
Stručný obsah

	Předmluva	13
1.	Přístup k záznamům	27
2.	Třídění výsledků dotazu	39
3.	Práce s více tabulkami	51
4.	Vkládání, aktualizace, mazání	83
5.	Dotazy na metadata	103
6.	Práce s řetězcí	115
7.	Práce s čísly	169
8.	Výpočty s kalendářními daty	197
9.	Operace s kalendářními daty	225
10.	Práce s intervaly	299
11.	Pokročilé vyhledávání	325
12.	Analytické funkce a tvorba reportů	361
13.	Hierarchické dotazy	431
14.	Doplňující příklady	459
A.	Analytické funkce	509
B.	Návrat k Rozenshteinovi	533

Předmluva **13**

1. Přístup k záznamům **27**

Přístup ke všem řádkům a sloupcům tabulky	27
Přístup k podmnožinám řádků tabulky	28
Vyhledávání řádek, které splňují různé podmínky	28
Přístup k podmnožině sloupců tabulky	29
Změna názvů sloupců výsledné množiny pro lepší čitelnost výsledků dotazu	29
Odkaz na sloupec prostřednictvím jeho přezdívky v klauzuli WHERE	30
Řetězení hodnot záznamů ve sloupcích	31
Používání podmínkové logiky v příkazu SELECT	32
Omezení počtu řádků výsledné množiny dotazu	33
Vygenerování výsledné množiny, která obsahuje n náhodných záznamů tabulky	35
Nalezení prázdných hodnot záznamů	36
Převod prázdných hodnot záznamů na hodnoty reálné	36
Hledání vzorů	37

2. Třídění výsledků dotazu **39**

Seřazení výsledné množiny dotazu v určitém pořadí	39
Třídění výsledků dotazu podle více různých polí	40
Třídění podle podřetězců	41
Třídění smíšených alfanumerických dat	42
Třídění výsledků a prázdné záznamy	44
Třídění podle hodnoty ve sloupci	50

3. Práce s více tabulkami **51**

Zobrazení dvou množin řádků najednou	51
Společné řádky	53
Nalezení společných řádek ve dvou různých tabulkách	54
Návrat hodnot záznamů, které existují pouze v jedné z více tabulek	56

Návrat řádků tabulky, které neodpovídají řádkům v jiné tabulce	59
Tvorba nových spojení v dotazu tak, aby nekolidovala se spojeními stávajícími	60
Určení shodnosti dvou tabulek	63
Detekce kartézských součinů a vyhnutí se jejich generování	69
Tvorba spojení při agregacích	70
Tvorba vnějších spojení při agregacích	74
Nalezení řádků, které nejsou společné pro více tabulek	77
Operace s prázdnými záznamy a jejich srovnávání	80

4. Vkládání, aktualizace, mazání **83**

Vložení nového záznamu	83
Vkládání standardních hodnot	84
Přepsání standardní hodnoty záznamu na prázdnou hodnotu záznamu	85
Kopírování řádků z jedné tabulky do druhé	86
Kopírování vlastností tabulky	86
Vkládání řádků do více tabulek najednou	87
Omezení vkládání do určitých sloupců	89
Modifikace záznamů v tabulce	90
Aktualizace řádků tabulky za předpokladu, že tyto řádky existují v jiné tabulce	91
Aktualizace hodnot záznamů řádku tabulky podle hodnot záznamů řádku jiné tabulky	92
Slučování záznamů	95
Smazání všech záznamů tabulky	96
Mazání specifických záznamů	96
Smazání jednoho záznamu	97
Mazání záznamů narušujících referenční integritu	97
Mazání duplikátů	98
Mazání záznamů, na které se odkazují záznamy jiné tabulky	100

5. Dotazy na metadata **103**

Výpis tabulek schématu	103
Výpis sloupců tabulky	104
Výpis sloupců tabulky, které se indexují	105
Výpis omezení pro danou tabulku	106
Výpis cizích klíčů, které se neindexují	107
Automatické generování dotazů	110
Popis pohledů datového slovníku v Oraclu	112

6. Práce s řetězcí **115**

Procházení řetězcem	115
Ohraničení znaků řetězce jednoduchými uvozovkami	117
Četnost výskytu znaku v řetězci	118

Odstranění nechtěných znaků z řetězce	118
Rozdělení řetězce na čísla a znaky	120
Nalezení alfanumerických řetězců	124
Extrakce iniciál ze jména	128
Třídění podle částí řetězce	131
Třídění podle čísla v řetězci	132
Generování dělených výpisů řádků tabulky	138
Konvertování dělených dat na vícepoložkovou podmínku klauzule IN	143
Seřazení znaků řetězce podle abecedy	148
Identifikace řetězců, které lze považovat za čísla	153
Extrakce něho odděleného podřetězce	159
Rozdělení IP adresy	165

7. Práce s čísly

169

Výpočet průměru	169
Nalezení nejvyšší a nejnižší hodnoty záznamu ve sloupci	171
Součet hodnot všech záznamů ve sloupci	172
Zjištění počtu řádků v tabulce	174
Zjištění počtu záznamů ve sloupci	175
Průběžný součet	176
Průběžný součin	179
Průběžný rozdíl	181
Modus	182
Medián	184
Výpočet procentuálního podílu	188
Agregace sloupců, ve kterých se vyskytují prázdné záznamy	190
Výpočet průměru bez nejvyšších a nejnižších hodnot	191
Konverze alfanumerických řetězců na čísla	193
Změna hodnot, které jsou součástí průběžného součtu	195

8. Výpočty s kalendářními daty

197

Přičítání a odčítání dnů, měsíců a roků	197
Počet dnů mezi dvěma kalendářními daty	199
Počet pracovních dnů mezi dvěma kalendářními daty	201
Počet měsíců a roků mezi dvěma kalendářními daty	205
Počet hodin, minut a sekund mezi dvěma kalendářními daty	208
Počet jednotlivých dnů týdne v daném kalendářním roce	209
Určení časového intervalu mezi dvěma záznamy o kalendářních datech	219

9. Operace s kalendářními daty **225**

Přestupný rok	225
Počet dnů roku	232
Extrakce časových jednotek z kalendářního data	234
První a poslední den měsíce	236
Určení kalendářních dat daného dne týdne v průběhu celého roku	239
Datum prvního a posledního specifického dne týdne v měsíci	245
Kalendář	251
Kalendářní data prvních a posledních dnů jednotlivých čtvrtletí daného roku	268
Kalendářní datum prvního a posledního dne daného čtvrtletí	273
Doplňování chybějících kalendářních dat	279
Vyhledávání založené na určité časové jednotce	288
Porovnávání hodnot záznamů na základě specifické části kalendářního data	289
Identifikace časových intervalů, které do sebe zasahují	292

10. Práce s intervaly **299**

Lokalizace spojitého intervalu hodnot záznamů	299
Nalezení rozdílů mezi řádky ve stejné skupině nebo partition	304
Lokalizace počáteční a koncové hodnoty záznamu spojitého intervalu	312
Doplňování chybějících hodnot záznamů intervalu	316
Generování spojitých číselných hodnot záznamů	320

