

2. rozšířené vydání

SQL pro blbce

Vladimír Klaus
Jaroslav Hajný



CREATE INDEX, INTERSECT, DROP TRIGGER, SWITCH..

SQL pro blbce

Copyright © 2014 by Vladimír Klaus, Jaroslav Hajný

Cover design © 2014 by Vladimír Klaus

Veškerá práva vyhrazena. Žádná část tohoto díla nesmí být reprodukována ani elektronicky přenášena či šířena bez předchozího písemného souhlasu majitele autorských práv.

Odpovědný redaktor: Jiří Vlček

Jazyková redakce: AUDREY software

Korektura: Vladimír Tobyáš

Sazba: Vladimír Klaus

Obálka: Vladimír Klaus

Vydání druhé

Vydalo nakladatelství Palmknihy s.r.o., Malá Štupartská 7, Praha 1, v únoru 2014

ISBN: 978-80-7486-033-1 (ePub)

ISBN: 978-80-7486-034-8 (mobi)

ISBN: 978-80-7486-035-5 (PDF)



Obsah

Úvod	7
Komu je příručka určena	7
Co zde najdete	7
Doprovodná databáze	7
Poděkování čtenářům	7
Co je nového	8
Základní příkazy	10
Příkaz SELECT	10
Příkaz ORDER BY	11
Příkaz WHERE	13
Příkaz GROUP BY	15
Příkaz HAVING	17
Agregační funkce	19
Funkce SUM	19
Funkce COUNT	19
Funkce AVG	20
Funkce MIN a MAX	21
Funkce FIRST a LAST	21
Statistické agregační funkce	24
Funkce STDEV	24
Funkce STDEVP	25
Funkce VAR	25
Funkce VARP	26
Další funkce	27
Funkce LCASE, UCASE	27
Funkce LEFT, RIGHT	27
Funkce MID	28
Funkce TRIM, LTRIM, RTRIM	29
Funkce REPLACE	30
Funkce SWITCH	31

Funkce CHOOSE	33
Funkce LEN	34
Přetypování čísla na text	35
Přetypování textu na číslo	37
Funkce ASC a ASCW	39
Funkce CHR a CHRW	40
Funkce SGN	42
Funkce DATEPART	45
Funkce ROUND	47
Trigonometrické funkce	50
Aritmetické funkce	53
Spojování tabulek	57
Příkaz INNER JOIN	57
Příkaz LEFT JOIN	59
Příkaz RIGHT JOIN	61
Příkazy INNER JOIN, LEFT JOIN a RIGHT JOIN graficky	62
Příkaz OUTER JOIN	62
Příkaz CROSS JOIN	63
Příkaz UNION a UNION ALL	66
Příkaz INTERSECT	68
Příkaz EXCEPT	69
Predikáty	72
Predikát DISTINCT	72
Predikát DISTINCTROW	73
Predikát TOP	75
Predikát LIKE	79
Predikát IS NULL	80
Predikát BETWEEN	82
Predikát IN	83
Predikát EXISTS	84
Predikáty ANY, SOME, ALL	86
Úprava dat v tabulkách	88

Příkaz INSERT	88
Příkaz UPDATE	90
Příkaz DELETE	90
Import dat z CSV	92
Práce s tabulkami	98
Příkaz CREATE TABLE	98
Příkaz ALTER TABLE	99
Příkaz DROP TABLE	101
Příkaz CREATE INDEX	101
Příkaz DROP INDEX	103
Referenční integrita	103
Praktické příklady	107
Duplicity v tabulkách	107
Ovlivnění pořadí ve výsledku dotazu	112
Triggery	118
Co to jsou triggery	118
Přidání triggeru	118
Odstranění triggeru	121
Chyba v triggeru	123
Ostatní	124
Proč zde používáme MS Access 2010?	124
Jak převést MDB databázi na novější ACCDB	124
Co to je NULL	125
SQL příkazy nejsou case-sensitive	126
Pozor na přehození ORDER a WHERE	126
Pořadí vyhodnocení dotazu	127
Používání uvozovek, apostrofů a středníků	127
Co to je ID typu Automatické číslo	128
Alias	128
Tabulky v příkladech	131
Tabulka Zakaznici	132
Tabulka Produkty	132

