

Porovnání dvou seznamů (listů, tabulek) je v praxi velmi frekventovaným tématem. Nejčastější úlohou je tvorba nového ceníku na základě stávajícího ve spojení s aktualizovaným seznamem dodavatele. S čím se můžeme setkat:

a) U produktu došlo k pozastavení (ukončení) výroby (tj. produkt z našeho seznamu již nenacházíme v seznamu dodavatele)

V tom případě probíhá výprodej a cena zůstává stejná až do vyprodání zásob. Po té by měl být produkt odstraněn z výběru. Ne úplně fyzicky, spíš doporučuji jakousi archivaci z důvodu zachování historie a kupříkladu celoročního vyhodnocení prodeje.

b) U produktu došlo ke změně ceny (tj. produkt se nachází v našem seznamu i v seznamu dodavatele) Cena produktu se pravděpodobně přizpůsobí nové ceně od dodavatele.

c) Nový produkt (produkt se nachází pouze v seznamu dodavatele)

Novinku je potřeba zařadit do stávajícího seznamu včetně zaváděcí ceny.

Odhadem 80 % běžných uživatelů znalých práce na listu se při porovnávání seznamů obrací na funkce. Pravděpodobně první na rání je v Excelu funkce SVYHLEDAT. Následující snímek (viz také příloha) ukazuje řešení základních požadavků.

	A	B	C	D	E	F	G	H	I	J	K	
1												
2		<i>DataStara</i>										
3												
4		IDStara	Zboží	Částka		Nalezeno v IDNova?	Na jakém řádku nalezeno?	Nová částka?		Na jakém řádku nalezeno?	Nová částka?	
5		1111	Tričko bílé	200 Kč		ne				nenalezeno	nenalezeno	
6		1112	Tričko modré	200 Kč		ano	19	180 Kč		19	180 Kč	
7		2222	Košíle	350 Kč		ano	15	350 Kč		15	350 Kč	
8		3333	Čepice	150 Kč		ne				nenalezeno	nenalezeno	
9		4444	Kalhoty	850 Kč		ano	16	900 Kč		16	900 Kč	
10		1111	Tričko	200 Kč		ne				nenalezeno	nenalezeno	
11												
12		<i>DataNova</i>										
13												
14		IDNova	Zboží	Částka		Nalezeno v IDStara?	Na jakém řádku nalezeno?	Stará částka?		Na jakém řádku nalezeno?	Nová částka?	
15		2222	Košíle	350 Kč		ano	7	350 Kč		7	350 Kč	
16		4444	Kalhoty	900 Kč		ano	9	850 Kč		9	850 Kč	
17		5555	Bunda	1 500 Kč		ne				nenalezeno	nenalezeno	
18		6666	Rukavice	120 Kč		ne				nenalezeno	nenalezeno	
19		1112	Tričko modré	180 Kč		ano	6	200 Kč		6	200 Kč	
20												
21												
22		F5: =KDYŽ(JE.NEDEF(POZVYHLEDAT(B5;IDNova;0)));"ne";"ano")									<i>DataStara</i>	=B5:D10
23		G5: =KDYŽ(JE.NEDEF(POZVYHLEDAT(B5;IDNova;0)));"";POZVYHLEDAT(B5;IDNova;0)+ŘÁDEK(IDNova)-1)									<i>IDStara</i>	=B5:B10
24		H5: =KDYŽ(JE.NEDEF(POZVYHLEDAT(B5;IDNova;0)));"";SVYHLEDAT(B5;DataNova;3;NEPRAVDA))									<i>DataNova</i>	=B15:D19
25												
26		J5: =IFERROR(POZVYHLEDAT(B5;IDNova;0)+ŘÁDEK(IDNova)-1;"nenalezeno")									<i>IDNova</i>	=B15:B19
27		K5: =IFERROR(SVYHLEDAT(B5;DataNova;3;NEPRAVDA);"nenalezeno")										
28												
29		<i>Maticový vzorec pro porovnávání výskytů ID nelze nasadit, protože každý zdroj dat obsahuje různý počet záznamů.</i>										
30												

Porovnání seznamů - vyhledávací funkce

Uvedené vzorce nepředstavují žádné geniální know-how, zkrátka dělají, co mají. Funkce JE.NEDEF, resp. IFERROR ošetřují stavy, kdy jedna či druhá vyhledávací funkce nenajde položku v seznamu.