11. Pokročilé vyhledávání **325**

Stránkování výsledné množiny	325
Vynechání n řádků tabulky	328
Logický operátor OR a vnější spojení	330
Nalezení řádků s obrácenými hodnotami záznamů	333
Výběr n horních záznamů	334
Nalezení nejvyšších a nejnižších hodnot záznamů	336
Analýza následujících řádků	337
Sousední hodnoty záznamů	340
Očíslování výsledků	343
Skrytí duplikátů	344
Závislé hodnoty záznamů	346
Generování jednoduchých prognóz	353

12. Analytické funkce a tvorba reportů **361**

Přeložení výsledné množiny dotazu do jednoho řádku	361
Přeložení výsledné množiny dotazu do více řádků	363
Obrácené přeložení výsledné množiny dotazu	370
Obrácené přeložení výsledné množiny dotazu do jednoho sloupce	372

Skrytí opakujících se hodnot záznamů ve výsledné množině	375
Přeložení výsledné množiny dotazu umožňující výpočty s hodnotami záznamů různých řádků	378
Tvorba datových skupin dané velikosti	379
Tvorba daného počtu datových skupin	383
Generování horizontálních histogramů	387
Generování vertikálních histogramů	389
Vrácení sloupců, které nejsou na seznamu parametrů klauzule GROUP BY	392
Jednoduché dílčí součty	397
Dílčí součty všech možných kombinací hodnot záznamů	401
Identifikace řádků, které neobsahují dílčí součty	410
Označení řádků prostřednictvím příkazu CASE	411
Vygenerování rozptýlené matice	413
Seskupení řádků podle časových údajů	414
Simultánní agregace různých skupin nebo partition	418
Agregace pohyblivého intervalu hodnot záznamů	420
Přeložení výsledné množiny dotazu obsahující dílčí součty	426

13. Hierarchické dotazy **431**

Vyjádření vztahu rodič-potomek	432
Vyjádření vztahu potomek-rodič-prarodič	435
Zobrazení hierarchických vztahů v celé tabulce	440
Nalezení všech potomků daného rodiče	447
Klasifikace řádků jako koncových uzlů, větví a/nebo kořenových uzlů	450

14. Doplnující příklady **459**

Přeložení výsledné množiny dotazu prostřednictvím klauzule PIVOT (SQL Server)	459
Zpětné přeložení výsledné množiny dotazu prostřednictvím klauzule UNPIVOT (SQL Server)	461
Přeložení výsledné množiny dotazu prostřednictvím klauzule MODEL (Oracle)	462
Extrakce podřetězců z nepravidelně děleného řetězce	466
Počet dnů roku (Oracle)	468
Práce s alfanumerickými řetězci	470
Převod celých čísel do dvojkové soustavy (Oracle)	472
Přeložení očíslovaných řádků výsledné množiny dotazu do sloupců	475
Vložení názvu sloupce do dvojnásobně přeložené výsledné množiny dotazu	478
Konverze dotazu vnořeného do příkazu SELECT na složený vnořený dotaz (Oracle)	489
Rozdělení serializovaných dat do řádků	491
Výpočet procentuálního podílu	495
Generování výstupu ve formátu CSV (Oracle)	497

Nalezení řetězců, které nejsou v daném formátu (Oracle)	501
Transformace dat prostřednictvím řádkového pohledu	504
Testování přítomnosti specifické hodnoty v datové skupině	505

A. Analytické funkce **509**

Seskupování	509
Analytické funkce	519

B. Návrat k Rozenshteinovi **533**

Tabulky dat v Rozenshteinově knize	533
Problémy, jejichž řešení vyžaduje použití negace	535
Problémy, jejichž řešení vyžaduje použití výrazu menší nebo rovno	542
Problémy, jejichž řešení vyžaduje použití výrazu větší nebo rovno	548
Problémy, jejichž řešení vyžaduje použití výrazu přesně (je rovno)	552
Problémy, jejichž řešení vyžaduje použití výrazů jakýkoliv a všichni	558

Rejstřík **565**

O autorovi

Anthony Molinaro pracuje jako vývojář databází ve firmě Wireless Generation, Inc. a má mnohaleté zkušenosti s podporou ostatních vývojářů. SQL je Anthonyho vášní a je známý tím, že když má některý ze zákazníků firmy problém se složitým SQL dotazem, Anthony ho vždy na místě vyřeší. Dobře rozumí relační teorii, jeho texty se dobře čtou a má devět let zkušeností s každodenním řešením náročných problémů v jazyce SQL. Anthony je také velmi dobře obeznámen s novými nástroji jazyka SQL, jako jsou například analytické funkce, které jsou jednou z novinek současného SQL standardu.

O obálce

Konečná podoba této knihy je výsledkem reakcí na ohlasy čtenářů, našich vlastních experimentů a zpětné vazby z distribučních sítí. Charakteristické přebaly našich knih jsou v souladu s naším charakteristickým přístupem k technickým tématům, který vnáší život a svéráz do jinak suchopárných oblastí lidského poznání.

Zvíře na přebalu knihy je agama. Tyto ještěrky náleží do čeledi agamovitých, která zahrnuje více než 300 různých živočišných druhů. Živočichové náležející do čeledi agamovitých žijí v Africe, Asii, Austrálii a v jižní Evropě. Jejich charakteristickým rysem jsou dlouhé nohy a některé druhy umí také měnit barvu. Ještěrky z čeledi agamovitých však nemají schopnost regenerace ocasu, v případě, že o něj nějakým způsobem přijdou, jako některé jiné ještěrky. Žijí v nejrůznějším prostředí, včetně pouští a vlhkých tropických pralesů.

Některé druhy agamidů jsou populární jako domácí mazlíčci. Jedním z nich je také agama bradatá (pogona barbata). Tato agama je velmi klidná a působí velmi nezvyklým dojmem. Dorůstá do asi půlmetrové délky, je považována za jednu z obřích ještěrek (ačkoliv půl metru není příliš) a vyžaduje hodně životního prostoru. Samečci obvykle brání svoje teritorium a i když je tato ještěrka společenským zvířetem, žije-li příliš mnoho jedinců na jednom místě, jsou nervózní, když se nemají kam schovat. Mohou pak spolu začít bojovat, zranit se a přijít o nohu nebo o ocas, či ztratit chuť k jídlu.

Agama bradatá má trojúhelníkový tvar hlavy a na bradě jí vyrůstá mnoho ostnů. Její brada je proto velmi nápadná a dala ještěrce i její jméno. Když je tato agama v ohrožení, otevírá tlamu a vystavuje na odív svoje ostny, aby vystrašila případného predátora nebo ostatní ještěrky. Navíc se ještě napřimuje, aby vypadala větší. Pokud žije v zajetí, může si zvyknout na svého chovatele a přestat vystavovat svoje ostny.

Ačkoliv agama bradatá pochází původně z Austrálie, jsou příslušníci tohoto živočišného druhu, které můžete koupit v USA, potomky zvířat dovezených z Evropy. Je to důsledek přísných australských zákonů, které omezují vývoz australské fauny.