Tabulka Objednavky	133
Tabulka Aritmetika	135
Tabulka Kontakty	135
Tabulka Log.....	136
Tabulka Pomocna	136
Tabulka TrigonometrickeUdaje	136
Další příručky	137
Již vyšlo	137
Připravujeme	138
O autorech	139
AUDREY software	139
Chcete nám napsat?	140

Úvod

Komu je příručka určena

Všem, kteří se chtějí seznámit s jazykem SQL, ale i pro ty, kteří celý život s SQL zápasí a občas si potřebují některé detaily nebo málo známé příkazy připomenout. Všechny příkazy a funkce jsou názorně a velmi jednoduše vysvětlovány a předvedeny na mnoha konkrétních příkladech z praxe.

Co zde najdete

- ➔ **140 stránek** plných snadno pochopitelného čtení
- ➔ přehledné členění do **90 kapitol**
- ➔ **180 příkladů SQL** včetně komentářů
- ➔ **220 obrázků** doprovázejících všechny příklady
- ➔ vysvětlení nejpoužívanějších **příkazů, funkcí, predikátů**
- ➔ užitečná upozornění a poznámky z praxe
- ➔ odlišnosti příkazů v různých databázích
- ➔ ...a další řadu bonusů a článků na [webu](#)

Doprovodná databáze



V příručce jsou na všech místech používané příklady, které vycházejí z existující MS Access databáze, která je **zdarma k dispozici na stránkách** <http://www.sqlproblbce.cz>.

Databáze také obsahuje velké množství SQL dotazů k okamžitému vyzkoušení. Na webu najdete dvě verze – klasický MDB soubor (pro MS Access 2002 a vyšší) a pak novější verzi ACCDB (pro MS Access 2007 a vyšší).

Poděkování čtenářům

Hned na úvod chceme poděkovat, že jste si zakoupili tuto příručku. Nebo nezakoupili a přesto chcete rychle a jednoduše podpořit autory v další práci?

Pak stačí jediné – převést **99 Kč** na účet **19-4092520297/0100**.

Děkujeme.

Co je nového

Pokud vlastníte první verzi této příručky, možná oceníte seznam zcela nových kapitol, tak i seznam kapitol, ve kterých došlo k nějakým změnám, doplnění příkladů nebo opravám.

Zcela nové kapitoly

- Funkce FIRST a LAST
- Funkce REPLACE
- Funkce SWITCH
- Funkce CHOOSE
- Funkce LEN
- Přetypování čísla na text
- Přetypování textu na číslo
- Funkce ASC a ASCW
- Funkce CHR a CHRW
- Funkce SGN
- Trigonometrické funkce
- Aritmetické
- Příkaz INTERSECT
- Příkaz EXCEPT
- Import dat z CSV
- Příkaz CREATE INDEX
- Příkaz DROP INDEX
- Referenční integrita
- Duplicity v tabulkách
- Ovlivnění pořadí ve výsledku dotazu
- Co to jsou trigger
- Přidání triggeru
- Odstranění triggeru
- Chyba v triggeru
- Jak převést MDB databázi na novější ACCDB
- Pozor na přehození ORDER a WHERE
- Pořadí vyhodnocení dotazu

Změněné kapitoly

- Doprovodná databáze
- Příkaz ORDER BY
- Funkce COUNT
- Funkce MIN a MAX
- Funkce STDEV
- Funkce LCASE, UCASE

- Funkce MID
- Funkce ROUND
- Příkaz INNER JOIN
- Příkaz LEFT JOIN
- Příkaz UNION a UNION ALL
- Predikát DISTINCT
- Predikát LIKE
- Predikát IN
- Predikát EXISTS
- Příkaz INSERT
- Příkaz ALTER TABLE
- Příkaz DROP TABLE
- Co to je NULL
- Aliasy
- Tabulky v příkladech

Základní příkazy

Příkaz SELECT

Tento základní a nejčastěji používaný příkaz slouží k získání (výběru) dat z databázové tabulky. Začínáme slovem SELECT (vybrat), následuje „,“ (všechny sloupce), klíčové slovo FROM (odkud) a nakonec i název tabulky.

Tímto příkazem provedeme výběr všech sloupců (*) a všech záznamů (řádek) z tabulky Zakaznici.

