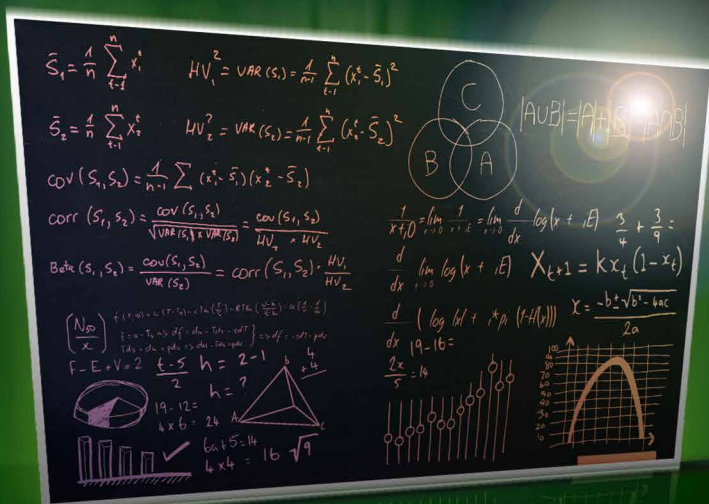




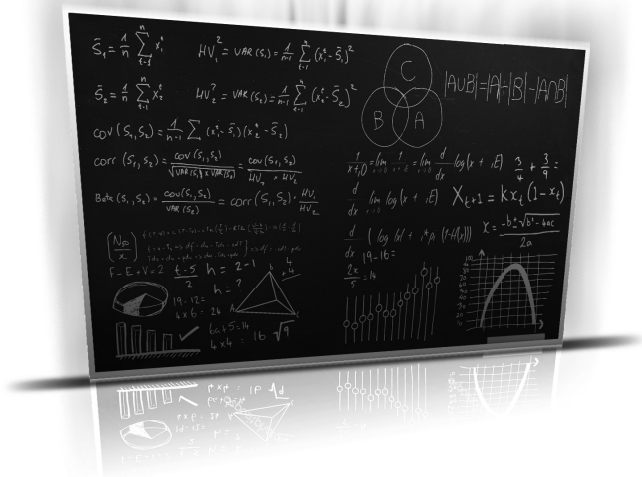
Programování v Excelu

záznam, úprava a programování maker 2019

Marek Laurenčík, Michal Bureš

- Práce s editorem VBA a základy jazyka VBA
- Tvorba maker a vlastních funkcí
- Práce s buňkami, listy, sešity a externími soubory
- Práce s grafy a kontingenčními tabulkami pomocí kódu
- Tvorba vlastních dialogových oken
- Export a import dat
- Spolupráce Excelu s dalšími aplikacemi
- Tvorba vlastních ovládacích prvků a doplňků





Programování v Excelu

záznam, úprava a programování maker 2019

Marek Laurenčík, Michal Bureš

Grada Publishing

Upozornění pro čtenáře a uživatele této knihy

Všechna práva vyhrazena. Žádná část této tištěné či elektronické knihy nesmí být reprodukována a šířena v papírové, elektronické či jiné podobě bez předchozího písemného souhlasu nakladatele. Neoprávněné užití této knihy bude **trestně stíháno**.

Marek Laurenčík, Michal Bureš

Programování v Excelu 2019

záznam, úprava a programování maker

Vydala Grada Publishing, a.s.
U Průhonu 22, Praha 7
obchod@grada.cz, www.grada.cz
tel.: +420 234 264 401
jako svou 8265. publikaci

Odpovědný redaktor Petr Somogyi
Sazba Petr Somogyi
Počet stran 240
První vydání, Praha 2021
Vytiskla TISKÁRNA V RÁJLI, s.r.o., Pardubice

© Grada Publishing, a.s., 2021
Cover Design © Grada Publishing, a. s., 2021
Cover Photo © Depositphotos/riariu, vectorfusionart

Názvy produktů, firem apod. použité v knize mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

ISBN 978-80-271-4572-0 (ePub)
ISBN 978-80-271-4571-3 (pdf)
ISBN 978-80-271-3145-7 (print)

	Úvod	9
1	Tvorba maker záznamem.....	11
	1.1 Použití maker v Excelu	11
	1.2 Záznam a spuštění makra.....	12
	1.3 Různé způsoby spuštění makra	15
	1.4 Spuštění makra v jiném sešitu	20
2	Základy jazyka VBA.....	21
	2.1 Práce s editorem VBA.....	21
	2.2 Tvorba makra kódem VBA	24
	2.3 Tvorba uživatelské funkce	25
	2.4 Použití proměnných a vzorců.....	28
	2.5 Deklarace proměnných.....	30
	2.6 Větvení kódu.....	34
	2.7 Příkazy cyklu.....	36
3	Standardní funkce VBA	39
	3.1 Podmíněné výrazy.....	39
	3.2 Matematické funkce	40
	3.3 Funkce pro práci s textem	40
	3.4 Funkce pro práci s datem a časem	43
	3.5 Převodní funkce.....	44
	3.6 Další typy funkcí.....	46
	3.7 Finanční funkce	47
	3.8 Standardní dialogová okna Visual Basicu	50
4	Práce s buňkami a oblastmi.....	53
	4.1 Odkazy na buňky a oblasti.....	53
	4.2 Zpracování tabulky po řádcích.....	57
	4.3 Objektový model Excelu	58
5	Vlastnosti buněk a oblastí	62
	5.1 Identifikace buněk a oblastí	63
	5.2 Vzhled buněk.....	64
	5.3 Zarovnání a slučování buněk.....	67
	5.4 Šířka a zobrazení.....	67
	5.5 Numerický formát buněk.....	68
	5.6 Tvorba vzorců	69



6	Metody buněk a oblastí	72
6.1	Mazání buněk, práce s řádky a sloupci.....	73
6.2	Kopírování a přesun.....	74
6.3	Vyhledávání a nahrazování.....	76
6.4	Dělení textu do sloupců	78
6.5	Plnění oblastí	80
6.6	Řazení a tvorba souhrnu.....	82
6.7	Filtrace.....	84
6.8	Odstranění duplicitních hodnot.....	88
7	Práce s listy, sešity a aplikací Excelu	90
7.1	Práce s listy.....	90
7.2	Práce se sešity.....	95
7.3	Aplikace Excelu.....	98
8	Události.....	104
8.1	Události listu.....	104
8.2	Události sešitu.....	106
8.3	Události aplikace	109
9	Pokročilé techniky.....	111
9.1	Makra s parametry.....	111
9.2	Odchycení chyb.....	112
9.3	Komentáře a hypertextové odkazy	113
9.4	Práce s částí textu v buňce.....	116
9.5	Podmíněné formátování	118
9.6	Rozšířené řazení.....	122
9.7	Formulářové prvky na listu.....	124
10	Analýza a kontrola dat.....	129
10.1	Ověření dat	129
10.2	Práce se scénáři	131
10.3	Iterační výpočet	133
10.4	Prognózy v časových řadách	134
11	Formátované tabulky.....	136
11.1	Tvorba formátované tabulky.....	136
11.2	Úprava vytvořené tabulky.....	137
11.3	Výpočty ve formátované tabulce.....	138
11.4	Souhrny u formátované tabulky.....	139
12	Kontingenční tabulky	140
12.1	Objekty v kontingenční tabulce	140
12.2	Aktualizace a tvorba kontingenční tabulky	141

