# VISUAL BASIC FOR APPLICATIONS - V PROSTREDÍ MS OFFICE

Ing. Roland Andrássy Strojnícka fakulta TU Košice Katedra automatizácie a riadenia Park Komenského č.8 041 87 Košice tel.: 055/6022349 e-mail: Roland.Andrassy@tuke.sk

## 1. Úvod

Práca v programoch kancelárskeho balíka Microsoft Office pri spracovávaní veľkého množstva údajov je práca na niekoľko hodín. Pritom sú mnohé operácie rutinné a často sa opakujú. Na z automatizovanie a tým aj urýchlenie vykonávaných výpočtov a úprav v dokumentoch alebo zošitoch aplikácií Ms Office sa používajú makrá a programy napísané v programovacom jazyku Visual Basic for Application.

## 2. Visual Basic for Applications

Programovací jazyk Visual Basic for Applications (VBA) je jediný programovací jazyk podporovaný aplikáciami kancelárskeho balíka Microsoft Office. Je to objektovo orientovaný jazyk. Objekty sú prvky predstavujúce nejakú časť aplikácie ako je zošit, graf alebo ovládací prvok formulára. Pomocou VBA možno dosiahnuť mnohých úloh, ktoré sú za pomoci užívateľského prostredia Excelu alebo Wordu nemožné. Pomocou VBA možno napr.

- vytvárať pokročilé nástroje overovania dát,
- navrhovať formuláre slúžiace ku zberu dát a vytváraniu ich zostáv,
- dynamicky získavať údaje z externých databáz,
- vytvárať zostavy kontingenčných tabuliek stlačením jediného tlačidla,
- zapisovať kód, ktorý dynamicky komunikuje s užívateľom pomocou dialógových okien a vstupných polí,
- dočasne si ukladať údaje do pamäte,
- reagovať na udalosti, ako je otvorenie listu (v slovenskej verzii je to hárok) alebo zmena údajov na liste,
- prípadne riadiť inú aplikáciu, napr. Word, Access.

## 3. Makro

Programový kód sa zapisuje ako postupnosť príkazov do procedúr ináč nazývaných ako makrá [1].

**Makro** sa používa hlavne na zrýchlenie a zautomatizovanie rutinných a často sa opakujúcich prác, čím sa zvýši produktivita práce a radikálne sa zníži doba vykonávania práce. V makre je zaznamenaná postupnosť prác, ktoré sa majú vykonať. Napríklad nasledujúce makro dokáže vypočítať súčin dvoch čísel, ktoré postupne zapíšete do textových polí a nakoniec výsledok vypíše na obrazovku a zapíše do bunky s adresou A5:

Sub Vypocet()

'Deklarácia premenných Dim a As Integer, b As Integer, c As Integer 'Načítanie prvej hodnoty a = InputBox("Napíšte prvé číslo") 'Načítanie druhej hodnoty b = InputBox("Napíšte druhé číslo") 'Výpočet c = a \* b 'Výsledok MsgBox c Range("A5") = c End Sub

Makro sa dá priradiť k určitej klávesovej skratke, k tlačidlu vytvorenému na panely s nástrojmi prípadne ho môžeme vyvolať pri práci s formulárom.

Každé makro (procedúra) sa začína slovami *Sub NázovMakra()* a končí *End Sub*.

## 4. Editor jazyka Visual Basic

Aby sme mohli programovať v programoch balíka Ms Office, musíme si spustiť Editor jazyka Visual Basic – *Nástoje* (angl. Tools) -> *Makro* (Macro) -> *Editor jazyka Visual Basic* (Visual Basic Editor) [rýchlejšie cez Alt + F11]. Okno editora je na obr. 1.



Obr. 1 – Editor jazyka Visual Basic 1. menu so štandardným panelom nástrojov, 2. VBA project, 3. Properties.

Prostredie editora je tvorené:

- Menu so štandardným panelom nástrojov,
- VBA project,
- Properties vlastnosti.