SELECT * FROM Zakaznici

ID	Prijmeni	Jmeno	Ulice	Mesto	PSC	Telefon
1	Klaus	Vladimír	Sokolovská 122	Praha 8	180 00	
2	Hajný	Jaroslav	Na příkopě 16	Praha 1	110 00	+420 608 456 789
3	Nováková	Jana	Dlouhá 17	Jičín 1	506 01	
4	Veselý	Libor	Křížíkova 88	Praha 8	180 00	
5	Nováčková	Helena	U jezu	Brno 2	602 00	604 222 555
6	Sobota	Jaroslav		Jičíněves 47	507 31	
7	Malá	Zuzana		Mirošovice 379	251 66	+420 608 147 258
8	Kuklík	Jan	Pernerova 51	Praha 8	180 00	+420 604 166 166
9	Balabán	Květoslav	Sokolovská 173	Praha 9	190 00	
10		Lukáš	Ke Smíchovu 47	Praha 5	150 00	
11	Jelínek		Sakařova 5a	Pardubice	530 03	605 147 147

Následující příkaz také vybírá všechny záznamy, ale již pouze sloupce ID a Jmeno.

SELECT ID, Jmeno FROM Zakaznici

ID	Jmeno
1	Vladimír
2	Jaroslav
3	Jana
4	Libor
5	Helena
6	Jaroslav
7	Zuzana
8	Jan
9	Květoslav
10	Lukáš
11	

Upozornění: Takže raději ještě jednou! Hvězdička neznamená všechny řádky, ale všechny sloupce. Všechny řádky to jsou automaticky, pokud ne zadáme nějakou podmínku, viz dále příkaz [WHERE](#).

Příkaz SELECT je možné dále vylepšovat pomocí dalších příkazů, funkcí a predikátů, což bude popsáno v dalších částech.

Příkaz ORDER BY

Už tedy umíme vybírat data z tabulky příkazem SELECT, nyní se je naučíme řadit. Pokud totiž příkaz k řazení nebude použit, budou záznamy většinou v pořadí, v jakém byly do tabulky vkládány. Přesněji řečeno – pořadí není definováno.

Takto jednoduše seřadíme všechny zákazníky dle příjmení, a to **vzestupně** – v tomto případě (jedná se o textové pole) tedy od A do Z.

SELECT * FROM Zakaznici ORDER BY Prijmeni

ID	Prijmeni	Jmeno	Ulice	Mesto	PSC	Telefon
10		Lukáš	Ke Smíchovu 47	Praha 5	150 00	
9	Balabán	Květoslav	Sokolovská 173	Praha 9	190 00	
2	Hajný	Jaroslav	Na příkopě 16	Praha 1	110 00	+420 608 456 789
11	Jelínek		Sakařova 5a	Pardubice	530 03	605 147 147
1	Klaus	Vladimír	Sokolovská 122	Praha 8	180 00	
8	Kuklík	Jan	Pernerova 51	Praha 8	180 00	+420 604 166 166
7	Malá	Zuzana		Mirošovice 379	251 66	+420 608 147 258
5	Nováčková	Helena	U jezu	Brno 2	602 00	604 222 555
3	Nováková	Jana	Dlouhá 17	Jičín 1	506 01	
6	Sobota	Jaroslav		Jičíněves 47	507 31	
4	Veselý	Libor	Křižíkova 88	Praha 8	180 00	

Chceme-li řadit **sestupně**, použijeme klíčové slovo DESC (u vzestupného se dá zase použít ASC, ale to není povinné). Takto získáme záznamy o prodejích, řazené dle částky od nejvyšší do nejnižší.