12.3	Rozmístění polí v kontingenční tabulce	142
12.4	Výpočty a formátování hodnot	144
12.5	Kontingenční tabulky s datovým modelem	146
13	Práce se soubory a vnějšími daty	152
13.1	Práce se soubory, složkami a disky	152
13.2	Import textového souboru	156
13.3	Připojení tabulky z databáze	157
13.4	Přímý přístup k datovým souborům	162
13.5	Práce se soubory XML	163
14	Výstupy	168
14.1	Práce s konci stránek	168
14.2	Nastavení vzhledu stránky	169
14.3	Oblast tisku	171
14.4	Náhled a tisk listu	172
14.5	Export do souboru PDF	172
14.6	Odeslání sešitu e-mailem	173
15	Práce s grafy	174
15.1	Objekty grafu	174
15.2	Tisk a export grafu	175
15.3	Změna dat v grafu	176
15.4	Úprava os	179
15.5	Popisky v grafu	179
15.6	Formátování datových řad	181
15.7	Další možnosti práce s grafy	182
16	Tvorba uživatelských formulářů	185
16.1	Tvorba uživatelského formuláře a jeho spuštění	185
16.2	Základní prvky na formuláři	186
16.3	Vzhled prvků na formuláři	188
16.4	Další prvky na formuláři	189
16.5	Seznamy	191
17	Použití kódu na formuláři	195
17.1	Přístup k formulářovým prvkům	195
17.2	Události formuláře	198
17.3	Nastavení vlastností pomocí kódu	199
17.4	Přepínač a RefEdit	202
17.5	Programování seznamů	204
17.6	Tvorba průvodce	207
17.7	Ovládání formuláře z kódu makra	208

18	Spolupráce s jinými programy	210
18.1	Spuštění jiného programu v makru.....	210
18.2	Spojení Excelu s Wordem.....	212
18.3	Spojení Excelu s Accessem	216
18.4	Spolupráce Excelu s Outlookem.....	218
19	Úprava sešitu pro využití jinými uživateli	220
19.1	Tvorba a úprava ovládacích panelů.....	220
19.2	Práce s pásem karet.....	223
19.3	Ochrana projektu VBA heslem	226
19.4	Tvorba vlastního doplňku	227
20	Úvod do tříd	230
20.1	Vytvoření třídy a její instance.....	230
20.2	Příklad využití třídy v kódu makra	233
20.3	Příklad využití třídy v kódu formuláře.....	234
20.4	Využití třídy pro programování událostí	234
	Rejstřík.....	238

Úvod

Tato kniha je věnována programování v tabulkovém kalkulátoru MS Excel 2019 pomocí jazyka Visual Basic for Applications. U čtenáře nepředpokládáme žádné předběžné znalosti programování, pouze znalost práce s kalkulátorem Excel na uživatelské úrovni.

V knize popisované programovací postupy jsou použitelné i v předchozích verzích Excelu (tedy 2016, popřípadě 2013). Excel 2019 však ve srovnání se staršími verzemi používá odlišný přístup při připojování vnějších dat z relačních databází, jako je MS Access a MS SQL Server. Tento rozdíl se projevuje i při programování a je v knize podrobně rozebraný.

Text knihy nevznikl pouhým doplněním předchozích vydání, ale provedli jsme důkladné přepracování celého textu, zejména se zaměřením na praktickou použitelnost popisovaných programovacích postupů. Přitom jsme vycházeli z vlastních zkušeností při školení na kurzech jazyka VBA a z řady na zakázku vytvořených projektů.

Kromě kapitoly 1, která popisuje tvorbu maker záznamem a různé způsoby jejich spouštění, je možné rozdělit ostatní kapitoly knihy do několika částí. Kapitoly 2 a 3 popisují jazyk Visual Basic. Poznatky, které budou uvedené v těchto kapitolách, můžete využít i u jiných produktů skupiny MS Office, především u MS Accessu. Kapitoly 4 až 8 jsou věnovány základním možnostem a postupům při použití jazyka VBA v Excelu. Kapitoly 8 až 15 rozebírají pokročilé a speciální postupy, které zahrnují programování grafů, kontingenčních tabulek včetně datového modelu, analýzu a kontrolu dat, práci s výstupy včetně exportu do formátu PDF a práce s formátovanými tabulkami. Značná pozornost je věnována programovacím postupům pro práci s vnějšími daty, tj. s textovými soubory a soubory ve formátu XML, a postupům při tvorbě tabulek připojených k relačním databázím MS Access a MS SQL Server. V kapitole 16 a 17 naleznete popis práce s uživatelskými formuláři (což jsou vlastní dialogová okna), a to jak uživatelským způsobem, tak i ve spojení s programovým kódem. Použití uživatelských formulářů představuje výrazné rozšíření možností Excelu.

Text knihy doplňují poslední tři kapitoly: Kapitola 18 popisuje propojení Excelu s dalšími produkty MS Office (Word, Access a Outlook). Kapitola 19 je věnována úpravě sešitu na samostatnou aplikaci, kterou mohou používat jiní uživatelé a která obsahuje vlastní ovládací prvky, a to buď ve formě sešitu, nebo vlastního doplňku. Poslední kapitola je věnovaná třídám a představuje určitý úvod do programování na vyšší úrovni. I u tohoto tématu však jistě najdete zajímavá praktická využití.

Domníváme se, že tato osnova knihy pokryje převážnou většinu toho, co budete od programování v Excelu očekávat. Důležitou součástí knihy jsou praktické příklady. Ke každé kapitole (s výjimkou první) náleží sešit s ukázkovými makry, funkcemi a formuláři. Tyto sešity jsou ve formátu Excel 2019 (XLSM). V několika případech se jedná o samostatnou složku s více soubory. Odkazy na jednotlivé příklady naleznete v jednotlivých kapitolách. Vlastní kód VBA je v sešitech podrobně okomentován, aby byl snadno srozumitelný a mohl sloužit jako zdroj inspirace

při vytváření vlastního kódu. Příklady si můžete stáhnout z webových stránek nakladatelství Grada (www.grada.cz) v příslušné sekci této knihy.

V knize jsou použity následující typografické prvky:

- Příkazy jazyka VBA jsou odlišeny *neproporcionálním písmem*.
- Položky v nabídkách Excelu a editoru jazyka VBA, názvy karet, souborů a složek jsou zvýrazněny **tučným písmem**.
- Klávesové zkratky jsou vysazeny KAPITÁLKAMI.
- Názvy souborů a internetové adresy jsou označeny *kurzivou*.
- Soubory s příklady k jednotlivým kapitolám jsou zvýrazněny **tučnou kurzivou**.

Ať se vám při práci s Excelem a jazykem VBA daří!

