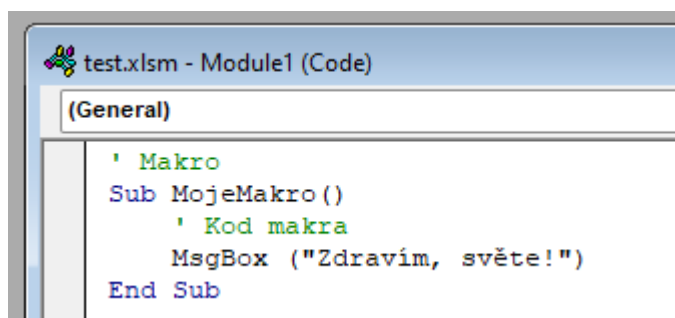


1.2 Lekce 1 – Procedury

Při používání programu Excel může jeho uživatel rozšířit funkčnost svého sešitu nebo listu pomocí dvou hlavních procedur, do kterých může zapsat nebo nahrát kód. Jedná se o procedury makro a funkce. Rozdíly mezi nimi jsou vysvětleny v následujících podkapitolách.

1.2.1 Makro

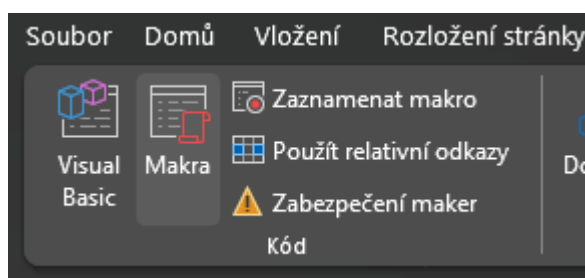
Makrem je označována každá procedura, která je definována v kódu pomocí klíčového slova *Sub* (vychází z anglického názvu *subroutine* – v češtině subrutina, podprogram nebo makro). Makro představuje sadu instrukcí, které se počítač při spuštění daného makra pokusí splnit. K úspěšnému dosažení konce makra, označeného pomocí klíčového slova *End Sub*, musí být všechny instrukce správně syntakticky zapsány, jinak dojde k chybě při běhu programu.



```
test.xlsm - Module1 (Code)
(General)
' Makro
Sub MojeMakro ()
    ' Kod makra
    MsgBox ("Zdravím, světe!")
End Sub
```

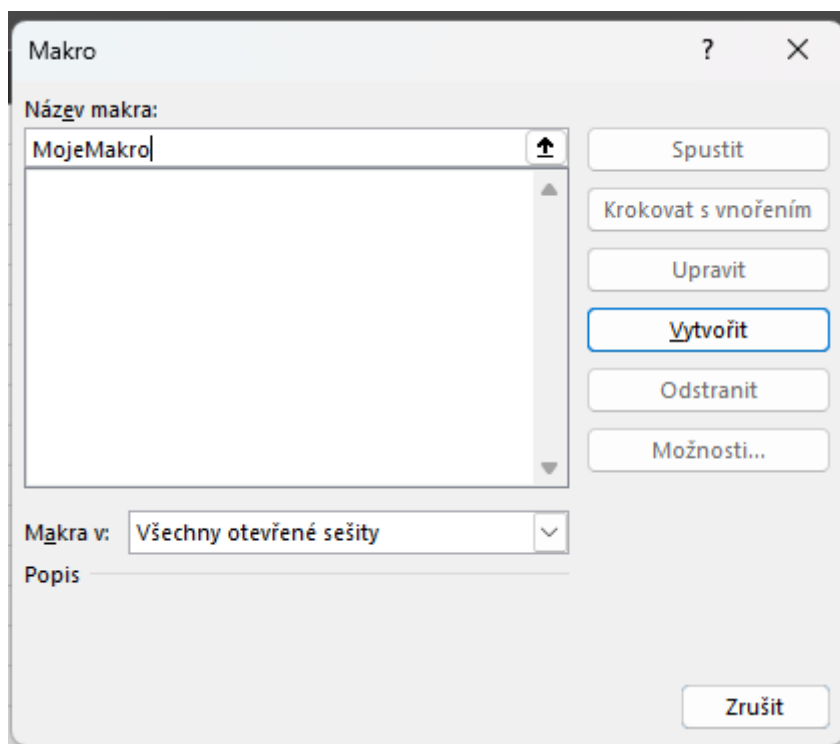
Obrázek 7 – Syntaktická ukázka kódu makra

Vytvořit makro lze několika způsoby. Jedním ze způsobů je pomocí karty *Vývojář* a tlačítka *Makra* ve skupině *Kód*.



