

TÉMA: Úprava maker – rozhodování, příkaz If..Then..Else

Sekretářka společnosti „Naše zahrada“ potřebuje upravit makra vytvořená pomocí záznamu tak, aby vyhovovala jejím požadavkům. Pro úpravy využije Editor jazyka Visual Basic a jeho vybrané příkazy pro zobrazování oken, práci s proměnnými a rozhodování v průběhu chodu makra.

Zadání:

Otevřete sešit *UpravaMakraRozhodovani.xlsm*. V dokumentu se nachází již vytvořené makro *Nadpis*, které po spuštění naformátuje na listu *Úvod* buňky s nadpisem. Makro upravte následujícím způsobem (pro úpravu makra využijte prostředí Editoru jazyka Visual Basic a jeho vybrané příkazy):

1. Do makra přidejte rozhodování – **když bude** na listu *Úvod* v buňce B5 číslo 1, makro **změní** při formátování sloučené buňky **barvu výplně** na barvu žlutou, pokud bude v buňce hodnota jiná, formátování nadpisu zůstane se světle zelenou barvou výplně.
2. Zjištění příkazů pro formátování výplně buňky žlutou barvou proveďte s využitím záznamu pomocného makra.
3. Upravené **makro vyzkoušejte**.
4. **Sešit uložte a uzavřete**.

Řešení

Algoritmus

Řešení:

Karta **Soubor/Otevřít** (nebo tlačítko **Otevřít** na panelu nástrojů **Rychlý přístup**, popř. klávesová zkratka **Ctrl+O**). V dialogovém okně **Otevřít** nalézt soubor dle zadání, tlačítko **Otevřít**. Sešit má příponu **.xism**, což je formát podporující makra. Po otevření sešitu se zobrazí informační pruh s oznámením, že v sešitě se nachází aktivní obsah (makro). Povolit práci s tímto makrem – tlačítko **Povolit obsah**. Pozn. Zobrazení informačního pruhu záleží na nastavení zabezpečení Excelu (karta **Vývojář**/skupina **Kód/příkaz Zabezpečení maker**/v dialogovém okně **Centrum zabezpečení**/sekce **Nastavení maker/Zakázat všechna makra s oznámením** – Excel při otvírání sešitu upozorní uživatele na přítomnost makra, uživatel může práci s tímto makrem povolit, pokud mu důvěřuje).

Makro vytvořené pomocí záznamu se ukládá na pozadí v podobě příkazů jazyka Visual Basic for Application (VBA) do procedury s názvem makra. Úpravy maker se provádějí v Editoru jazyka Visual Basic – karta **Vývojář**/skupina **Kód/příkaz Visual Basic**. Pozn. Pokud není karta **Vývojář** **zobrazena**, lze ji zobrazit - karta **Soubor/příkaz Možnosti**/sekce **Přizpůsobit pás karet**/zatrhnout položku **Vývojář**.

Spustí se Editor jazyka Visual Basic, který je součástí instalace Excelu. V okně projektů (**Project – VBAProject** v levé části) se nachází seznam právě otevřených sešitů včetně šablony *Personal*; pro každý sešit je vytvořen samostatný projekt (název sešitu je vždy uveden v závorce). Pozn. Pokud není okno projektů zobrazeno, lze je zobrazit přes nabídku **View/Project Explorer**. Je-li vytvořeno v sešitě makro, nachází se v příslušném projektu pod skupinou **Modules/Module1**. Zobrazení kódu makra – dvakrát kliknout na položku **Module1** u příslušného projektu, zobrazí se okno kódu s příkazy zaznamenaného makra. Každé makro je uloženo do samostatné procedury, která začíná příkazem **Sub NázevMakra ()** a končí příkazem **End Sub**. Uprostřed procedury se na každém řádku nacházejí příkazy makra. Text označený zeleně (začínající apostrofem ' má funkci komentářů, které nejsou makrem prováděny).