Autoři

1 Tvorba maker záznamem

Co se v této kapitole naučíte:

- zaznamenání makra,
- absolutní a relativní odkaz v makru,
- různé způsoby spuštění maker,
- využití maker z jiného sešitu.

1.1 Použití maker v Excelu

Při rutinní práci s Excelem se často stává, že je zapotřebí několikrát za sebou opakovat série několika operací. Sešit Excelu je schopen zaznamenat řadu akcí po sobě, uložit je pod zvoleným názvem a podle potřeby tyto akce kdykoliv zopakovat. Takový záznam se označuje jako makro. Použití maker vám při práci s Excelem ušetří spoustu práce a zabrání zbytečným chybám.

Excel (podobně jako další aplikace z balíku MS Office) je vybaven programovacím jazykem Visual Basic for Application (VBA) a při záznamu makra se jednotlivé akce zaznamenají právě pomocí příkazů zapsaných v tomto jazyku. Editor jazyka VBA je součástí instalace kompletu MS Office. Z toho plyne, že makro můžete vytvořit dvojím způsobem:

- Uživatelsky: v sešitu zapnete zvláštní režim, který všechny provedené akce pomocí kódu v jazyku VBA zaznamená a tento kód uloží jako součást sešitu.
- Programátorsky: zobrazíte si editor jazyka VBA, ve kterém zapíšete potřebný kód.

Je možné používat i kombinaci obou způsobů – zaznamenat makro uživatelsky a později jej pozměnit úpravou vytvořeného kódu.

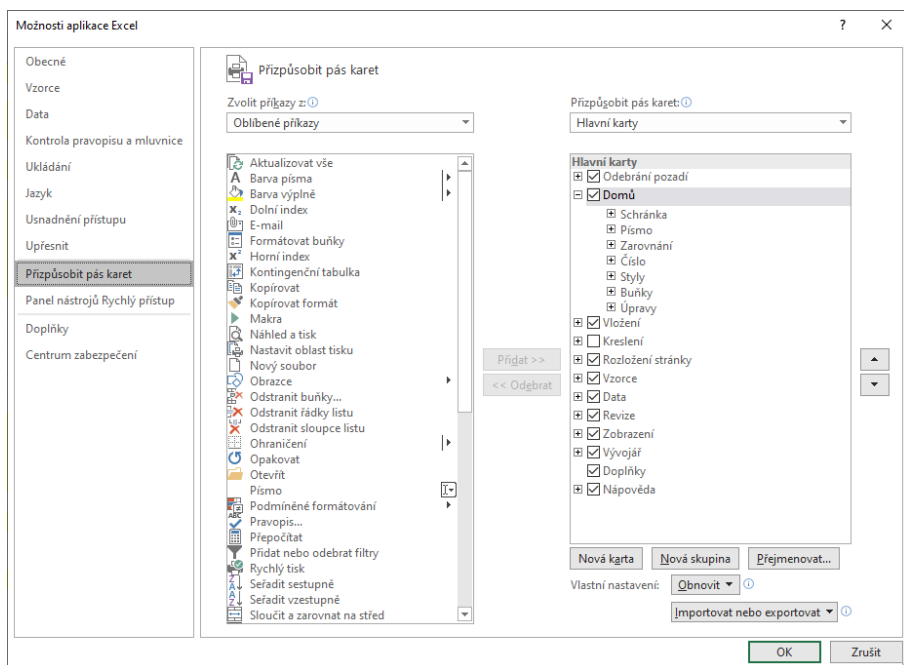
Při uživatelském přístupu může makro obsahovat pouze akce, které jste schopni provádět běžným způsobem. Programátorský přístup poskytuje další možnosti:

- využívat ke zpracování údajů v sešitu způsob, který při běžné práci nelze použít,
- pracovat se složkami a soubory,
- vytvářet nové vlastní funkce,
- spouštět automatické akce při otevření a zavření sešitu,
- vytvářet vlastní dialogová okna pro zadávání vstupních údajů.

V této kapitole se seznámíte s uživatelským způsobem tvorby pomocí záznamu maker. Tvorba kódu v jazyku VBA je předmětem dalších kapitol.

Pro práci s makry je potřebná karta **Vývojář**. Pokud karta není zobrazena, zobrazíte ji takto:

1. Klepnete na kartu **Soubor**.
2. Vpravo dole klepnete na položku **Možnosti**. Zobrazíte tím okno pro nastavení programu.
3. V levé části okna klepnete na položku **Přizpůsobit pás karet**.
4. V pravém seznamu označíte volbu **Vývojář** a potvrdíte tlačítkem **OK**.



Obrázek 1.1 Zobrazení karty Vývojář

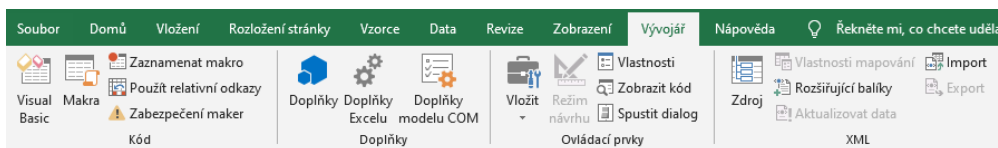
Toto nastavení Excelu je trvalé a při příštím spuštění již bude karta **Vývojář** zobrazena.



Jazyk VBA se v jednotlivých verzích Excelu neliší. Proto makro, které bylo zaznamenáno v nižší verzi Excelu, bude bez problémů fungovat i ve verzích vyšších. Nemusí to však platit obráceně, a to ani u některých akcí, které je možné používat v nižší i vyšší verzi Excelu. Některé funkce Excelu mají ve vyšších verzích více možností, kód VBA je u vyšší verze vytvořen jiným způsobem a ve verzi nižší nebude fungovat. Jedná se například o řazení tabulky a podmíněné formátování.

1.2 Záznam a spuštění makra

Pro práci s makry jsou určena tlačítka, která naleznete na kartě **Vývojář** vlevo ve skupině **Kód**. Na kartě **Zobrazit** je rovněž k dispozici tlačítko **Makra**, které umožňuje záznam a spuštění makra, práce s kartou **Vývojář** je však přehlednější.



Obrázek 1.2 Karta Vývojář

Pro záznam makra použijete tento postup:

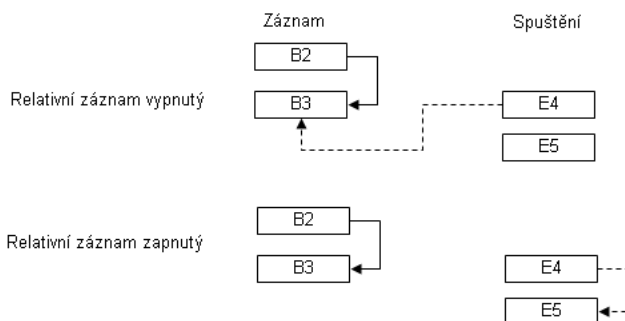
1. Přejdete na kartu **Vývojář**.
2. Pomocí tlačítka **Relativní odkazy** určíte, jak se má v makru zaznamenat výběr buněk, tedy pohyb kurzoru. Jestliže toto tlačítko stisknete, pohyb kurzoru se zaznamená relativně vůči

výchozí buňce. Pokud jej ponecháte vypnuté, zaznamená se pevná adresa buňky, na kterou se kurzor během záznamu přemístil.