Dalším zajímavým živočišným druhem z čeledi agamovitých je dráček létavý (draco volans). Tato ještěrka měří okolo třiceti centimetrů a má dlouhé hubené tělo s kapsami kůže na žebrech. Teritoriem jednoho samečka jsou dva až tři stromy. Na každém z těchto stromů pak žije jedna až tři samičky. Aby se mohl sameček přesunovat z jednoho stromu na druhý, roztahuje kožovité kapsy a plachtí vzduchem. Většinou však nelétá když prší nebo fouká vítr. Kožovité kapsy také někdy roztahuje, když je v ohrožení, aby vypadal větší.

Dalším zajímavým druhem z čeledi agamovitých je agama osadní (agama agama), která se vyskytuje v subsaharské Africe. Tito živočichové obvykle žijí v deseti až dvacetičlenných skupinách, které vede starší sameček. V noci jsou tmavě hnědí, ale za úsvitu se barva jejich těla mění na světle modrou

a jejich hlavy na svítivě oranžovou. Barva jejich kůže se mění podle jejich nálady a když spolu samečci bojují, mají tmavě hnědou hlavu a bílé tečky po celém těle.

Redaktorem výroby této knihy byl Darren Kelly. Pomocným redaktorem byl Kenneth Kimball a korekturu provedla Karmyn Guthrie. Výrobu provedla firma nSight, Inc. Jamie Peppard and Genevieve d'Entremont provedli kontrolu kvality. Jansen Fernald zajistil podporu výroby a Beth Palmer napsala Index.

Přebal navrhla Karen Montgomery v souladu s designem série, jehož autorem je Edie Freeman. Obrázek na přebalu je rytinou z 19. století, která pochází z Doverského archivu ilustrací. Návrh přebalu byl vytvořen v programu Adobe InDesign CS a použitý font je Adobe ITC Garamond.

Grafickou úpravu textu provedl David Futato. Kniha byla převedena do programu FrameMaker 5.5.6 Keithem Fahlgrenem s pomocí nástroje pro konverzi, který využívá technologie Perl a XML, a jehož autory jsou Erik Ray, Jason McIntosh, Neil Walls a Mike Sierra. Font použitý pro text knihy je linotype Birka, pro nadpisy Adobe Myriad Condensed a pro zdrojové kódy LucasFont's TheSans Mono Condensed. Knihu ilustrovali Robert Romano, Jessamyn Read a Lesley Borash v programech Macromedia FreeHand MX a Adobe Photoshop CS. Obrázky označující upozornění a tipy nakreslil Christopher Bing. Tuto tiráž napsal Jansen Fernald.

Předmluva

Ve světě databází je SQL jazykem s velkým J. Ať už relační databáze vyvíjíte, nebo z nich získáváte výpisy dat, vaše schopnost data do databáze vkládat a vracet je z ní zpět závisí hlavně na tom, do jaké míry ovládáte SQL. Mnoho programátorů však používá SQL pouze povrchním způsobem, aniž si byli vědomi možností, které jim doopravdy nabízí. Cílem této knihy je tento stav změnit, otevřít vám oči, ukázat pravou sílu SQL a představit, co je v něm doopravdy možné.

Knih, kterou držíte v ruce, je knihou návodů. Jedná se o sbírku řešení běžných problémů, která by vám měla usnadnit vaši každodenní práci. Jednotlivé návody jsou rozděleny do kapitol s ohledem na příbuznost probíraných témat. Když se setkáte s novým problémem, na který jste při své práci ještě nenarazili, najdete kapitolu, jež by mohla obsahovat jeho řešení, projděte si návody, které obsahuje, a měli byste najít buď samotné řešení nebo alespoň inspiraci.

Tato kniha obsahuje více než 150 návodů na zhruba 600 stránkách textu a přesto se jedná o pouhý úvod do SQL. Počet řešení vašich každodenních problémů závisí pouze na počtu těchto problémů. Proto v knize nenajdete řešení úplně všech problémů, se kterými se můžete setkat. Napsat takovou knihu je totiž nemožné. Najdete v ní však rozbor mnoha běžných problémů a jejich řešení. Když se naučíte pracovat s technikami, které se využívají při řešení problémů, jež jsou v knize uvedené, budete je moci následně využít při řešení problémů jiných, se kterými jsem se nikdy nesetkal, a proto mě ani nenapadlo je v knize uvést.



Spolu s mým nakladatelem neustále hledáme nové návody k řešení problémů v SQL, které by stálo za to do knihy zařadit. Podaří-li se vám nějaký problém chytře vyřešit, měli byste se o to podělit s ostatními. Pošlete nám jeho řešení, abychom ho mohli případně zařadit do dalšího vydání této knihy. Kontakt na nás najdete v sekci Otázky a komentáře.

Proč jsem napsal tuto knihu

Dotazy, dotazy, dotazy. Od začátku bylo mým cílem napsat spíše knihu Dotazy krok za krokem, než SQL Kuchařka programátora. Doufal jsem, že si čtenáři osvojí nejrůznější techniky, které jsem použil při psaní těchto dotazů, a následně je použijí při řešení svých specifických problémů. Zaměřil jsem se proto na to, aby kniha pokrývala všechny možné druhy dotazů, od jednodušších až po složitější. Rád bych se se čtenáři podělil o nejrůznější techniky psaní dotazů v SQL, se kterými jsem se ve svém profesním životě setkal. Rád bych, kdybyste vy, čtenáři, těmto technikám porozuměli, poučili se z nich a případně je ještě vylepšili; tento cyklus je užitečný pro nás pro všechny. Získání požadovaných dat z databáze vypadá velmi jednoduše, avšak ve světě informačních technologií je nezbytné, aby tato operace probíhala co nejefektivnějším způsobem. Techniky efektivního výběru

dat bychom měli všichni sdílet, abychom všichni mohli odvádět dobrou práci a společně se zlepšovat v tom, co děláme.

Vzpomeňte si na obrovský přínos, kterým byla pro matematiku teorie množin Georga Cantora, jenž si jako první uvědomil neuvěřitelný potenciál práce s množinami (oproti výzkumu jejich jednotlivých prvků). Mnozí jeho vrstevníci jeho práci nejprve nepřijali. Časem však nebyla tato teorie pouze uznána, ale dokonce je v současné době považována za základ moderní matematiky. Avšak tato teorie ve své současné podobě není dílem samotného Georga Cantora, ale spíše Ernsta Zermela, Gottloba Fregeho, Abrahama Fraenkela, Thoralfa Skolema, Kurta Gödela a Johna von Neumanna, kteří rozvinuli jeho myšlenky. Tato spolupráce nejenom umožnila všem z nich danou problematiku lépe pochopit, ale zároveň díky ní bylo možné teorii množin podstatně vylepšit.

Co si tato kniha klade za cíl

Cílem této knihy je nastínit vám, čtenářům, čeho je možné v SQL dosáhnout a řešení kterých problémů jsou považována za doménu SQL. Jazyk SQL prošel v posledním desetiletí dlouhým vývojem. Problémy, které se dříve obvykle řešily prostřednictvím procedurálních programovacích jazyků, jako je C, nebo jazyk Java, lze v současné době řešit přímo v SQL, ale mnoho vývojářů o tom stále neví. Díky této knize byste měli získat lepší představu o možnostech řešení těchto problémů prostřednictvím SQL.

