo blok příkazů, u kterého bude se hodnota Case rovnat výrazu za Select Case. Celý blok Select Case je povinně zakončen příkazem Case Else. Následný příklad testuje jaká hodnota je zadána v buňce A1 a podle toho zobrazí příslušné okno s hlášením.

```
Sub Makro1()
'
' Makro1 Makro
'
Range("A1").Select

Select Case ActiveCell.FormulaR1C1
    Case 1
        MsgBox "Byla zadána jednička"
    Case 2
        MsgBox "Byla zadána dvojka"
    Case 3
        MsgBox "Byla zadána trojka"
    Case 4
        MsgBox "Byla zadána čtyřka"
    Case 5
        MsgBox "Byla zadána pětka"
End Select

End Sub
```

**Zdroje:**

Archiv autora



Autorem materiálu a všech jeho částí, není-li uvedeno jinak, je Ing. Daniel Velek

Material je publikován pod licencí Creative Commons 