3. Klepnete na tlačítko **Záznam makra**. Nápis na tlačítku se změní na **Zastavit záznam**.
4. V zobrazeném okně vyplníte textové pole **Název makra**. V názvu používejte písmena, číslíce nebo podtržítka. Jiné znaky (mezera, čárka, pomlčka atd.) nejsou přípustné.
5. V seznamu **Uložit makro do** vyberete, kam se má vytvářené makro zapsat (viz dále).
6. Potvrdíte tlačítkem **OK**. Tím se spustí záznam makra.
7. Provedete jednotlivé kroky, které má makro obsahovat.
8. Klepnete na tlačítko **Zastavit záznam**. Tím se záznam makra ukončí a obnoví se normální režim práce se sešitem.

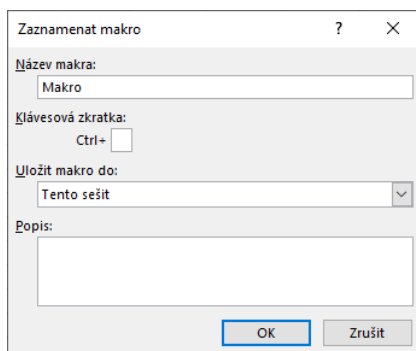


Název makra nesmí kolidovat s adresou buňky, například vytvořit makro s názvem **AA1** není možné. U názvů buněk, oblastí, listů a standardních funkcí Excelu tato kolize nenastává, ale toto pojmenování makra rozhodně nelze doporučit.



Obrázek 1.3 Vypnutý a zapnutý relativní záznam makra

Akce, které jsou zaznamenávány při tvorbě makra, zpravidla zahrnují i pohyb kurzoru a budou se používat u různých buněk. Při tvorbě většiny maker je proto nutné zapnout relativní záznam. Tlačítko **Relativní odkazy** je však po spuštění Excelu vypnuté a musíte jej stisknout před zahájením vlastního záznamu.



Obrázek 1.4 Okno pro záznam makra

Při záznamu makra se uloží všechny akce prováděné Excelem (tedy i chyby, kroky zpět a automatické uložení). Proto je vhodné před spuštěním záznamu sešit uložit.

Během vlastního záznamu většinou nezáleží na tom, jak určitou akci provedete. Můžete použít tlačítko na kartě, místní nabídku, nebo klávesovou zkratku. Výjimkou je nastavení písma: jestliže pro nastavení některého atributu písma použijete tlačítko nebo klávesovou zkratku, do makra se zaznamená pouze toto nastavení (například změna písma na tučné). Pokud ale nastavíte písmo pomocí dialogového okna, do makra se uloží všechna nastavení, tedy i font, velikost, barva atd. – bez ohledu na to, jestli se jedná o hodnoty výchozí, nebo upravené.

1.2.1 Uložení a vyzkoušení makra

Před spuštěním vlastního záznamu makra můžete v seznamu **Uložit makro do** vybrat tři možnosti, kam se bude vytvářené makro ukládat:

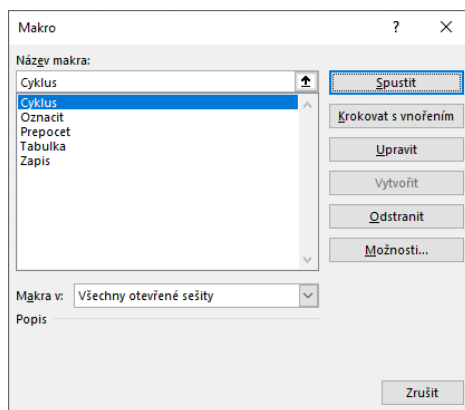
- Volba **Tento sešit** (výchozí): makro se ukládá do stejného sešitu, ve kterém byly zaznamenávané akce prováděny.
- Volba **Nový sešit**: založí se nový prázdný sešit, do kterého se makro uloží.
- Volba **Osobní sešit maker**: založí se speciální sešit s názvem *Personal.xlsb*, do kterého se vytvářené makro uloží. Pokud tento sešit již existuje, vytvářené makro se do sešitu přidá.

Sešit *Personal.xlsb* se při každém dalším spuštění Excelu automaticky otevře jako skrytý. Vede to k tomu, že všechna makra uložená v tomto sešitu jsou po spuštění Excelu vždy přístupná.



Uložení makra v sešitu *Personal.xlsb* (a tedy jeho zpřístupnění ve všech otevřených sešitech) může vést k nechtěnému spuštění makra. Pokud vytvářené makro uložíte do tohoto sešitu a nechcete, aby bylo všude přístupné, jedinou možností je sešit *Personal.xlsb* vymazat.

Pro vyzkoušení makra klepnete na tlačítko **Makra** na kartě **Vývojář**, v zobrazeném okně označíte vytvořené makro a klepnete na tlačítko **Spustit**. Po ukončení běhu makra je tlačítko **Zpět** (umístěné na panelu **Rychlý přístup**) znepřístupněno. To znamená, že akce provedené makrem jsou vždy NEVRATNÉ!



Obrázek 1.5 Okno se seznamem maker

1.2.2 Uložení a otevření sešitu s makrem

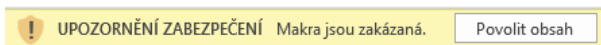
Současné verze Excelu (konkrétně Excel 2007 a výše) rozlišují dva typy sešitů: obyčejné sešity (přípona XLSX) a sešity s podporou maker (přípona XLSM). Sešit se zaznamenaným makrem

proto musíte uložit ve formátu XLSM (v dialogovém okně pro uložení sešitu použijete seznam **Uložit jako typ**). Při pokusu o uložení sešitu s makrem běžným způsobem se zobrazí varovné hlášení. Platí to pro nově vytvořený sešit i pro sešit s příponou XLSX, ve kterém chcete dodatečně vytvořit makro.



Jestliže se pokusíte uložit sešit se zaznamenaným makrem ve formátu XLSX a přes varovné hlášení v ukládání pokračujete, sešit se uloží, avšak zaznamenané makro se ze sešitu odstraní.

Při otevření sešitu s makrem se nad řádkem vzorců zobrazí výstražné hlášení s tlačítkem **Povolit obsah**. Po klepnutí na tlačítko se sešit otevře. Toto povolení se ukládá spolu se sešitem, takže při jeho dalším otevření se již hlášení nezobrazí a sešit se ihned otevře. Jestliže však sešit přejmenujete nebo jej přesunete do jiného umístění, výstražné hlášení se objeví znovu.



Obrázek 1.6 Povolení maker v sešitu

1.2.3 Odstranění makra ze sešitu

Nepotřebná nebo chybně pracující makra je záhodno ze sešitu odstranit, aby nedošlo k jejich nechtěnému spuštění. Klepnutím na tlačítko **Makra** zobrazíte seznam maker, v seznamu příslušné makro označíte a použijete tlačítko **Odstranit**. Akce je jistěna zpětným dotazem a je nevratná.