Před tím, než špatně pochopíte to, co jste si zrovna přečetli, mi dovoluňte zdůraznit, že jsem zastáncem hesla „co není rozbité, nespravuj“. Představte si například, že řešíte konkrétní problém v práci tím způsobem, že využíváte SQL k výběru dat z databáze a následně tato data zpracováváte s ohledem na komplexní obchodní logiku vaší firmy podle daných požadavků prostřednictvím jiného nástroje (jazyka), než je SQL. V případě, že tento postup funguje a je dostatečně efektivní, je to skvělé. V žádném případě vám nechci říkat, že byste měli vaše data zpracovávat pouze s pomocí SQL; pouze vás chci poprosit, abyste byli otevření novým věcem, a uvědomili si, že SQL, ve kterém jste programovali v roce 1995, není stejně jako SQL v současnosti, protože jeho možnosti jsou dnes mnohem větší.

Komu je tato kniha určena

Tato kniha je jedinečná, protože je určena širokému spektru potenciálních čtenářů, aniž by to bylo na úkor její odbornosti. Obsahuje jednoduché i komplexní způsoby řešení daných problémů a v případě, že není možné použít jediný způsob řešení pro všechny databázové systémy, které jsou jejím předmětem, najdete vždy v příslušné sekci způsob řešení daného problému pro každý z nich. Spektrum čtenářů, pro které je tato kniha určena, je tedy skutečně široké:

Začátečníci

Možná jste si zrovna koupili knihu, která je úvodem do SQL, nebo jste právě nastoupili do prvního semestru kursu, jenž se věnuje databázím, a chcete doplnit vaše nové znalosti nějakými příklady řešení opravdových problémů. Možná jste zrovna viděli dotaz, který záhadným způsobem přeložil řádky tabulky do sloupců nebo rozdělil řetězec serializovaných dat do výsledné množiny. Návody, které najdete v této knize, pak tyto zdánlivě nepochopitelné dotazy vysvětlují.

Programátoři jiných jazyků

Možná umíte programovat v jiném jazyce a teď po vás v práci někdo chce, abyste obsluhovali komplexní SQL kód, který napsal někdo jiný. V návodech, které najdete v posledních kapitolách knihy, jsou komplexní dotazy rozloženy na jednotlivé součásti a máte tak možnost postupně porozumět kódu, jež jste možná po někom zdělili.

Středně pokročilí každodenní uživatelé

Pro středně pokročilého vývojáře je tato kniha pokladem (to je možná příliš silné slovo; omluvte lásku autora k tématu, o kterém píše). Takže: nějakou dobu už s SQL pracujete, ale stále jste nenašli cestu k analytickým funkcím. Rádi byste to však zkusili. Dny ukládání průběžných výsledků dotazů do dočasných tabulek jsou u konce; díky analytickým funkcím dosáhnete stejných výsledků prostřednictvím jediného dotazu! V tuto chvíli bych opět rád zdůraznil, že není mým cílem vnucovat mé postupy zkušeným programátorům. Místo toho pokládejte tuto knihu za nástroj, jehož prostřednictvím se můžete naučit něco nového, pokud jste dosud neměli čas seznámit se s některými novými postupy při psaní dotazů.

SQL odborníci

Bezpochyby jste se již někdy v minulosti s návody uvedenými v této knize setkali, a určitě máte i vlastní způsoby řešení daných problémů. Proč je pro vás potom tato kniha přínosem? Možná jste celý život pracovali s jedním ze systémů správy databází a teď byste se rádi naučili Oracle. Možná, že jste ještě nikdy nepoužili MySQL, a nebo vás zajímá, jak vypadá řešení stejného problému pro PostgreSQL. Předmětem této knihy je více systémů pro správu databází a v každé kapitole najdete způsoby řešení daných problémů pro jednotlivé systémy. To je šance k rozšíření vašich vědomostí.

Jak používat tuto knihu

Pročtete si důkladně celou předmluvu, protože obsahuje důležité informace, které byste mohli postrádat v případě, že byste četli pouze jednotlivé kapitoly nebo recepty. V sekci Systémy správy relačních databází a jejich verze se dozvíte, o kterých z těchto systémů kniha pojednává. Věnujte zvláštní pozornost sekci Tabulky použité v této knize, abyste se dobře seznámili s tabulkami, se kterými se pracuje ve většině příkladů. Důležitá je i sekce Použité konvence. Všechny tyto sekce následují.

Pamatujte, že tato kniha je knihou návodů, sbírkou zdrojových kódů, které by vám měly sloužit jako vodítko při řešení podobných (stejných) problémů. Nepokoušejte se s pomocí této knihy SQL naučit, nebo alespoň ne od základů. Tato kniha by měla sloužit jako doplněk komplexní učebnice SQL a ne jako její náhrada. Věnujte také pozornost níže uvedeným doporučením, která vám pomohou používat knihu efektivnějším způsobem:

- Návody uvedené v této knize využívají specifické funkce jednotlivých databázových systémů. Hodí se mít po ruce knihu *SQL Pocket Guide*, kterou napsal Jonathan Gennick, a v níž najdete jejich přesný popis.
- Pokud jste se ještě nikdy nesetkali s analytickými funkcemi, nebo jestli máte problém porozumět dotazům, ve kterých je potřeba seskupovat, přečtete si nejprve přílohu A, kde najdete definici skupiny a základní popis analytických funkcí. Analytické funkce jsou jedním z nejdůležitějších nových nástrojů, které byly do SQL přidány během posledních deseti let.