## 5. Menu, VBA Project a Properties

**Menu** obsahuje všetky príkazy, ktoré sú potrebné pri programovaní. Na štandardnej lište nájdeme tlačidlá s najvyužívanejšími príkazmi, ako napr. View Ms Word(Excel) – prepnúť do ..., Insert UserForm (Module, Class Modul, Procedure) – vložiť formulár (modul modul triedy, procedúru), Save current document – uložiť dokument, Run macro – spustiť makro, Break – pozastaviť makro, Reset – zastaviť makro, Design mode – dizajn mód, Object Browser – prehliadač objektov, atď.

Každý zošit aplikácie Excel prípadne dokument aplikácie Word obsahuje **VBA project**. Tento projekt obsahuje všetky komponenty súvisiace so zošitom, vrátane modulov organizujúcich VBA kód a všetkých nami vytvorených dialógových okien. Môže obsahovať tieto komponenty:

- Aplikačné objekty (Microsoft Excel Objects) zahrňujú objekty hostiteľskej aplikácie. V Exceli sú to všetky listy v zošite a všetky listy grafov. Každý objekt má svoj vlastný modul kódu.
- Formuláre (Forms) sú dialógové okná užitočné pre také činnosti ako napr. zber informácií od užívateľa. Každý formulár má svoj vlastný modul kódu.
- <u>Štandardné moduly</u> (Modules) sú typické moduly kódu, ktoré obsahujú generický kód používaný v celom zošite.
- <u>Moduly tried</u> (Class Modules) sú špecializované moduly kódu, ktoré sa používajú k definovaniu nových objektov.

Vo všetkých moduloch kódu je kód zoskupovaný do oddelených blokov označovaných za procedúry. Procedúra obsahuje jeden alebo viac riadkov kódu vykonávajúceho určitú činnosť.

Okno **Properties** umožňuje nastavovať vlastnosti jednotlivých objektov. Tieto vlastnosti závisia od nastavovaného objektu, či už je to formulár, tlačidlo, textbox alebo hárok. V combo zozname je napísaný názov objektu (tučné písmo) a typ objektu. V ľavom stĺpci sú názvy vlastností a v pravom stĺpci sú hodnoty týchto vlastností. Vlastnosť name majú všetky objekty, musí byť jedinečná, pretože určuje názov objektu. Úpravou vlastností meníme umiestnenie, vzhľad a správanie sa objektov.

#### 6. Microsoft Office References

Aby nám program fungoval v celom prostredí Ms Office, je nutné mať správne

nastavené odkazy na dynamické knižnice, v ktorých sú zadefinované funkcie, ktoré chceme používať. V menu Tools -> References je možné vybrať z dostupných knižníc tie, ktoré sú práve potrebné.

Ak pracujete v aplikácii Ms Word sú zaškrtnuté knižnice Microsoft Office XX. Object Library a Microsoft Word XX. Object Library (XX značí verzia Office, napr. 10). Pre prenos údajov do Ms Excel je nutné zaškrtnúť aj Microsoft Excel XX. Object Library. Potom je možné využívať funkcie Excelu aj v prostredí Wordu alebo prenášať údaje medzi týmito aplikáciami.

#### 7. Objekty, Vlastnosti a Udalosti

Objekt je niečo, čo môže bvť programované alebo riadené. Sú to napr. zošit, hárok, formulár, textbox, či tlačidlo. Každý objekt má vlastnosti, ktoré je možné nastavovať alebo využívať ich hodnoty. K hodnotám vlastností sa pristupuje takto: <Názov Objektu>.<Vlastnosť> . Vlastnosti môžeme nastavovať t.j. priradiť im hodnotu, napr. Application.Caption = "My Word" nastaví názov programu Word na My Word (pôvodne Microsoft Word). Tieto vlastnosti môžme aj čítať a v programme ich využiť, napr. nazov = Application.Caption uloží do premennej "nazov" názov aplikácie.

Udalosti umožňujú vykonávať akcie na danom objekte. Napr. Application.Quit zavrie všetky dokumenty.

Viac objektov rovnakého typu sa nazýva kolekcia. Názov kolekcie je v množnom čísle (má koncovku "s"), napr. Documents, Worksheets. Objekty z kolekcie sa vyberajú buď indexom prvku v zátvorkách alebo jeho menom. Výber indexom je napr. Worksheets(1).Name = "Prvý list", čiže listu s indexom 1 nastavíme meno na text "Prvý list". Výber menom je napr. Worksheets("Prvý list").Activate, aktivuje sa list s menom "Prvý list".