SELECT * FROM Objednavky ORDER BY Castka DESC

ID ▾	IdZakaznika ▾	IdProduktu ▾	Castka ▾	Datum ▾	Vyrizeno ▾	Poznamka ▾
20	3	6	4 499,00 Kč		☐	
3	2	6	4 499,00 Kč	15.7.2012	☐	
13	1	6	4 499,00 Kč	20.9.2012	☒	
5	3	1	2 999,00 Kč	5.8.2012	☐	
16	8	1	2 999,00 Kč	15.10.2012	☒	
1	1	1	2 999,00 Kč	14.7.2012	☒	
22	23	7	1 999,00 Kč	16.12.2012	☐	
8	7	7	1 999,00 Kč	30.8.2012	☒	
14	4	10	1 790,00 Kč	30.9.2012	☐	
7	8	10	1 790,00 Kč	23.8.2012	☒	
21		5	888,00 Kč	15.12.2012	☐	
9	4	5	888,00 Kč	2.9.2012	☒	
17	6	3	799,00 Kč	27.10.2012	☐	
11	2	8	650,00 Kč	14.9.2012	☐	
15	7	8	650,00 Kč	5.10.2012	☐	
10	6	12	599,00 Kč	14.9.2012	☐	
18	1	12	599,00 Kč	2.11.2012	☐	
19	10	2	480,00 Kč	13.12.2012	☐	
6	6	2	450,00 Kč	23.8.2012	☐	
2	1	2	450,00 Kč	14.7.2012	☐	
4	1	9	399,00 Kč	3.8.2012	☒	
12	3	9	399,00 Kč	14.9.2012	☒	

A dá se řadit podle více sloupců? Samozřejmě! Krásným příklad je seřazení zákazníků dle měst a pak (dalo by se říci v rámci měst) i dle příjmení a (v rámci příjmení) také dle jména. V níže uvedeném příkladu, kde je velmi málo dat v tabulce, je to dobře patrné pouze u Prahy 8 a Příjmení.

SELECT * FROM Zakaznici ORDER BY Mesto, Prijmeni, Jmeno

ID ▾	Prijmeni ▾	Jmeno ▾	Ulice ▾	Mesto ▾	PSC ▾	Telefon ▾
5	Nováčková	Helena	U jezu	Brno 2	602 00	604 222 555
3	Nováková	Jana	Dlouhá 17	Jičín 1	506 01	
6	Sobota	Jaroslav		Jičíněves 47	507 31	
7	Malá	Zuzana		Mirošovice 379	251 66	+420 608 147 258
11	Jelínek		Sakařova 5a	Pardubice	530 03	605 147 147
2	Hajný	Jaroslav	Na příkopě 16	Praha 1	110 00	+420 608 456 789
10		Lukáš	Ke Smíchovu 47	Praha 5	150 00	
1	Klaus	Vladimír	Sokolovská 122	Praha 8	180 00	
8	Kuklík	Jan	Pernerova 51	Praha 8	180 00	+420 604 166 166
4	Veselý	Libor	Křižíkova 88	Praha 8	180 00	
9	Balabán	Květoslav	Sokolovská 173	Praha 9	190 00	

Ještě jedna specialita. Vypíšeme všechny zákazníky, seřadíme dle PSČ a dále pak podle toho, jak byli přidáváni do tabulky, ale počínaje naposledy přidaným. Využijeme k tomu ID, což je údaj, který se

automaticky zvyšuje o 1 při každém přidání záznamu. Samozřejmě se nezvyšuje sám od sebe, ale musíme [zvyšování tomuto poli nastavit](#).

```
SELECT * FROM Zakaznici ORDER BY PSC, ID DESC
```

ID ▾	Prijmeni ▾	Jmeno ▾	Ulice ▾	Mesto ▾	PSC ▾	Telefon ▾
2	Hajný	Jaroslav	Na příkopě 16	Praha 1	110 00	+420 608 456 789
10		Lukáš	Ke Smíchovu 47	Praha 5	150 00	
8	Kuklík	Jan	Pernerova 51	Praha 8	180 00	+420 604 166 166
4	Veselý	Libor	Křižíkova 88	Praha 8	180 00	
1	Klaus	Vladimír	Sokolovská 122	Praha 8	180 00	
9	Balabán	Květoslav	Sokolovská 173	Praha 9	190 00	
7	Malá	Zuzana		Mirošovice 379	251 66	+420 608 147 258
3	Nováková	Jana	Dlouhá 17	Jičín 1	506 01	
6	Sobota	Jaroslav		Jičíněves 47	507 31	
11	Jelínek		Sakařova 5a	Pardubice	530 03	605 147 147
5	Nováčková	Helena	U jezu	Brno 2	602 00	604 222 555