- **Používejte zdravý rozum!** Uvědomte si, že není možné napsat knihu, jež by uváděla řešení každého problému, se kterým se můžete setkat. Místo toho považujte návody v knize za vzor nebo pomůcku, díky které se můžete naučit používat techniky, jež následně využijete při řešení vašich specifických problémů. Počítá se s tím, že si někdy budete říkat: „Skvělé, tenhle návod funguje, když pracuji s touto množinou dat, ale moje množina dat se liší, a proto nemůžu problém řešit úplně stejně.“ V takovém případě se pokuste najít něco, co mají vaše data s daty v knize společného. Rozložte dotaz v knize na základní stavební kameny a postupně ho upravujte do komplexnější podoby. Všechny dotazy začínají příkazem SELECT a klauzulí FROM, takže ve své nejjednodušší formě jsou všechny stejné. Když je budete postupně upravovat krok za krokem do komplexnější podoby, přidávat vždy pouze jednu funkci, tvořit vždy pouze jedno nové spojení, nebudete mít pouze možnost sledovat, jak jednotlivá rozšíření mění podobu výsledné množiny dotazu, ale zároveň uvidíte, jak se dotaz v knize liší od dotazu, který vyřeší váš specifický problém. Pak již stačí dotaz upravit tak, aby správně pracoval i s vašimi specifickými daty.
- **Zkoušejte, zkoušejte, zkoušejte.** Každá vaše tabulka má bezpochyby více řádků, než je 14 řádků vzorové tabulky EMP, takže musíte vždy vyzkoušet, jestli uvedené řešení lze použít také pro vaše data. Nikdo nezná přesnou podobu vašich tabulek, neví, které z jejich sloupců mají indexy, a jaké vztahy reprezentuje datové schéma vaší databáze. Neměli byste proto slepě implementovat řešení v knize do vašeho zdrojového kódu, aniž byste jim plně porozuměli a bylo vám jasné, jak se budou chovat, pokud jejich vstupem budou vaše data.
- **Nebojte se experimentovat a buďte kreativní!** Nebojte se použít jiné způsoby řešení, než ty, které najdete v knize. Cílem této knihy je představit co nejvíce různých specifických funkcí, kterými disponují jednotlivé databázové systémy. To však neznamená, že neexistuje ještě mnoho jiných funkcí, které se hodí k řešení daného problému úplně stejně, jako ty, jež najdete v jednotlivých návodech.
- **Nové neznamená automaticky lepší.** Pokud nevyužíváte některé z novějších nástrojů SQL (například analytické funkce), neznamená to ještě, že váš zdrojový kód není tak efektivní, jak by mohl být. V mnoha různých případech je tradiční způsob řešení stejně dobrý, nebo dokonce lepší, než způsob využívající nové postupy nebo nástroje. Myslete na to zejména při četbě přílohy B, Návrat k Rozensteinovi. Také byste neměli po přečtení této knihy nabýt dojmu, že musíte přepsat všechny zdrojové kódy, které jste dosud napsali. Místo toho si jenom uvědomte, že SQL disponuje mnoha novými a velmi efektivními nástroji, kterými nedisponoval před deseti lety a jež stojí za to se naučit používat.
- **Pokud budete mít při četbě sekce Řešení problému pocit, že dotaz nemůžete pochopit, nenechte se tím odradit.** Každý dotaz je v sekci Rozbor řešení rozložen na základní kameny a najdete zde i výsledné množiny jeho jednotlivých součástí. Možná vám celý princip řešení nebude jasný na první pohled, ale když si pozorně pročtete jeho rozbor, uvidíte, že i složité vypadající dotazy není tak těžké pochopit.
- **Když je to třeba, programujte defenzivně.** Ve snaze udělat dotazy čitelnějšími je z nich vyjmuto mnoho obranných mechanismů. Představte si například dotaz, který určuje průběžný součet platů všech zaměstnanců. Může se jednat o případ, kdy jste definovali hodnoty ve sloupci jako typ VARCHAR a máte proto (bohužel) v jednom poli uložená zároveň čísla i řetězce znaků. Návod pro výpočet průběžného součtu uvedený v této knize s takovým případem nepočítá (a selže, protože funkce SUM není určená ke zpracování znakových řetězců). Takže pokud máte tento problém (záznamy v daném sloupci jsou tohoto typu), budete ho muset nějak vyřešit nebo (pokud to je možné) upravit příslušná data, protože návod uvedený v knize s podob-

nou situaci nepočítá. Snažte se proto pochopit hlavně princip řešení daného problému; řešení obdobných problémů je pak již triviální.

- Opakování je matka moudrosti. Návodům, které obsahuje tato kniha, nejlépe porozumíte, když si sednete a naprogramujete je. Pokud jde o programování, je četba odborných textů rozhodně důležitá, ale programování samotné je ještě lepší. V knize se dočtete, proč je třeba některé problémy řešit odpovídajícím způsobem, ale sami budete obdobné problémy umět vyřešit pouze v případě, že si uvedené dotazy vyzkoušíte v praxi.

Mnoho příkladů, jejichž řešením se zabývá tato kniha, je smyšlených. Avšak samotné problémy, které je potřeba vyřešit, smyšlené nejsou, ale vycházejí z praxe. Přesto je však potřeba mít na paměti, že tyto příklady se týkají pouze malých množin tabulek, které obsahují data o zaměstnancích. Je to proto, abyste se ve zdrojových datech lépe orientovali a mohli věnovat plnou pozornost řešení problémů v jednotlivých receptech. Možná si při rozboru některého z problémů budete říkat, že obdobnou operaci nikdy nebudete muset provádět s daty o zaměstnancích. Pokuste se však nebrat ohledy na konkrétní charakter zdrojových dat příkladů a zaměřte se hlavně na uvedený způsob řešení daného problému. Důležitá je hlavně použitá technika. Já i mí kolegyně tyto techniky používáme denně.

Co v této knize nenajdete

Kvůli časovým možnostem a objemu není možné, aby jediná kniha obsahovala řešení všech problémů, se kterými se můžete setkat. V následujícím výčtu najdete témata, kterými se kniha nezabývá:

Definice dat

Žádná část knihy není věnována tématům, jako je tvorba indexů, přidávání omezujících podmínek tabulky a import dat. Řešení podobných problémů obvykle vyžaduje znalost syntaxe, která je velmi specifická pro každý ze systémů, takže je nejlepší podívat se do manuálu příslušného systému. Navíc tyto úkoly nepředstavují neřešitelné oříšky, kvůli kterým by si někdo koupil knihu. Ve čtvrté kapitole knihy přesto najdete návody k řešení běžných problémů tohoto typu, jakými jsou například vkládání, aktualizace a mazání dat.

XML

Podle mého názoru rozbor řešení problémů v XML do knihy o SQL nepatří. Ukládání XML dokumentů do relačních databází je stále populárnější a každý systém správy relačních databází má svá vlastní rozšíření a nástroje pro manipulaci s tímto typem dat. Tato manipulace často vyžaduje zdrojový kód, který je procedurální, a proto leží mimo pole působnosti této knihy. Nejnovější technologie typu XQuery představují úplně odlišné téma a měly by jim být věnovány samostatné knihy.

Objektově orientovaná rozšíření SQL

Dokud nebude vyvinut použitelnější jazyk pro práci s objekty, jsem proti používání objektově orientovaných nástrojů a návrhů v relačních databázích. V současné době se objektově orientované nástroje, které nabízejí některé ze systémů, hodí spíše k procedurálnímu programování, než pro práci s množinami, pro niž byl jazyk SQL navržen.

Teoretická diskuse

V této knize nenajdete argumenty svědčící pro nebo proti relační podstatě SQL, ani diskusi o tom, zda je správné používat prázdné hodnoty záznamů. Tento druh teoretické diskuse je jistě důležitý, ale ne v knize, jejímž prvořadým cílem je řešení každodenních problémů. Abyste si

uměli poradit s problémy, se kterými se můžete setkat při každodenní praxi, musíte umět využívat nástroje, jež máte k dispozici. Musíte pracovat s tím co máte, a ne s tím, co byste mít chtěli.



Jestli se chcete dozvědět víc o teorii, dobrý zdroj informací představuje kterákoliv z knih Chrise Deata. Můžete si třeba koupit jeho poslední knihu *Database in Depth*.

Srovnání výkonu jednotlivých systémů

Tato kniha obsahuje řešení problémů pro pět různých databázových systémů. Odpověď na otázku, které z řešení je nejlepší nebo nejrychlejší, chce znát každý. Za tímto účelem každý z výrobců jednotlivých databázových systémů poskytuje dostatek pádných argumentů, avšak není mým cílem zabývat se jimi v rámci této knihy.