Algoritmus slovně:

1. Pro rozhodování bude použit příkaz **If..Then..Else**. Za klíčové slovo **If** se uvádí podmínka, za klíčové slovo **Then** se uvádí příkazy, které makro provede, pokud podmínka platí (žlutý vzorek buňky). Následuje klíčové slovo **Else** a příkazy, které makro provede, pokud podmínka neplatí (zelený vzorek buňky).
2. Podmínkou bude v tomto případě testování, zda se v buňce B5 nachází číslo 1. Pro zjištění obsahu buňky slouží příkaz **Range("B5")**. Podmínka bude tedy vypadat takto: **Range("B5")=1**.
3. Abychom zjistili příkazy VBA, které formátují výplň buňky na žlutou barvu, zaznamenáme pomocného makro (např. s názvem *Pom*), které pouze změní výplň buňky na příslušnou barvu. Zjistíme tak číslo vlastnosti **Interior.Color** pro žlutý vzorek aktivní buňky.

Provedení:

Před úpravou makra doporučujeme makro nejprve vyzkoušet – karta **Vývojář**/skupina **Kód/příkaz Makra**/v dialogovém okně **Makra** vybrat makro *Nadpis*, příkaz **Spustit**.

Úprava makra:

1. Nejprve vytvořit pomocné makro pro zjištění příkazů, které naformátují výplň buňky na žlutou barvu – označit libovolnou buňku, karta **Vývojář**/skupina **Kód/příkaz Záznam makra**/v dialogovém okně **Záznam makra** nadefinovat název (např. *Pom*), v seznamu **Uložit makro do** zvolit např. položku **Tento sešit** (makro pak bude v Editoru jazyka VBA uloženo do projektu s názvem sešitu), spustit záznam makra – tlačítko **OK**. V záznamu makra pouze naformátovat výplň buňky žlutou barvou – karta **Domů**/skupina **Písmo**/šipka příkazu **Barva výplně**/zvolit žlutou barvu. Ukončit záznam makra – karta **Vývojář**/skupina **Kód/příkaz Zastavit záznam**. Pomocné makro bude vytvořeno a uloženo v podobě procedury pomocí příkazů jazyka VBA.
2. Úprava makra *Nadpis* - karta **Vývojář**/skupina **Kód/příkaz Makra**/v dialogovém okně **Makra** vybrat makro *Nadpis*, příkaz **Upravit**. Otevře se prostředí editoru **Microsoft Visual Basic** s již vytvořeným makrem *Nadpis*.
3. V makru *Nadpis* nalézt příkazy pro formátování výplně buňky (pokud je neznáme, můžeme se podívat do příkazů pomocného makra *Pom*, které jsme vytvořili (nachází se ve stejném projektu jako makro *Nadpis*, ovšem v modulu **Module2** – platí pouze za podmínky, že jsme makro uložili do aktuálního sešitu. Jinak se makro nachází v projektu *Personal.xls* nebo v novém sešitě). Jedná se o skupinu příkazů začínajících řádkem **With Selection.Interior**. Skupina končí příkazem **End With**. Pozn. Jazyk VBA je objektově orientovaný. Znamená to, že jednotlivým objektům (označená oblast, list, sešit atd.) můžeme nastavovat jejich vlastnosti a spouštět metody pomocí tečkové notace. Příkaz **Selection.Interior.Color = 65535** znamená, že vybrané oblasti buněk (objekt **Selection**) bude nastavena vlastnost výplně (**Interior**), konkrétně se jedná o barvu (**Color**), každá barva je reprezentována číslem. Příkaz **With** se používá pouze pro zkrácení zápisu – to, co je uvedeno na řádku za příkazem **With** se bude v následujících řádcích vždy opakovat před příkazy začínajícími tečkou až po řádek **End With**.
4. Vložit prázdný řádek nad příkaz **With Selection.Interior** – vložit kurzor na začátek řádku, klávesou **Enter** odřádkovat. Před tyto příkazy vložit začátek příkazu pro rozhodování (příkaz **If..Then..Else**). Začátek bude obsahovat podmínku, konkrétně testování, zda se v buňce B5 nachází hodnota 1. Pro zjištění hodnoty v buňce použijeme příkaz **Range()**:
If Range("B5")=1 Then
5. Pod tento řádek vložit příkazy pro formátování výplně buňky žlutou barvou – zkopírovat příkazy z pomocného makra *Pom*: zkopírovat pouze příkazy začínající řádkem **With Selection.Interior** a končící řádkem **End With**. Pozn. Pokud jsme makro *Pom* uložili do aktuálního dokumentu, nachází se v projektu *UpravaMakraRozhodovani*, v modulu **Module2**.
6. Za vloženými příkazy odřádkovat a vepsat klíčové slovo **Else** (prostřední část příkazu **If..Then..Else**).
7. Za příkazem **Else** nechat příkazy pro formátování výplně zelenou barvou – začínají řádkem **With Selection.Interior** a končí řádkem **End With**.