Funkce ŘÁDEK poslouží k dopočtu pořadí hledané položky vůči počátku listu (A1). Ačkoliv pracuje s oblastí, vrátí řádek její první buňky. Pojmenované oblasti lehce zpřehledňují celý zápis, nejsou nutné. Tím bychom mohli skončit, nicméně posuneme řešení na vyšší úroveň. Tu reprezentuje databázový přístup k seznamům. I k sešitům, listům a seznamům na nich se dá přistupovat prostřednictvím jazyka SQL. My se budeme držet Excelu 2010, ve kterém máme k dispozici k opracování dat pazourek, tedy Microsoft Query.

Sešity s daty pro porovnání

Porovnání probíhá na základě jedinečného kódu produktu (klíče), nikdy ne na základě textového popisu (přejmenovaný produkt, chyby v diakritice, nadbytečné mezery, ...). I tak by prvním krokem měla být kontrola stávajícího seznamu (v příkladu viz duplicita trička s kódem 1111).

Dotazy do seznamů jsem rozdělil na dva poddotazy slovně asi takto:

- A) „Vezmi všechny produkty stávajícího seznamu, a pokud k nim existují nové ceny, doplň je.“
- B) „Pokud se v dodaném seznamu nacházejí nové produkty, doplň je do stávajícího i s cenou.“

Tento článek si neklade za cíl vás naučit pracovat v editoru Microsoft Query, proto jen letný textový postup pro první poddotaz.

- 1) Data / Z jiných zdrojů / Z aplikace Microsoft Query / dialog Zvolit zdroj dat / záložka Databáze, Excel Files
- 2) Dialog Vybrat sešit, Zobrazit soubory typu: Všechny soubory (aby byly vidět i novodobé sešity s příponou XLSX) a v části Název databáze vybrat sešit se stávajícím seznamem.
- 3) Dialog Průvodce dotazem, převést sloupce (pokud nevidíte žádnou dostupnou tabulku, pod tlačítkem Možnosti zaškrtněte Systémové tabulky) a v poslední části dialogu vybrat volbu Zobrazit data nebo upravit dotaz v aplikaci Microsoft Query.

- 4) V okně Microsoft Query zvolit Tabulka / Přidat tabulky, a vybrat sešit s dodaným seznamem (rozbalovací nabídka napravo od popisu Sešit, tlačítko Přidat).
- 5) Ze zástupců obou seznamů přetažením položek do tabulkové části poskládat dotazované sloupce a případně přejmenovat hlavičky.
- 6) Přetáhnout položku ID ze zástupce stávajícího seznamu nad položku ID v zástupci dodaného seznamu, poklepat na vzniklou spojnicu, zvolit typ vazby a potvrdit výběr tlačítkem Přidat. Jsem si vědom tohoto, že přesně tento bod je překážkou, přes kterou se řada uživatelů špatně přenáší (spojování tabulek typu INNER JOIN, LEFT JOIN, ...).