V dialogovém okně pro seznam maker je k dispozici také tlačítko **Upravit**. Toto tlačítko však neumožní opakovat záznam makra, ale zobrazí jeho kód zapsaný v jazyce Visual Basic. Jestliže tedy potřebujete makro upravit, ale nechcete pracovat přímo s jeho kódem, nezbude vám nic jiného než makro odstranit a zaznamenat jej znovu.

1.3 Různé způsoby spuštění makra

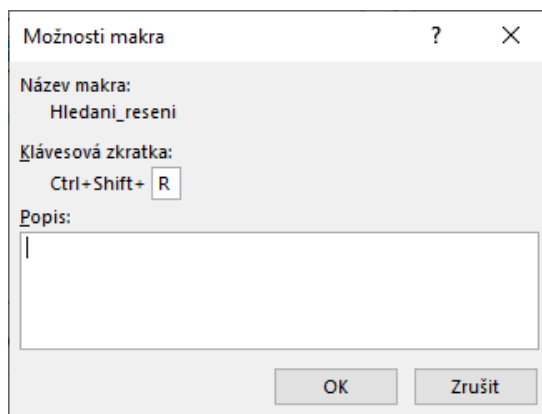
Spuštění makra pomocí seznamu maker se dobře hodí pro jeho testování, ale není příliš pohodlné pro rutinní práci. Excel proto nabízí několik dalších způsobů, jak makra spouštět:

- klávesovou zkratku,
- tlačítko na panelu **Rychlý přístup**,
- tlačítko na některé z ovládacích karet,
- tlačítko na listu.

1.3.1 Spuštění makra klávesovou zkratkou

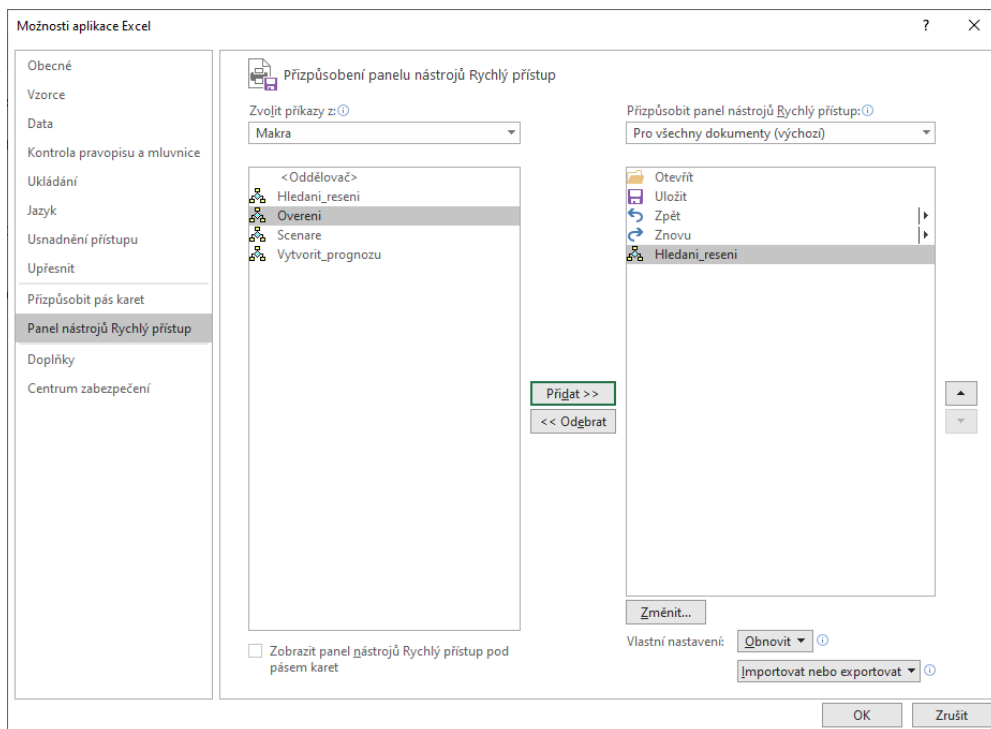
Přiřazení klávesové zkratky makru je možné už při záznamu makra, nebo dodatečně:

1. Tlačítkem **Makra** na kartě **Vývojář** zobrazíte seznam maker a v něm označíte potřebné makro.
2. Stisknete tlačítko **Možnosti**.
3. Do položky **Klávesová zkratka** zapíšete písmeno, které spolu s klávesou **CTRL** bude tvořit klávesovou zkratku.
4. Potvrdíte tlačítkem **OK**.



Obrázek 1.7 Přiřazení klávesové zkratky makru

V klávesových zkratkách se rozlišují malá a velká písmena. Jestliže do položky **Klávesová zkratka** zapíšete malé písmeno, klávesovou zkratkou bude klávesa CTRL spolu s písmenem. Pokud zapíšete velké písmeno, klávesovou zkratkou bude kombinace CTRL+SHIFT+PÍSMENO.



Obrázek 1.8 Úprava panelu Rychlý přístup

Pro klávesovou zkratku je třeba zvolit písmeno, které není použito ve standardních klávesových zkratkách Excelu. Pokud to provedete, klávesová zkratka se předefinuje, tedy neprovede standardní akci Excelu, ale spustí makro. Toto je ovšem problém, protože standardní klávesové zkratky již vyčerpaly prakticky celou abecedu. Klávesová zkratka CTRL+Q, která spolehlivě fun-

govala v dřívějších verzích Excelu, není od verze 2013 již použitelná, protože je rezervována pro rychlou analýzu dat. Řešením je použití velkých písmen, protože standardní klávesové zkratky Excelu používají vesměs malá písmena.

V seznamu maker nejsou přiřazené klávesové zkratky zobrazeny, pro jejich zjištění musíte použít tlačítko **Možnosti**.



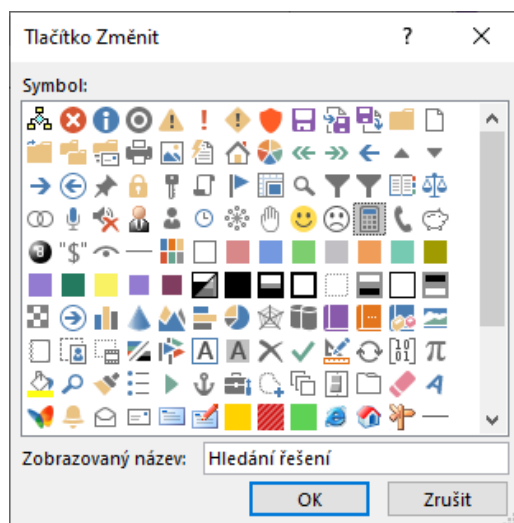
Jestliže se pokusíte přiřadit tutéž klávesovou zkratku dvěma různým makrům v jednom sešitu, Excel nezobrazí žádné upozornění a duplicitní přiřazení provede. Při jejím použití se pak spustí makro, jehož název je v abecedě „menší“ (nikoliv makro, které bylo vytvořeno dříve).