ANSI syntaxe

Autoři mnoha textů o SQL se stydí používat specifické funkce jednotlivých databázových systémů. Tato kniha naopak klade důraz na jejich využití v maximální možné míře. Není mým cílem psát spletité neefektivní dotazy jenom proto, aby je bylo možné použít ve stejné podobě pro každý ze systémů. Nikdy mi nikdo v práci nezakazoval používat specifické funkce systému, se kterým jsem pracoval. Když za tyto funkce platíte, proč byste je neměli používat?

Rozšíření jednotlivých systémů mají svůj význam a často nabízejí lepší výkon, než standardní SQL. Dáváte-li přednost řešení problémů pouze v ANSI syntaxi, je to vaše volba. Jak jsem již zmínil výše, tato kniha si neklade za cíl donutit vás přepsat všechny zdrojové kódy, které jste až dosud napsali. Pokud jsou všechny vaše kódy v ANSI syntaxi a splňují vaše očekávání, je to skvělé. Když na to přijde, všichni chodíme do práce, platíme účty a chceme být doma z práce včas a užít si po svém zbytek dne. Neříkám, že ANSI syntaxe je špatná. Dělejte, co je pro vás nejlepší. Chci jenom říci, že jestli hledáte řešení vašeho problému v ANSI syntaxi, měli byste je hledat někde jinde.

Tradiční způsoby řešení

Návody uvedené v této knize pracují s nejnovějšími nástroji, které byly k dispozici v době, kdy byla tato kniha napsána. Pracujete-li se staršími verzemi databázových systémů, o kterých tato kniha pojednává, mnohá řešení problémů zde uvedená v nich nebudou fungovat. Technologie nestojí na místě a to byste neměli ani vy. Hledáte-li řešení těchto problémů pro starší verze databázových systémů, je vám k dispozici mnoho starších textů, které obsahují návody k řešení stejných problémů, jakými se zabývá tato kniha.

Struktura knihy

Knihu obsahuje čtrnáct kapitol a dvě přílohy.

- 1. kapitola, *Přístup k záznamům*, představuje ty nejjednodušší typy dotazů. Příklady se věnují tématům, jako je využití klauzule WHERE k omezení řádků výsledné množiny, přiřazení aliasu sloupcům, používání řádkových pohledů, které odkazují na sloupce s aliasem, používání jednoduché podmínkové logiky, omezení počtu řádků výsledné množiny, návrat náhodných záznamů a lokalizace prázdných hodnot záznamů. Většina příkladů je velmi jednoduchá, ale některé z nich se později objevují i ve složitějších návodech, a proto byste si měli tuto kapitolu přečíst, pokud s SQL začínáte, nebo jste se s dotazy uvedenými v této kapitole ještě nikdy nasetkali.

- 2. kapitola, *Třídění výsledků dotazu*, obsahuje návody pro třídění výsledků vašich dotazů. V této kapitole se seznámíte s klauzulí ORDER BY a naučíte se jejím prostřednictvím třídit výsledky vašich dotazů. Od nejjednodušších příkladů, jako je třídění hodnot záznamů v jednom sloupci, budete postupovat k těm složitějším, jako je třídění podle podřetězců, nebo třídění podle podmínkové logiky.
- 3. kapitola, *Práce s více tabulkami*, představuje techniky, s jejichž pomocí lze využívat data z více tabulek najednou. Pokud s SQL začínáte, nebo si nejste úplně jistí v tvorbě spojení, měli byste si tuto kapitolu dobře přečíst před tím, než začnete číst kapitolu 5, nebo později. Spojování tabulek je jednou z nejdůležitějších technik SQL, kterou musíte ovládnout, abyste dokázali psát efektivní dotazy. Příklady v této kapitole se věnují tvorbě vnitřních i vnějších spojení, identifikaci kartézského součinu, základním operacím s množinami a vlivu vytvořených spojení na výstup agregačních funkcí.
- 4. kapitola, *Vkládání, aktualizace, mazání*, představuje způsoby, jakými lze data do databáze vkládat, aktualizovat je a mazat. Většina příkladů v této kapitole je velmi přímočará (možná dokonce poněkud bez fantazie). Nicméně jsou tyto znalosti, jako například vkládání nových řádků z jedné tabulky do druhé, využití korelovaných vnořených dotazů při aktualizaci, pochopení významu prázdných hodnot záznamů a znalost nových nástrojů, jako je možnost vkládat data do více tabulek najednou a využití příkazu MERGE, velmi užitečné.
- 5. kapitola, *Dotazy na metadata*, se věnuje dotazům, jejichž prostřednictvím můžete získat informace o samotné databázi. Často je velmi užitečné znát indexy, omezení a tabulky vaší databáze. Jednoduché návody uvedené v 5. kapitole vám představí způsoby, jakými lze informace o datovém schématu vaší databáze získat. Navíc se v této kapitole naučíte generovat zdrojový kód prostřednictvím samotného SQL.
- 6. kapitola, *Práce s řetězci*, obsahuje návody pro operace s řetězci. I když jazyk SQL nebyl za tímto účelem navržen, můžete dosáhnout dobrých výsledků, budete-li trochu kreativní (většinou ve využití kartézského součinu) a využijete-li širokou škálu funkcí pro zpracování řetězců, které nabízejí jednotlivé systémy. V této kapitole se kniha začíná stávat zajímavější. V některých z nejzajímavějších příkladů této kapitoly se využívají postupy, jejichž prostřednictvím lze zjistit, kolikrát se daný znak v řetězci vyskytuje, vytvořit dělený výpis řádků tabulky, konvertovat dělený výpis a řetězce na řádky a vyjmát číslice a písmena z alfanumerických řetězců.
- 7. kapitola, *Práce s čísly*, se věnuje běžným výpočtům. Problémy, jejichž řešení v této kapitole najdete, jsou velmi běžné. Naučíte se, jak jednoduchým způsobem s pomocí analytických funkcí vyřešit problémy s výpočty pohyblivých součtů a agregací. Naučíte se generovat průběžné součty, vypočítat průměr, medián, modus, procentuální podíl a naučíte se zahrnout do vašich agregací prázdné hodnoty záznamů.
- 8. kapitola, *Výpočty s kalendářními daty*, je první z dvojice kapitol věnovaných operacím s kalendářními daty. Znalost běžných výpočtů s kalendářními daty je zcela zásadní pro vaši každodenní práci. V 8. kapitole najdete návody, jak určit počet pracovních dnů mezi dvěma daty, vypočítat časový interval mezi dvěma kalendářními daty v různých časových jednotkách (dny, měsíce, roky) a jak vypočítat počet výskytů daného dne týdne v daném roce.
- 9. kapitola, *Operace s kalendářními daty*, je druhou z dvojice kapitol věnovaných operacím s kalendářními daty. Najdete v ní příklady nejobvyklejších operací, se kterými se setkáváte při práci každý den. Příklady se věnují tématům, jako je získání kalendářních dat všech dnů v daném roce, určení přestupných roků, určení prvního a posledního dne daného měsíce, tvorba kalendáře a doplňování chybějících kalendářních dat do časových intervalů.