The screenshot shows the Microsoft Query interface with two tables, 'ListDataStara\$' and 'ListDataNova\$', being joined. The 'Spojení' dialog box is open, showing a left outer join between 'ListDataStara\$.ID' and 'ListDataNova\$.ID'. The 'SQL' dialog box shows the resulting SQL query:

```
SELECT 'ListDataStara$.ID, 'ListDataStara$.Zboží,
'ListDataStara$.Částka AS 'Stará částka', 'ListDataNova$.Částka
AS 'Nová částka'
FROM {oj 'D:\SesitDataStara.xlsx'; 'ListDataStara$'; 'ListDataStara$'
LEFT OUTER JOIN 'D:\SesitDataNova.xlsx'; 'ListDataNova$'
'ListDataNova$' ON 'ListDataStara$.ID = 'ListDataNova$.ID}'
```

Below the dialog boxes, a table displays the results of the query:

	ID	Zboží	Stará částka	Nová částka
	1111.0	Tričko bílé	200,0000	
	1112.0	Tričko modré	200,0000	180,0000
	2222.0	Košile	350,0000	350,0000
	3333.0	Čepice	150,0000	
	4444.0	Kalhoty	850,0000	900,0000
	1111.0	Tričko	200,0000	

Microsoft Query – tvorba dotazu v designéru

	A	B	C	D
1	ID	Zboží	'Stará částka'	'Nová částka'
2	1111	Tričko bílé	200 Kč	
3	1112	Tričko modré	200 Kč	180 Kč
4	2222	Košile	350 Kč	350 Kč
5	3333	Čepice	150 Kč	
6	4444	Kalhoty	850 Kč	900 Kč
7	1111	Tričko	200 Kč	

Vlastnosti připojení

Název připojení: Dotaz z Excel Files
 Popis:

Použití: Použití | Definice

Typ připojení: Databázový dotaz
 Soubor připojení: Proházet...
 Vždy použít soubor připojení

Připojovací řetězec: DSN=Excel Files;DBQ=D:\SesitDataStara.xlsx;DefaultDir=D:\;DriverId=1046;MaxBufferSize=2048;PageTimeout=5;

Uložit heslo

Typ příkazu: SQL

Text příkazu:


```
SELECT 'ListDataStara$.ID', 'ListDataStara$.Zboží', 'ListDataStara$.Částka AS 'Stará částka', 'ListDataNova$.Částka AS 'Nová částka'
FROM {oj 'D:\SesitDataStara.xlsx', 'ListDataStara$' 'ListDataNova$' LEFT OUTER JOIN
'D:\SesitDataNova.xlsx', 'ListDataNova$' 'ListDataNova$' ON 'ListDataStara$.ID =
'ListDataNova$.ID}
```

Excel Services: Nastavení ověření...
 Upravit dotaz... Parametry... Exportovat soubor připojení...
 OK Storno

Výsledek Microsoft Query – Vlastnosti připojení

Nyní si dovoluji malou odstavku tématu a povzdechnutí...

Před sepsáním tohoto článku jsem několik hodin řešil, proč mi Microsoft Query špatně zobrazuje české znaky.

Nějaký dobrák počestil designer v Microsoft Query natolik, že přimotal český překlad i do klíčových slov jazyka SQL. Pamatuji si na kurz, kdy se jeho účastníci chtěli naučit pracovat v Microsoft Query. A chápal jsem jejich finální „no to ne, to si radši dotazy a pohledy připravíme v Accessu“.

I když Microsoft Query umí pracovat s definovanými názvy, ignoruje Tabulky.

Excel sám neví, které chtějí uvozovky v popiscích sloupců a textových řetězcích (zpětné, jednoduché, dvojité). Pro existující názvy sloupců používá jednoduché zpětné, pro textové řetězce snad očekává jednoduché uvozovky, nicméně ty pak viditelně promítá do hlavičky na listu. A editor Microsoft Query má problémy jak s jednoduchými, tak s dvojitými.

Pro Microsoft Query nebyla nikdy pořádně dokončena nápověda a tak leckdy hádáme, co použít za funkce, datové typy, a jaká syntaxe je platná. Na hromadě leží pojmy jako Microsoft Jet, ADO DB a nezdá se, že bychom měli pocit, že se pohybujeme ve VBA (funkce IIf a jiné).

Humor nás definitivně přejde v případě, že zdrojem dat jsou právě listy Excelu. Čeká nás peklo s rozpoznáváním typu hodnot ve sloupcích (viz pojem IMEX).