1.3.2 Tlačítko na panelu Rychlý přístup

Přidání tlačítka pro spouštění makra na panel **Rychlý přístup** provedete následujícím způsobem:

1. Klepnete na kartu **Soubor** a vpravo dole použijete položku **Možnosti**.
2. V levé části zobrazeného okna klepnete na položku **Panel nástrojů Rychlý přístup**.
3. V seznamu **Zvolit příkazy** nastavíte položku **Makra**.
4. Označíte makro a klepnete na tlačítko **Přidat**.
5. V rozevíracím seznamu v pravé horní části okna **Přizpůsobit panel nástrojů Rychlý přístup** nastavíte, kde bude vytvořené tlačítko zobrazeno.
6. Potvrdíte tlačítkem **OK**.

U vytvářeného tlačítka můžete také změnit ikonu a upravit text bublinové nápovědy, která se u tlačítka zobrazí. Před uzavřením okna pro úpravu panelu **Rychlý přístup** označíte v pravém seznamu přidané makro a klepnete na tlačítko **Změnit**. V dalším zobrazeném okně vyberete vhodnou ikonu, podle potřeby upravíte zobrazovaný text a potvrdíte tlačítkem **OK**.



Obrázek 1.9 Změna ikony a textu u tlačítka

V rozevíracím seznamu v pravé horní části okna **Přizpůsobit panel nástrojů Rychlý přístup** je možné použít dvě volby:

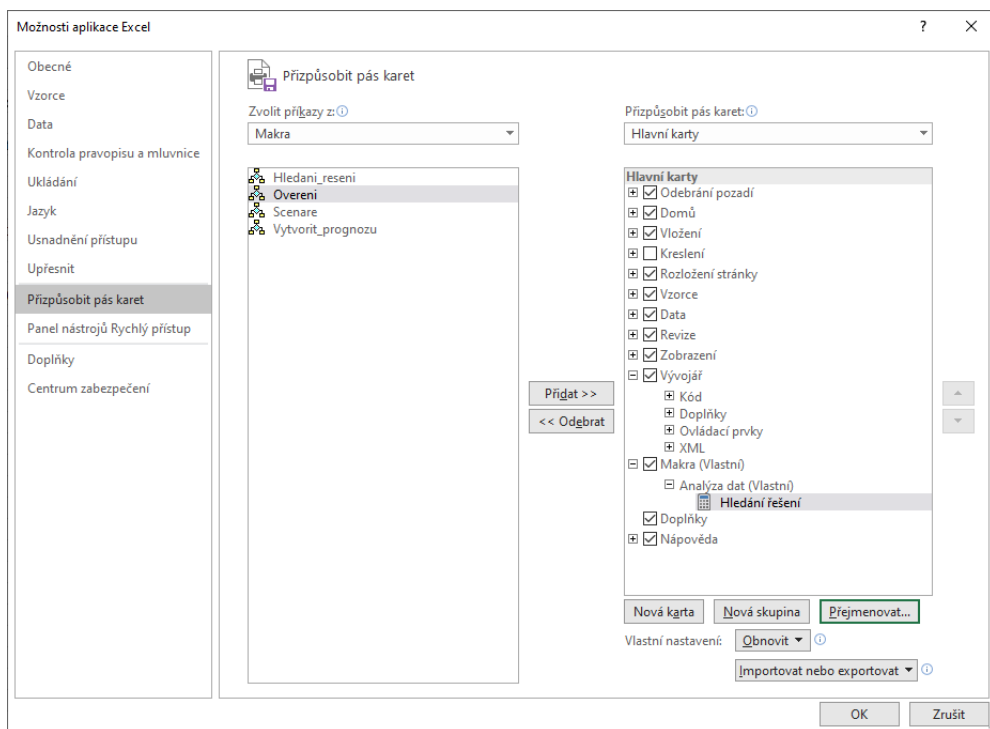
- Jako výchozí je nastavena volba **Pro všechny dokumenty (výchozí)**. Při této volbě se tlačítko na panelu **Rychlý přístup** zobrazí vždy, i když sešit s makrem není otevřen. V takovémto případě se sešit s makrem automaticky otevře a makro se spustí (pokud sešit s makrem nebyl přejmenován nebo přesunut do jiného umístění).
- Volba **Pro dokument** + název otevřeného souboru způsobí, že se přidané tlačítko zobrazí jen v souboru s vytvořeným makrem.

Podle toho, jakou možnost zvolíte, se v dlouhém seznamu vpravo zobrazí buď tlačítka dostupná všude, nebo tlačítka dostupná jen v otevřeném sešitu. Totéž se uplatní i u odstranění tlačítka z panelu **Rychlý přístup**, což provedete tlačítkem **Odebrat**. V rozevíracím seznamu vpravo nahoře je třeba nastavit odpovídající volbu.

1.3.3 Přidání tlačítka na kartu

Tlačítko pro spuštění makra můžete také přidat na některou z karet, nebo si vytvořit kartu vlastní. Postupujte takto:

1. Klepnete na kartu **Soubor** a vpravo dole použijete položku **Možnosti**.
2. V levé části zobrazeného okna klepnete na položku **Přizpůsobit pás karet**.
3. Klepnete na tlačítko **Nová karta**. Tím se v pravém seznamu vytvoří nová karta a v ní nová skupina.
4. Označíte postupně novou kartu a novou skupinu a pomocí tlačítka **Přejmenovat** zadáte vhodné názvy.
5. Označíte novou skupinu. V rozevíracím seznamu **Zvolit příkazy** nastavíte volbu **Makra** a tlačítkem **Přidat** zařadíte do skupiny nové tlačítko.



Obrázek 1.10 Přidání tlačítka na kartu

6. Vytvořené tlačítko označíte a klepnete na tlačítko **Přejmenovat**. Vyberete ikonu a změníte název tlačítka.
7. Polohu nové karty můžete upravit pomocí tlačítek se šipkami nahoru a dolů.
8. Potvrdíte tlačítkem **OK**.

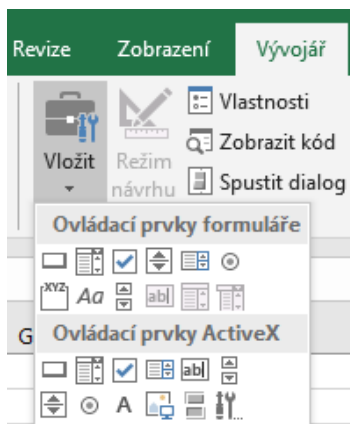
Jestliže označíte některou ze standardních karet a použijete tlačítko **Nová skupina**, do karty se zařadí nová prázdná skupina. Do této skupiny je možné přidávat popsáním způsobem nová tlačítka. U standardních skupin to možné není.

Při přidávání tlačítek do některého pásu karet není možné zvolit umístění vytvořeného tlačítka. Tlačítko se vždy zobrazí v jakémkoliv souboru. Při klepnutí na tlačítko je reakce stejná jako u tlačítka na panelu **Rychlý přístup**, tedy sešit s makrem nemusí být otevřený a při použití tlačítka se automaticky otevře.