8. Za těmito příkazy opět odřádkovat a vložit klíčové slovo **End If** (poslední část příkazu **If . .Then . .Else**).
9. **Vyzkoušení makra:** zobrazit aktuální sešit, zobrazit list *Úvod*, do buňky B5 vepsat hodnotu (je vhodné vyzkoušet obě varianty – nejprve hodnotu *1*, podruhé vepsat hodnotu jinou). Karta **Vývojář**/skupina **Kód/příkaz Makra**/v dialogovém okně **Makra** najít vytvořené makro, tlačítko **Spustit**. Pozn. Makro lze spustit také v prostředí Editoru jazyka Visual Basic prostřednictvím nabídky **Run/Run Macro**.
Pozn. Pokud dojde při provádění makra k chybě, upozorní na ni Excel uživatele zprávou. Dále lze pokračovat tlačítkem **End** – ukončení makra, nebo **Debug** – přerušení chodu makra. V obou případech se zobrazí Editor jazyka Visual Basic s kódem makra, ve druhém případě bude žlutě zvýrazněn příkaz, ve kterém došlo k chybě. Po odstranění chyby je nutné makro ukončit (nabídka **Run/Reset**), aby jej bylo možné spustit opět od začátku. Pokud je kód v Editoru zobrazen červeně, znamená to, že je v příkazu syntaktická chyba (často chybí závorka, uvozovky, čárka apod.).
10. Uložit sešit - karta **Soubor/Uložit** (nebo tlačítko **Uložit** na panelu nástrojů Rychlý přístup nebo klávesy **Ctrl+S**). Zavřít sešit - karta **Soubor/Zavřít** (nebo tlačítko **Zavřít** – **x** v pravém horním rohu okna).

Zpět na zadání

Algoritmus:

```
Sub Nadpis()  
'  
' Nadpis Makro  
'  
    Sheets("Úvod").Select  
    Range("A2:K3").Select  
  
' ... Zde následují příkazy pro další formátování buněk (makro bylo  
zkráceno z důvodu šetření místa!  
  
If Range("B5") = 1 Then 'Rozhodování, zda se v buňce nachází číslo 1  
  
'Podmínka platí, buňka bude mít žlutou výplň  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .Color = 65535  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
  
Else  
  
'Podmínka neplatí, buňka bude mít zelenou výplň  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .Color = 5296274  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
  
End If 'Konec příkazu If..Then..Else  
  
    With Selection  
        .HorizontalAlignment = xlCenter  
        .VerticalAlignment = xlCenter  
        .WrapText = False  
        .Orientation = 0  
        .AddIndent = False  
        .IndentLevel = 0  
        .ShrinkToFit = False  
        .ReadingOrder = xlContext  
        .MergeCells = True  
    End With  
End Sub
```

[Zpět na zadání](#)