Excel 2010 navíc nezvládá běžným způsobem importovat časové údaje (musí se převádět na desetinné číslo).

Prostě žůžo dobrodružo.

Možná i proto druhý poddotaz a další úpravy už zpravidla řeším rozkopírováním a úpravou přímo v

Excelu a jeho dialogu Vlastnosti připojení (do kterého se proklikávat co půl minuty při ladění je také radost).

Ačkoliv nejsem masochista, úlohu jsem si před spojením poddotazů ještě o trochu ztížil. Jednak je pro další práci vhodnější mít nové ceny ve stejném sloupci, jednak jsem chtěl slovně okomentovat nedohledané produkty.

	A	B	C	D	E	F	G
1	ID	Zboží	Stará částka	Nová částka	Finální částka	Poznámka	
2	1111	Tričko	200 Kč		200 Kč	výprodej	
3	1111	Tričko bílé	200 Kč		200 Kč	výprodej	
4	1112	Tričko modré	200 Kč	180 Kč	180 Kč		
5	2222	Košile	350 Kč	350 Kč	350 Kč		
6	3333	Čepice	150 Kč		150 Kč	výprodej	
7	4444	Kalhoty	850 Kč	900 Kč	900 Kč		
8	5555	Bunda		1 500 Kč	1 500 Kč	novinka	
9	6666	Rukavice		120 Kč	120 Kč	novinka	
10							

Konečná podoba tabulky

Finální SQL dotaz tedy může vypadat nějak takto (nejsem databázovým expertem, proto mi prosím případně odpusťte složitější formu):

```

1 SELECT `ListDataStara$`.ID, `ListDataStara$`.Zboží, `ListDataStara$`.Částka AS `Stará částka`, `ListDataNova$`
2 .Částka AS `Nová částka`, IIf(IsNull(`ListDataNova$`.Částka), `ListDataStara$`.Částka, `ListDataNova$`.Částka)
3 AS `Finální částka`, IIf(IsNull(`ListDataNova$`.Částka), 'výprodej', '') AS `Poznámka`
4 FROM [oj `D:\SesitDataStara.xlsx`. `ListDataStara$` `ListDataStara$` LEFT OUTER JOIN `D:\SesitDataNova.xlsx`. `
5 ListDataNova$` `ListDataNova$` ON `ListDataStara$`.ID = `ListDataNova$`.ID)
6 UNION
7 SELECT `ListDataNova$`.ID, `ListDataNova$`.Zboží, `ListDataStara$`.Částka AS `Stará částka`, `ListDataNova$`.
8 Částka AS `Nová částka`, `ListDataNova$`.Částka AS `Finální částka`, IIf(IsNull(`ListDataStara$`.Částka),
9 'novinka', '') AS `Poznámka`
10 FROM [oj `D:\SesitDataNova.xlsx`. `ListDataNova$` `ListDataNova$` LEFT OUTER JOIN `D:\SesitDataStara.xlsx`. `
11 ListDataStara$` `ListDataStara$` ON `ListDataNova$`.ID = `ListDataStara$`.ID)
12 WHERE (`ListDataStara$`.ID Is Null)

```

Konečná podoba SQL dotazu

Sloučení poddotazů v jeden zajišťuje klíčové slovo Union. Bez něj se horko těžko obejdeme i v případě, že potřebujeme prosté spojení datových zdrojů stejné struktury v jeden. Tuhle jednoduchou věc neumí ani souhrny, ani kontingenční tabulky, jež musí za každou cenu nad daty provádět nějakou matematickou operaci.

Tip: SQL dotaz, který vrátilo Microsoft Query, si zkopírujte kupříkladu do textového editoru Notepad++ a zvolte syntaxi SQL. Klíčová slova se obarví a s textem je pak snazší pracovat (viz i obrázek výše).

Příloha

[excel-porovnani-seznamu.zip](#)