- 10. kapitola, *Práce s intervaly*, obsahuje řešení problémů spojených s identifikací intervalů hodnot a s jejich generováním. Naučíte se automaticky generovat sekvence řádků, doplňovat do intervalu chybějící číselné hodnoty, lokalizovat počáteční a koncové hodnoty intervalu a hodnoty, které na sebe navazují.
- 11. kapitola, *Pokročilé vyhledávání*, představuje návody k řešení problémů týkajících se vyhledávání dat. Tyto návody nejsou o nic složitější, než návody v jiných kapitolách, přesto často mnoho vývojářů přistupuje k řešení obdobných problémů velmi neefektivním způsobem. Naučíte se lokalizovat vzájemně závislé hodnoty, stránkovat výslednou množinu dotazu, vynechávat z výsledné množiny specifické řádky, vynechávat řádky tabulky, hledat obrácené hodnoty záznamů, vracet n horních záznamů a číslovat řádky.
- 12. kapitola, *Analytické funkce a tvorba reportů*, se věnuje dotazům, které se obvykle využívají při tvorbě datových skladů nebo komplexních výpisů. Tato kapitola je podle mého názoru nejzajímavější z celé knihy. Naučíte se překládat řádky do sloupců a obráceně, vytvářet datové skupiny, generovat histogramy, určovat dílčí a celkové součty, agregovat pohyblivé intervaly hodnot záznamů a seskupovat řádky podle časových údajů.
- 13. kapitola, *Hierarchické dotazy*, pojednává o dotazech na data v určité hierarchii. Nezávisle na tom, v jaké formě vaše data jsou, po vás nakonec někdo bude chtít, abyste je zobrazili ve stromové struktuře, nebo s ohledem na vztah rodič-potomek, procházeli danou hierarchií od kořene až ke koncovému uzlu a nebo hierarchii stupňů po stupni zobrazili.
- 14. kapitola, *Doplňující příklady*, je sbírkou dotazů, které svou povahou neodpovídají tématu žádné z předchozích kapitol, avšak přesto jsou zajímavé a užitečné. Tato kapitola se liší od ostatních v tom, že se zaměřuje pouze na specifické způsoby řešení daných problémů v rámci jediného systému. Jedná se o jedinou kapitolu, ve které nejsou uvedena řešení pro všech pět systémů. Příčina je dvojitá. Za prvé by vás tato kapitola měla spíše pobavit a za druhé smysl některých návodů, jež jsou v této kapitole uvedeny, je pouze v tom, aby zdůraznil některé specifické funkce daného systému, jež nemají v ostatních systémech žádný ekvivalent (například klauzule PIVOT a UNPIVOT systému SQL Server, nebo klauzule MODEL systému Oracle). Přesto je možné některé z návodů snadno upravit tak, aby fungovaly i v jiných systémech, než pro jaké jsou určené.
- Příloha A, *Analytické funkce*, vám pomůže osvěžit si vaše znalosti analytických funkcí a také obsahuje důkladné pojednání o funkci skupin v SQL. Pro většinu z vás jsou analytické funkce něčím novým, takže tato příloha slouží zároveň jako stručný úvod do jejich problematiky. Všiml jsem si, že používání klauzule GROUP BY v dotazech působí mnoha vývojářům problémy. V této kapitole proto najdete přesnou definici skupiny a mnoho dotazů, které dokazují její správnost. Také se dozvíte, jaký efekt mají prázdné hodnoty záznamů na tvorbu skupin, agregací a partition. Na konci kapitoly pak najdete diskusi o podivuhodné ale velmi užitečné klauzuli analytických funkcí OVER (klauzule pro windowing nebo framing).
- Příloha B, *Návrat k Rozenshteinovi*, je poctou Davidu Rozenshteinovi, kterému vděčím za svoje vývojářské úspěchy, jichž jsem dosáhl v jazyce SQL. Rozenshteinova kniha *The Essence of SQL* (nakladatelství Coriolis), byla mou první knihou o SQL mimo školní skriptu, kterou jsem si koupil. Vědomosti, které jsem nabyl při její četbě, využívám dodnes. Tato kniha je úplně jiná, než všechny ostatní knihy o SQL, které jsem kdy četl, a jsem vděčný za to, že byla první knihou o SQL, již jsem si sám vybral. Příloha B se věnuje některým dotazům z knihy *The Essence of SQL* a představuje jejich alternativní podobu využívající analytické funkce (jež nebyly v době, kdy byla kniha *The Essence of SQL* napsána, k dispozici).

Jednotlivé platformy a jejich verze

SQL prochází nepřetržitým vývojem. Jednotliví výrobci neustále přidávají do svých systémů nové nástroje a funkce. Proto je třeba znát konkrétní verze systémů, o kterých pojednává tato kniha:

- DB2 v.8
- Oracle Database 10g (až na několik případů však lze uvedený způsob řešení daného problému vždy použít i ve verzích Oracle 8i Database a Oracle 9i Database)
- PostgreSQL 8
- SQL Server 2005
- MySQL 5



Poznámka českého vydavatele: Pro databázi Oracle se v knize používá zkrácený název. Například Oracle 10g místo Oracle Database 10g (podobně je tomu u ostatních verzí této databáze).

Použité tabulky

Většina příkladů pracuje se dvěma tabulkami, EMP a DEPT. EMP je jednoduchá tabulka, která má 14 řádků, a obsahuje pouze pole čísel, znakových řetězců a kalendářní dat. DEPT je jednoduchá tabulka, která má 4 řádky, a obsahuje pouze pole čísel a znakových řetězců. S těmito tabulkami se pracuje v mnoha starších textech o databázích a vztah mezi zaměstnanci a oddělením, ve kterém pracují, je snadno pochopitelný.

Zde je také vhodné zmínit, že většina způsobů řešení daných problémů, kromě několika z nich, využívá tyto tabulky. V žádném z příkladů pak nejsou tyto tabulky přizpůsobeny uvedenému způsobu řešení, tak, jak je tomu v mnoha jiných knihách a následkem čehož je uvedené řešení nepoužitelné v praxi.

Dále je třeba v tuto chvíli říci, že pokud to bylo alespoň trochu možné, pokusil jsem se použít způsob řešení, který je možné použít pro všech pět databázových systémů, o nichž pojednává tato kniha. V některých případech to však nebylo možné a někdy lze sice příslušné řešení použít pro více systémů, ale ne pro všechny z nich. Jediný způsob řešení často sdílí například systémy Oracle a DB2, což je dáno tím, že disponují obdobnými analytickými funkcemi. V případě, že je způsob řešení shodný, nebo alespoň téměř shodný pro více systémů najednou, vztahuje se na řešení daného problému pro všechny tyto systémy jeho jediný rozbor.

