

Čas od času se potkávám s požadavkem měnit jazyk popisků v buňkách listu. Co taková vícejazyčná faktura? Musíme vytvářet pro každý jazyk zvláštní list/sešit, nebo je možné připravit jednu vícejazyčnou formu listu? Toť otázka. Konkrétně u faktury je potřeba řešit nejen popisky, ale také měnu, daň, kurz, zvyklosti dané země a požadavky dané zákony. V tom případě je jednodušší zpracovat každou fakturu samostatně. My se ale nyní budeme věnovat technickému řešení méně komplikované úlohy. Více obrázků.



### Vícejazyčný popis v buňce listu

Slovník pojmů máme na listu Excelu v Tabulce (TableLanguage) a vyhledávání v ní probíhá prostřednictvím funkce SVYHLEDAT. Každý pojem má své vlastní číslo (klíč, index). Zde náhodou odpovídá obsahu, nicméně si vymyslete vlastní smysluplné číslování (není nutná posloupnost, naopak, počítejte se zařazením chybějícího pojmu a následného setřídění seznamu filtrem v Tabulce). Podle klíče a vybraného jazyka (IndexLanguage, třetí parametr v SVYHLEDAT) vracíme odpovídající překlad. Nic víc, nic míň.

Je zřejmé, že řešení není všeobjímající. Vzorce mohou ovlivňovat pouze buňky, ze kterých jsou volány. Máme tu ještě přinejmenším názvy listů (ouška), název celého sešitu, popisky ovládacích prvků, případně názvy složek... A ačkoliv nám v něčem pomohou makra, stejně dospějeme do stavu, kdy budeme muset vypustit a říct dost.

Změna názvu listu se automaticky promítne do přímých (netextových) odkazů ve vzorcích, definovaných názvů apod. Přesto bych to nedělal a držel se „primárního“ jazyka uživatelů (nejlépe angličtina). Od přejmenování složek ruce pryč...

Tip: Jestliže se chcete ve VBA vyhnout problémům s odkazy na přejmenovaný list, nepoužívejte odkaz Worksheets(„List1“), ale kódový název listu – viz vlastnost Name daného listu v okně Properties (F4), kde doporučuji název upravit k obrazu svému (pro List1 je výchozí název List1). Jestliže jste jej pojmenovali jako wshListImport, pak v kódu VBA s ním pracujete jako wshListImport.Range(„A1“) oproti původnímu Worksheets(„List1“).Range(„A1“).



### Kódový název listu

Tip: Ouška listů je možné skrývat a poté řešit přechody s pomocí hypertextových odkazů (s grafickými symboly vhodného fontu) nebo s využitím vlastního menu.

Ačkoliv je to zřejmé, přesto připomínám, že se kupříkladu nemůžeme spoléhat ani na složku s českým názvem Dokumenty v systému. Jednou procedurou budeme dolovat údaje z registru, druhou zjišťovat nastavení z Application.International v Excelu a z Ovládacích panelů ve Windows, a ve třetí s pomocí API, WMI či WSH prstíkem škemrat o další informace...

Měnit jazyk až po spuštění Excelu na listu „Nastavení“ může být pozdě (výchozí by měla být zmíněná angličtina). Kam tedy s nastavením jazyka pro uživatele? Pokud máme k dispozici makra, pak není od věci vrátit se k historickým INI souborům (viz [článek](#)). Jedná se o prosté textové soubory, kde data jsou uspořádána do sekcí ve tvaru Proměnná = Hodnota. Čtení z těchto souborů pak obstarávají při otevírání sešitu (událost Workbook\_Open) API funkce (podporující Unicode).

Na formulářích je vhodné nastavit popisky ovládacích prvků v rámci události UserForm\_Initialize(), která nastává ještě před jejich vlastním zobrazením. A nejsou to jen popisky, co nás bude trápit, ale i formát datumu, desetinná čísla a další obsah ovládacích prvků a jejich vazba na list). Jestliže předpokládáme jazykové mutace včetně české, rozhodně neumísťujeme slovníček do kódu VBA. Vhodným místem je opět list Excelu, který na rozdíl od editoru VBA pracuje s Unicode. Čeština jednoduše řečeno musí pryč z editoru VBA (komentáře, veškeré texty pro dialogy typu MsgBox apod.). Nemusíme si pak lámat hlavu s jazykem cílového operačního systému Windows a balíčku Office.

MsgBox z VBA nezvládá cizí abecedy, jako je třeba azbuka. K tomu účelu musíme rovněž volat API funkci podporující Unicode (viz [článek](#)).

Pozn. Zžidlevstáníhodná je i skutečnost, že není možné kopírovat napřímo kód VBA do okna aplikace Skype...