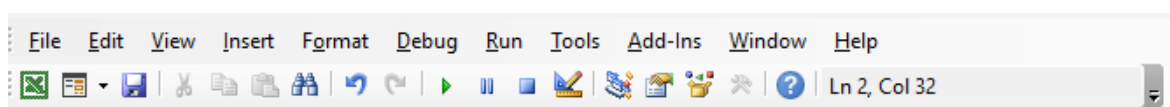
Obrázek 8 – Tlačítko Makra pro otevření podokna s organizací maker

Po kliknutí na tlačítko se zobrazí nové menší podokno, které zobrazuje všechna doposud vytvořená makra. Uživatel musí zapsat název svého makra bez mezer, čísel a diakritiky. Pokud název již neexistuje, mělo by se zpřístupnit tlačítko *Vytvořit*. Stiskem tohoto tlačítka se otevře Visual Basic Editor s vytvořenou kostrou makra.



Obrázek 9 – Podokno s organizací maker v sešitech

Po doplnění kódu, který by měla procedura splnit, lze makro spustit ve stejném podokně, akorát s označením daného makra a s pomocí tlačítka *Spustit*. Makro lze přímo spustit i z editoru pomocí zelené šipky v liště pod nabídkou menu, v menu pomocí *Run Sub/User Form* nebo pomocí klávesy *F5*. Pomocí lišty lze makro i pozastavit či kompletně zastavit.

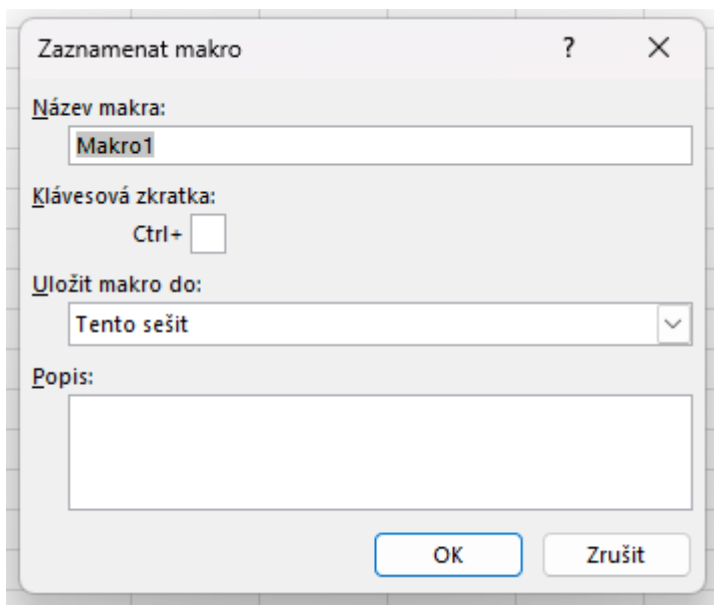


Obrázek 10 – Lišta s často používanými funkcemi, i pro ladění kódu

1.2.2 Záznam makra

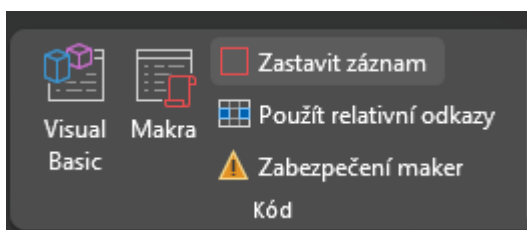
Pro urychlení a ulehčení vývojářům psaní kódu VBA existuje možnost zaznamenat kód přímo v aplikaci Excel. To znamená, že pokud vývojář stiskne nahrávání, veškeré jeho akce, které v programu Excel provede se zapíší do kódu VBA, dokud nahrávání neukončí.

Nahrávání makra se spouští na kartě *Vývojář* pomocí tlačítka *Zaznamenat makro* ve skupině *Kód*. Po kliknutí na tlačítko se zobrazí nové podokno, které vyžaduje zadat název nově zaznamenávaného makra, klávesovou zkratku pro rychlé spuštění, informaci o místě uložení makra a popis makra.



Obrázek 11 – Podokno se záznamem makra

Po potvrzení údajů pomocí tlačítka *OK* se spustí nahrávání makra. Každá změna v sešitě se nyní zaznamená do VBA jako instrukce v subrutině. Záznam se zastaví pomocí tlačítka *Zastavit záznam* na kartě *Vývojář*.



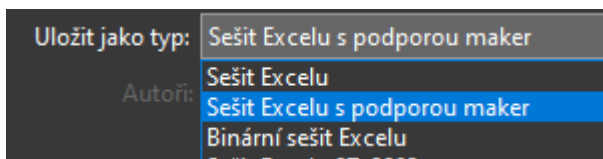
Obrázek 12 – Tlačítko zastavení záznamu makra

Po dokončení nahrávání lze ve VBE nahrané makro upravit dle potřeb vývojáře. Makro lze najít v podokně maker a upravit tlačítkem *Upravit*, které otevře VBE na místě, kde se zaznamenané makro nachází.

V rámci nahrávání makra může uživatel vybrat možnost *Použít relativní odkazy*. Při vybrání této možnosti se makro bude nahrávat s odsazením od aktuálně vybrané buňky, místo toho, aby si zapisovalo vybrané souřadnice buňky.

1.2.3 Ukládání sešitu

Ukládání souboru do normálního excel dokumentu *Sešit Excelu* (koncovka .xls) způsobí, že vytvořená makra se v souboru neuloží. K uložení dokumentu včetně maker je potřeba soubor uložit jako typ *Sešit Excelu s podporou maker* (.xlsm).