1.3.4 Přidání tlačítka na list

Tlačítko pro spouštění makra můžete také vložit na některý z listů. Postup je následující:

1. Na kartě **Vývojář** klepnete na šipku u tlačítka **Vložit**. Zobrazí se nabídka ovládacích prvků, které je možné na listu vytvářet.
2. V nabídce klepnete na tlačítko umístěné v první řadě zcela vlevo (bublina „Tlačítko (ovládací prvek formuláře)“).
3. V listu klepnete na místo, kde má být tlačítko umístěno.
4. Po klepnutí se zobrazí okno se seznamem maker. Označíte potřebné makro a klepnete na tlačítko **OK**.



Obrázek 1.11 Nabídka pro vytvoření prvku na listu

Na vytvořené tlačítko klepnete pravým tlačítkem myši, z místní nabídky použijete položku **Upravit text** a přepíšete nápis na tlačítku. Jestliže na tlačítko klepnete pravým tlačítkem myši a poté použijete klávesu Esc, tlačítko zůstane ohraničeno úchyty (obdobně, jako je označen obrázek). U označeného tlačítka je možné měnit jeho velikost tahem za úchyty, měnit jeho polohu (myši nebo klávesovými šipkami) a na kartě **Domů** nastavit tučné písmo nebo kurzívu. Tlačítko pro nastavení barvy písma je nedostupné. Můžete však na tlačítko klepnout pravým tlačítkem myši, z místní nabídky vybrat příkaz **Formát ovládacího prvku** a nastavit potřebnou změnu na kartě **Písmo**. Klávesou DELETE se označené tlačítko z listu odstraní.

1.4 Spuštění makra v jiném sešitu

Makro, které je vytvořené v nějakém sešitu, je možné spustit v libovolném jiném sešitu. Podmínkou je, aby byly oba sešity otevřené. Makro lze tímto způsobem využít i v sešitu bez podpory maker (přípona XLSX). Ke spuštění můžete použít seznam maker (před názvem makra je v tomto případě zobrazen ještě název sešitu, oddělený vykřičníkem), klávesovou zkratku, tlačítko na panelu **Rychlý přístup** nebo tlačítko na některé z karet. Tímto způsobem také funguje zmíněný sešit *Personal.xslb*, který je otevřen automaticky jako skrytý.

Pokud sešit s makrem otevřen není, seznam maker se zobrazí jako prázdný a nefunguje ani klávesová zkratka. Tlačítko na panelu **Rychlý přístup** a tlačítko na některé z karet, jež je určeno ke spuštění makra, automaticky sešit s makrem otevřou a makro spustí. Sešit s makrem však nesmí být přejmenován nebo přesunut do jiného umístění.

2 Základy jazyka VBA

Co se v této kapitole naučíte:

- práci s editorem VBA,
- vytvoření makra a uživatelské funkce, jejich využití,
- přidání nápovědy k vytvořené funkci,
- práci s proměnnými a tvorbu vzorců,
- větvení výpočtu na několik částí,
- použití cyklů.

Příklady k této kapitole naleznete v sešitu **Zaklady_VBA.xlsm**.

2.1 Práce s editorem VBA

Záznamník maker vytváří kód v programovacím jazyku Visual Basic for Application (VBA). V tomto jazyku můžete vytvářet také vlastní kódy tří typů:

- Procedury (Sub), které použijete stejně jako makra vytvořené záznamem.
- Uživatelské funkce (Function), které lze vkládat do buněk jako vzorce.
- Vlastnosti (Property), které se používají při tvorbě vlastních tříd.

V této a dalších kapitolách se seznámíte s tvorbou maker a funkcí. Práce s třídami je popsána v poslední kapitole knihy.

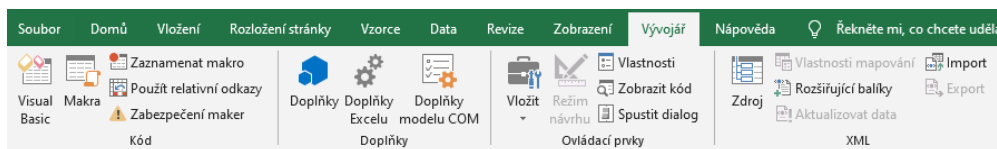
Do okna editoru VBA přejdete stisknutím tlačítka **Visual Basic**, které je umístěno na kartě **Vývojář** ve skupině **Kód**. Okno editoru VBA vytvoří na hlavním panelu samostatné tlačítko.¹ V okně editoru naleznete tři podokna:

- Podokno projektu (vlevo nahoře) obsahuje seznam listů v sešitu, položku **ThisWorkbook** (což představuje celý sešit), složku **Modules**, popřípadě složku **Forms** s uživatelskými formuláři. Jestliže chcete zobrazit kód některého z modulů, poklepejte na jeho název.
- Podokno vlastností **Properties** (vlevo dole) obsahuje u modulu pouze jeho název (Name). Hlavní použití tohoto podokna je při tvorbě formulářů.
- Okno programového modulu (vpravo) slouží k vlastnímu zápisu kódu.

Jestliže v editoru VBA není zobrazeno levé podokno projektu nebo podokno vlastností, zobrazíte je pomocí položek **View | Project Explorer** nebo **View | Properties Window**.

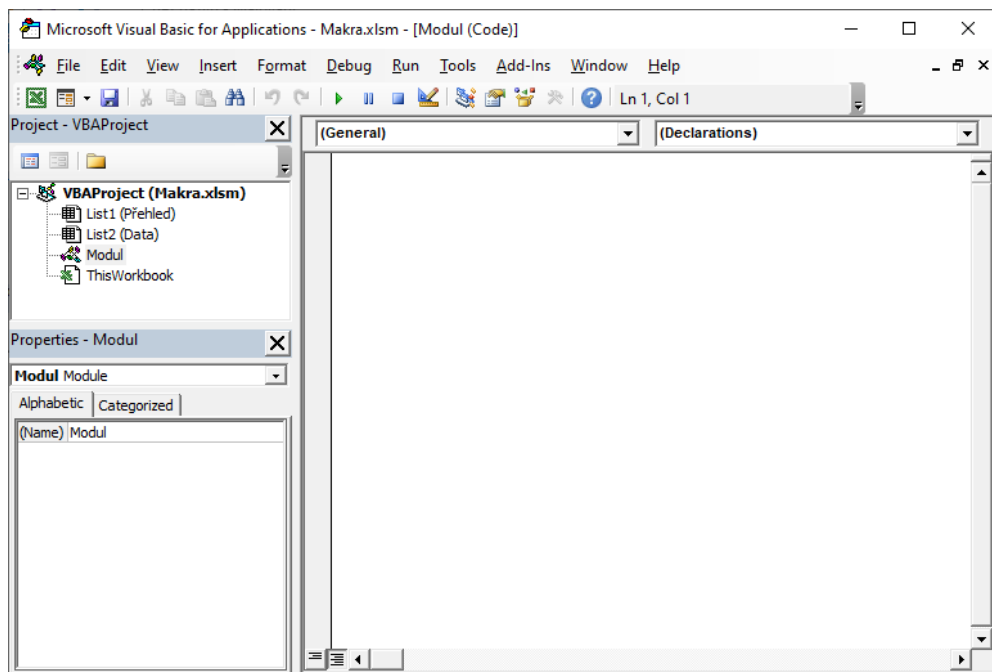
Aby bylo možné v sešitu vytvořit nějaký kód, musíte nejprve do sešitu vložit programový modul. Při tvorbě makra pomocí záznamu se programový modul vloží do sešitu automaticky, při tvorbě kódu zápisem jej musíte vložit sami s použitím následujícího postupu:

¹ Okno Excelu i okno editoru VBA náleží k jednomu a témuž souboru. Proto uložením sešitu uložíte i případné změny v kódu a naopak.