Vlastnosti a udalosti jednotlivých objektov je možné nájsť pomocou *Object Browser* (menu View -> Object Browser alebo kláves F2). V ľavom stĺpci sú všetky triedy Classes, ktoré je možné použiť a v pravom okne sa zobrazujú vlastnosti a udalosti vybratej triedy, ktoré je možné využívať. Tieto vlastnosti a udalosti sa zobrazujú pri písaní zdrojového kódu. Napr. ak napíšeme do procedúry (makro) "Application.", zobrazí sa nám za bodkou zoznam vlastností a udalostí, ktoré sú použiteľné pre daný objekt (pozri Obr. 2).

	ación.	
End Sub	📾 Width	~
	🔊 Windows	
	WindowsForPens	
	MindowState	
	🔊 Workbooks	
	B WorksheetFunction	
	🔊 Worksheets	*

Obr.2 – Zobrazenie vlastností a udalostí

Potom si stačí, kurzorovými šípkami hore a dole, vybrať položku, stlačiť bodku, ak vybratý objekt obsahuje ďalšie vlastnosti alebo objekty a zobrazí sa nový zoznam použiteľných vlastností a udalostí. Napr. *Application.Worksheets.Add* – udalosť Add pridá čistý hárok do aktuálneho zošita.

#### 8. Príklady

Nasledujúce príklady treba písať do procedúry (makra) v programe Excel.

#### Zmeny prostredia aplikácie:

Sub Environment() With Application 'názov aplikácie Excel .Caption = "My Excel application" 'schová vzorcový panel .DisplayFormulaBar = False 'schová status bar .DisplayStatusBar = False End With With ActiveWindow 'schová mriežku (hranice medzi bunkami) .DisplayGridlines = False schová označenia stĺpcov a riadkov .DisplayHeadings = False 'schová vodorovný posuvník .DisplayHorizontalScrollBar = False 'schová zvislý posuvník .DisplayVerticalScrollBar = False 'názov hárku .Caption = "Môj hárok" End With End Sub

Po zámene slova False za True sa schované veci zobrazia.

Kopírovanie obsahu bunky z Ms Excel do dokumentu Ms Word: Makro musí byť vytvorené v Ms Excel!

Sub Kopirovanie()

'spustenie aplikácie Ms Word Set aplWord = New Word.Application 'zviditel'nenie aplikácie aplWord.Visible = True 'vytvorenie nového dokumentu Set doc = aplWord.Documents.Add 'vloženie textu z bunky A1 z Ms Excel do dokumentu Ms Word Word.Selection.TypeText ("Text z bunky A1:" & vbTab & Excel.Range("A1")

## End Sub

#### Zmena formátu vybranej bunky:

Sub ZmenaFormatu() With Selection vodorovné zarovnanie vľavo .HorizontalAlignment = xlLeft 'zvislé zarovnanie nadol .VerticalAlignment = xlBottom .WrapText = True 'zalomit' text . Orientation = 0'uhol sklonu .MergeCells = True 'zoskupiť viac buniek End With With Selection.Font .Name = "Arial" 'písmo .FontStyle = "Normálne" 'rez písma normálny .Size = 11'veľkosť písma 11 .Strikethrough = True 'preškrtnuté písmo .Superscript = True 'použit' horný index .Subscript = False 'nepoužiť dolný index 'podčiarknuté písmo nie je .Underline = xlUnderlineStyleNone 'farba písma automatická .ColorIndex = xlAutomatic End With End Sub

#### 9. Záver

Programovanie v jazyku VBA umožňuje automatizáciu práce v programoch kancelárskeho balíka Microsoft Office. Takýmto spôsobom je možné urýchliť rutinnú a často sa opakujúcu prácu. Pomocou makier je možné prenášať údaje medzi aplikáciami. Napr. údaje z meraní sa spracujú v programe Ms Excel a protokol z merania sa vytvorí v programe Ms Word.

### 10. Literatúra

- [1] MARTIN, M. 2000. *Excel 2000, EdiceProfesional, II. díl.* Praha: Grada Publishing, 2000.
- [2] KALIŠ, J. 1995. Excel Učebnice programování. Praha: GComp, 1995. 240 s. ISBN 80-85649-33-0