Obrázek 13 – Ukázka možnosti uložení sešitu s podporou maker

1.2.4 Funkce

Procedury typu funkce na rozdíl od maker vracejí tzv. návratovou hodnotu, jinými slovy nějaký finální výsledek, ke kterému během svého chodu dospěli. Tento výsledek může být výsledek výpočtu, hláška, datum atd.

Funkci lze zavolat v kódu a její výsledek uložit do proměnné nebo ji lze použít přímo v sešitě Excel. Návratová hodnota se v takovém případě objeví v buňce, ze které funkce byla volána.

Před použitím funkce je potřebné prvně danou funkci definovat. Definice funkce probíhá klíčovým slovem *Function*, které je následované názvem. Název funkce musí být v rámci dokumentu jedinečný, aby nedošlo ke kolizi funkcí. Konec funkce je obdobně jako u makra označen klíčovými slovy *End Function*. Před ukončením funkce se v určitém momentu měla uložit návratová hodnota. Tento proces proběhne přiřazením dané hodnoty názvu funkce pomocí operátoru rovná se. V případě nepřirazení funkce nebude vracet žádnou hodnotu.

```
test.xlsm - Module1 (Code)
(General)
' Funkce
Function MojeFunkce()
'Kod funkce
MsgBox ("Zdravim, z funkce!")
'Přiřazení návratové hodnoty funkce
MojeFunkce = "Pozdrav"
End Function
```

Obrázek 14 - Syntaktická ukázka kódu funkce

V rámci kódu je možné funkci opustit příkazem *Exit Function*, který ukončí kód v místě použití. Klíčové slovo *End* tedy označuje pouze konec definice subrutiny, zatímco *Exit* ji opouští s tím, že nebude pokračovat ve vykonávání dalšího kódu.

Předtím zmíněné definice funkce by neustále vracely stejnou hodnotu a jejich užitek by byl poměrně malý. Z tohoto důvodu lze funkcím přiřadit vstupní parametry, které ovlivňují dosažený výsledek funkce. Vstupní parametry funkce, lze nadefinovat ve funkci do závorek po uvedení názvu funkce. Množství vstupních parametrů není nijak omezeno.

```
' Funkce výpočet obsahu čtverce
Function ObsahCtverce(stranaA)
    'Kód funkce
    ObsahCtverce = stranaA * stranaA
End Function
```

Obrázek 15 – Ukázka funkce *ObsahCtverce* s parametrem *stranaA*

Pro příklad řekněme, že existuje funkce pojmenovaná *ObsahCtverce*. Jelikož nechceme po každé upravovat funkci pro výpočet obsahu čtverce s tím, že v ní změním číslo s velikostí strany, musí tato funkce obsahovat vstupní parametr. Ten by se měl měnit dle potřeb případu použití funkce. Pro výpočet obsahu čtverce je potřeba znát délku jeho strany, kterou lze označit například *stranaA*. Z toho vyplývá, že *stranaA* bude vstupním parametrem pro tuto funkci. Návrátová hodnota funkce bude obsahovat výpočet za použití tohoto vstupního parametru, v tomto případě $stranaA * stranaA$.

Některé parametry funkcí mohou být volitelné (uživatel funkce je nemusí zadávat k tomu, aby funkce fungovala v pořádku). Ve VBA se takový parametr označuje klíčovým slovem *Optional*. Pokud je potřeba přiřadit tomuto parametru výchozí hodnotu, přiřadí se pomocí znaku rovná se.

```
'Nadefinována funkce s názvem "MojeFunkce"
Function MojeFunkce(Jmeno, Optional Prijmeni = "ve VBA")
    'Návratová hodnota funkce - Vrátí vložené jméno a příjmení, pokud bylo vloženo
    'Pokud Prijmeni nebylo vloženo, vrátí místo něj "ve VBA"
    MojeFunkce = Jmeno & " " & Prijmeni
End Function
```

Obrázek 16 – Ukázka funkce s volitelným parametrem s výchozí hodnotou

1.2.5 Úkoly k procvičení

- 1) Vytvořte makro *Pozdrav*, které zobrazí zprávu „Pozdrav z UTB“.
- 2) Nahrajte makro *NovyList*, které vytvoří nový list v sešitě.
- 3) Vytvořte funkci *MojeJmeno*, která po použití zobrazí v dané buňce Vaše jméno.
- 4) Vytvořte v dokumentu *osoby.xlsm* nový modul a novou funkci *CeleJmeno*, která bude mít vstupní parametry (povinný) *Jmeno*, (povinný) *Prijmeni*, (volitelný) *TitulPred* a (volitelný) *TitulZa*. Výstupem této funkce bude kompletní jméno včetně titulů, pokud jsou zadány. Doplňte pomocí funkce sloupec *Celé Jméno* v daném sešitě. Bonus: Odstraňte všechny mezery navíc před a za celými jmény.