Obrázek 2.1 Karta Vývojář

1. Klepnete na záložku karty **Vývojář** a na této kartě stisknete tlačítko **Visual Basic**. Tím zobrazíte okno editoru VBA.
2. V hlavní nabídce tohoto okna použijete položku **Insert | Module**.
3. Klepnete do podokna **Properties** a vlastnost **Name** přepíšete na vhodné pojmenování (výchozí je **Module1**).
4. Klepnete vpravo do podokna modulu. Tím je editor kódu připraven k práci.



Obrázek 2.2 Okno editoru VBA

Při zápisu kódu postupujete obdobně jako při práci s poznámkovým blokem. Jednotlivé příkazy ukončíte klávesou ENTER. Prázdné řádky a odsazení od levého okraje (mezery, tabulátor) jsou ignorovány. Klávesové zkratky CTRL+C, CTRL+X a CTRL+V můžete použít ke zkopírování, vyjmutí a vložení označeného textu.



Kód v jazyku VBA často obsahuje do sebe vnořené příkazy. Vnořené příkazy je výhodné odsadit více doprava pomocí mezer nebo tabulátoru. Jednotlivé části kódu můžete od sebe oddělit prázdným řádkem. Obojí usnadňuje orientaci ve vytvořeném kódu.

Při rozdělení příkazu na dva řádky ukončíte první část mezerou a podtržítkem, další řádek je pak chápán jako pokračování. Na druhé straně je možné na jeden řádek napsat i více příkazů a oddělit je dvojtečkou. Tedy řádek:

```
<příkaz1>: <příkaz2>
```

funguje stejně, jako dva řádky:

```
<příkaz1>  
<příkaz2>
```

Řádky s apostrofem jsou komentáře a při provádění kódu se ignorují. Komentáře začínající apostrofem je také možné připsat za výkonný příkaz.

Editor VBA zobrazuje zapsaný kód pomocí čtyř barev:

- modrou barvou klíčová slova jazyku Visual Basic,
- černou barvou názvy maker, funkcí, proměnných a odkazy na prvky v sešitu (buňky, listy atd.),
- zelenou barvou komentáře,
- červenou barvou chybné řádky.

```
Public Sub Moje_makro()  
' Tato řádka je komentářem  
cena = 10 ' Komentář může být i za příkazem  
Tato řádka je chybná  
End Sub
```

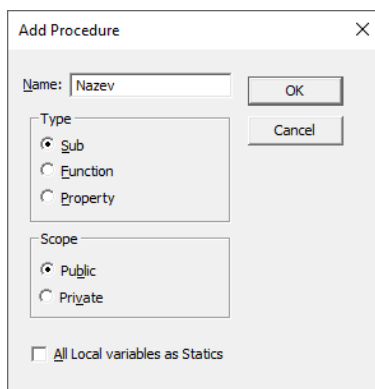
Obrázek 2.3 Barvy v kódu VBA

Při zápisu kódu je třeba pamatovat na to, že editor VBA není počestěn. Musíte tomu tedy přizpůsobit zápis kódu:

- jako oddělovač desetinných míst používejte tečku (nikoliv čárku),
- jako oddělovač v seznamu používejte čárku (nikoliv středník).

Tato pravidla platí v anglické i české verzi Excelu.

Pro zobrazení okna nápovědy použijete položku **Help | Microsoft Visual Basic for Applications Help**. Jestliže chcete zobrazit konkrétní nápovědu pro některý z příkazů jazyka VBA, označíte jej jako blok a stisknete klávesu F1.



Obrázek 2.4 Vložení kódu do modulu

2.2 Tvorba makra kódem VBA

Vložení kódu do modulu provedete těmito kroky:

1. V okně editoru VBA klepnete do podokna modulu.
2. Použijete položku **Insert | Procedure**.
3. V zobrazeném okně označíte typ vkládaného kódu (**Sub** – makro).
4. Zadáte název makra (**Name**).
5. Potvrdíte tlačítkem **OK**.

Tím se do modulu vloží první a poslední řádek kódu makra:

```
Public Sub Nazev_makra()
```

```
End Sub
```

V jednom modulu lze vytvořit více maker nebo funkcí, jejich kód je oddělen vodorovnou čarou.

Mezi první a poslední řádek zapíšete vlastní výkonné příkazy. Při psaní kódu je vhodné dodržovat následující pravidla:

- Všechny příkazy pište malými písmeny. Pokud je příkaz zapsán správně, po opuštění zapsaného řádku se počáteční písmena klíčových slov jazyku VBA změní na velká.
- Při tvorbě vnořených příkazů zapíšete nejprve začátek a konec a teprve potom vnořené řádky. Jedna z nejčastějších chyb totiž je, že při psaní vnořených příkazů zapomenete na konec.
- Do kódu si hned zapisujete komentáře. Využívejte přitom možnosti psát komentář za příkaz do jednoho řádku.
- Pokud chcete některý příkaz vyřadit, je mnohdy výhodnější jej nemazat, ale zapsat před něj apostrof. Tím se z něho stane komentář. Odstraněním apostrofu se příkaz znovu aktivuje.



Pro všechny názvy v editoru VBA (moduly, makra, funkce, proměnné atd.) používejte pouze písmena bez české diakritiky, číslice nebo podtržítka. Nepoužívejte mezery a další interpunkční znaky.

Pro názvy zapisovaných maker platí tato pravidla:

- Při tvorbě názvů se nerozlišují malá a velká písmena.
- Název může obsahovat českou diakritiku. Protože však editor VBA není počestěn, bude lepší písmena s háčky a čárkami nepoužívat.
- Název makra lze vytvořit jako adresu buňky (například **AA1**), ale makro nebude možné spustit.
- Název makra musí být v rámci modulu jedinečný. V jednom modulu nelze vytvořit makro i funkci se stejným názvem.
- Je možné použít stejný název makra ve dvou různých modulech, těmto duplicitám je však lepší se vyhnout.

Makro zpravidla pracuje s buňkami: v následující kapitole se dozvíte, jak se na buňky odvolat. Jako příklad jednoduchého makra zatím postačí zpráva na obrazovku, která má tvar:

```
MsgBox "Text zprávy"
```

Příkaz `MsgBox` zobrazí okno s textem zprávy a tlačítkem **OK**. Pomocí tohoto příkazu je možné zobrazovat i hodnotu nějaké proměnné:

```
promenna = 10  
MsgBox promenna
```