Aby to nebylo tak jednoduché, je tu ještě jedna možnost, jak zadat řazení výsledku – a to pomocí relativní pozice sloupců ve výsledku. To může být velká výhoda, pokud měníte sloupce za příkazem SELECT, ale zároveň to ze stejného důvodu může způsobit neočekávané řazení – když někdo pouze změní vybírané sloupce a zapomene upravit hodnoty v ORDER BY.

```
SELECT Nazev, Cena FROM Produkty ORDER BY 2, 1
```

Tento příkaz tedy odpovídá tomuto:

```
SELECT Nazev, Cena FROM Produkty ORDER BY Cena, Nazev
```

Příkaz WHERE

Zatím jsme vždy vybírali všechny záznamy, nyní se je naučíme filtrovat pomocí jedné nebo více podmínek.

Takto získáme všechny zákazníky, jejichž příjmení je Sobota.

```
SELECT * FROM Zakaznici WHERE Prijmeni = "Sobota"
```

ID ▾	Prijmeni ▾	Jmeno ▾	Ulice ▾	Mesto ▾	PSC ▾	Telefon ▾
6	Sobota	Jaroslav		Jičíněves 47	507 31	

A takto získáme záznamy objednávek, které byly alespoň za 1000 (korun).

```
SELECT * FROM Objednavky WHERE Castka >= 1000
```

ID	IdZakaznika	IdProduktu	Castka	Datum	Vyrizeno	Poznamka
1	Klaus	MP3 přehrávač	2 999,00 Kč	14.7.2012	☒	
3	Hajný	Psací souprava	4 499,00 Kč	15.7.2012	☐	
5	Nováková	MP3 přehrávač	2 999,00 Kč	5.8.2012	☐	
7	Kuklík	Žehlička	1 790,00 Kč	23.8.2012	☒	
8	Malá	Kabelka	1 999,00 Kč	30.8.2012	☒	
13	Klaus	Psací souprava	4 499,00 Kč	20.9.2012	☒	
14	Veselý	Žehlička	1 790,00 Kč	30.9.2012	☐	
16	Kuklík	MP3 přehrávač	2 999,00 Kč	15.10.2012	☒	
20	Nováková	Psací souprava	4 499,00 Kč		☐	
22		Kabelka	1 999,00 Kč	16.12.2012	☐	

A co když potřebujeme všechny objednávky pana Soboty, za které utratil více než 500 korun? Tak to je už složitější úkol (údaje jsou ve dvou tabulkách), o tom si povíme později v části [INNER JOIN](#).

Kombinace podmínek se provádí pomocí logických operátorů (AND, OR apod.). Takto získáme všechny zákazníky z Prahy 8, kteří se jmenují Veselý.

```
SELECT * FROM Zakaznici WHERE Prijmeni = "Veselý" AND Mesto = "Praha 8"
```

AND = a současně, tj. musí platit obě podmínky.

ID	Prijmeni	Jmeno	Ulice	Mesto	PSC	Telefon
4	Veselý	Libor	Křížíkova 88	Praha 8	180 00	

Takto získáme všechny produkty, které se jmenují „Myš“ nebo „Klávesnice“.

```
SELECT * FROM Produkty WHERE Nazev = "Myš" OR Nazev = "Klávesnice"
```

OR = nebo, tj. musí platit buď jedna podmínka, nebo druhá, případně obě.

ID	Nazev	Cena	Popis
9	Myš	399,00 Kč	Myš Logitech Mouse M105 je skvělá barevná kabelová myš za rozumnou cenu. Díky provedení s technologií plug-

Jak vidíte, výsledkem je pouze jeden záznam, protože produkt, který by se jmenoval „Klávesnice“ vůbec neexistuje.

Malinko složitější je sestavení dotazu, kde je v podmínce uvedeno datum. Ono jde totiž o to, že se datum v různých zemích může zapisovat různě, a proto bylo třeba vymyslet a stanovit jednotný zápis bez ohledu na zemi nebo nějaké lokální nastavení. Tímto zápisem je **#měsíc/den/rok#**.

Poznámka: V jiných databázích je možné (nutné) použít místo znaku # apostrof. V MS Accessu to nelze, protože by to pak chápal jako textový řetězec

Pomocí tohoto dotazu získáme objednávky, které byly provedeny 14. 9. 2012.