Tabulky EMP a DEPT obsahují tato data:

```
select * from emp;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-1980	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-1981	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-1981	1250	500	30
7566	JONES	MANAGER	7839	02-APR-1981	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-1981	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-1981	2850		30
7782	CLARK	MANAGER	7839	09-JUN-1981	2450		10
7788	SCOTT	ANALYST	7566	09-DEC-1982	3000		20
7839	KING	PRESIDENT		17-NOV-1981	5000		10

7844	TURNER	SALESMAN	7698	08-SEP-1981	1500	0	30
7876	ADAMS	CLERK	7788	12-JAN-1983	1100		20
7900	JAMES	CLERK	7698	03-DEC-1981	950		30
7902	FORD	ANALYST	7566	03-DEC-1981	3000		20
7934	MILLER	CLERK	7782	23-JAN-1982	1300		10

```
select * from dept;
```

```
DEPTNO DNAME          LOC
-----
10 ACCOUNTING      NEW YORK
20 RESEARCH        DALLAS
30 SALES            CHICAGO
40 OPERATIONS      BOSTON
```

Dále se při řešení jednotlivých příkladů využívají tyto kontingenční tabulky: T1, T10, T100 a T500. Protože se tyto tabulky využívají pouze při překládání, nepovažuji za důležité vymýšlet pro ně nějaké reprezentativní názvy. Číslo za písmenem T představuje počet řádků každé z těchto tabulek počínaje číslem 1. Pro ilustraci následuje výpis hodnot záznamů v tabulkách T1 a T10:

```
select id from t1;
```

```
      ID
-----
      1
```

```
select id from t10;
```

```
      ID
-----
      1
      2
      3
      4
      5
      6
      7
      8
      9
     10
```

Některé ze systémů také dovolují používat neúplné příkazy SELECT. Například můžete použít příkaz SELECT bez klauzule FROM. Nemám to zrovna v oblíbenosti, takže raději používám příkaz SELECT nad pomocnou tabulkou s jedním řádkem T1, než abych použil neúplný dotaz.

Jakékoliv jiné tabulky, které v knize najdete, jsou definovány pouze pro potřeby jednotlivých příkladů a kapitol a jejich definice bude v textu vždy uvedena na příslušném místě.

Konvence použité v knize

Úprava textu použitá v této knize se řídí několika konvencemi. Najděte si čas a seznamte se s nimi, budete se v textu lépe orientovat. Důležité jsou zejména zásady zápisu zdrojového kódu, protože není možné je rozebírat pokaždé na začátku každého návodu. Místo toho najdete všechny důležité konvence zde.

Typografické konvence

Při úpravě textu byly dodržovány tyto konvence:

VELKÁ PÍSMENA

Velká písmena označují klíčová slova jazyka SQL v běžném textu.

malá písmena

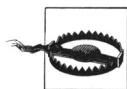
Malými písmeny jsou zapsány všechny zdrojové kódy. V jiných programovacích jazycích, jako jsou například C nebo Java, se většina klíčových slov zapisuje malými písmeny a připadá mi to mnohem čitelnější. Proto jsou všechny zdrojové kódy v knize zapsány malými písmeny.

malá písmena tučná

Označují zásah uživatele do zdrojového kódu.



Takto vypadají různé tipy, doporučení nebo obecné poznámky.



Takto vypadají různá varování nebo upozornění.

Konvence dodržované při zápisu zdrojového kódu

Při zápisu příkazů upřednostňuji malá písmena pro samotná klíčová slova i uživatelem definované identifikátory. Například:

```
select empno, ename  
from emp;
```

Možná raději zapisujete zdrojové kódy jiným způsobem. Mnoho programátorů například pro zápis příkazů používá písmena velká. Pište svůj zdrojový kód způsobem, který vám vyhovuje.

Ačkoliv jsou všechny zdrojové kódy zapsány malými písmeny, všechny identifikátory i klíčová slova v textu jsou zapsány písmeny velkými. Je to z toho důvodu, aby byly zřetelně označeny; například:

Předchozí dotaz představuje provedení příkazu SELECT nad tabulkou EMP.

Ačkoli tématem knihy je pět různých databázových systémů, rozhodl jsem se, že výsledné množiny jednotlivých dotazů budou v textu vždy ve stejné podobě:

```
EMPNO ENAME  
-----  
7369 SMITH  
7499 ALLEN  
...
```

Mnoho řešení využívá řádkové pohledy, nebo dotazy vnořené do klauzule FROM. ANSI syntaxe jazyka SQL vyžaduje, aby byly těmto pohledům přiřazeny alias. (Jediným systémem, který to nevyžaduje, je Oracle.) Za tímto účelem využívám aliasy, jako jsou X nebo Y:

```
select job, sal  
from (select job, max(sal) sal  
      from emp  
      group by job) x;
```

Všimněte si písmena X, které následuje za poslední kulatou závorkou. Toto písmeno je názvem tabulky, která je výsledkem dotazu vnořeného do klauzule FROM. Zatímco aliasy sloupců představují cenný nástroj, díky kterému je zdrojový kód lépe pochopitelný pro ostatní programátory, aliasy řádkových pohledů jsou (ve většině návodů v této knize) pouhou formalitou. Tyto aliasy jsou obvykle velmi triviální, jako například X, Y, Z, TMP1 a TMP2. V případech, kdy aliasy řádkových pohledů umožňují lepší porozumění zdrojovému kódu, jsou jejich názvy voleny smysluplně.

Jednotlivé řádky zdrojového kódu v sekci Řešení jsou obvykle očíslovány; například:

```
1 select ename
2   from emp
3  where deptno = 10
```

Čísla řádků nejsou součástí syntaxe; pouze umožňují odkazovat se na části dotazů v textu sekce Rozbor řešení.

Používání zdrojových kódů v knize

Tato kniha byla napsána proto, aby vám pomohla řešit problémy, se kterými se setkáváte v práci každý den. Obecně vzato můžete používat zdrojové kódy uvedené v knize ve vašich vlastních programech a jejich dokumentaci. Pokud nereprodukuje větší část zdrojového kódu z této knihy, nemusíte kontaktovat naše nakladatelství a žádat o povolení. Když například píšete program a chcete použít několik úryvků zdrojového kódu z knihy, povolení nepotřebujete. Když odpovídáte na něčí otázky a citujete při tom tuto knihu, není třeba, abyste kvůli tomu žádali o povolení. Chcete-li však použít větší část zdrojového kódu z knihy v dokumentaci vašeho programu, musíte požádat o povolení nakladatelství.

Uvádění zdrojů citací nevyžadujeme, ale oceňujeme. Jakákoliv citace obvykle sestává z názvu knihy, jména jejího autora a kódu ISBN. Například: SQL Cookbook, Anthony Molinaro, O'Reilly Media Inc., 0-596-00976-3.

Pokud máte pocit, že způsob, jakým používáte zdrojové kódy z této knihy, vyžaduje povolení nakladatelství, kontaktujte nás, prosím, na adrese permission@oreilly.com (adresa původního nakladatele).

Poznámka redakce českého vydání

Nakladatelství Computer Press, které pro vás tuto knihu přeložilo, stojí o zpětnou vazbu a bude na vaše podněty a dotazy reagovat. Můžete se obrátit na následující adresy:

Computer Press
redakce počítačové literatury

Holandská 8
639 00 Brno

nebo

knihy@cpress.cz.

Další informace a případné opravy českého vydání knihy najdete na internetové adrese <http://knihy.cpress.cz/K1696>. Prostřednictvím uvedené adresy můžete též naší redakci zaslat komentář nebo dotaz týkající se knihy. Na vaše reakce se srdečně těšíme.

Poděkování

Tuto knihu bych nikdy nemohl napsat, kdyby mě při jejím psaní nepodporovalo tolik skvělých lidí. Chtěl bych poděkovat své matce, Connie, které tuto knihu také věnuji. Bez její tvrdé práce a sebeobětování bych dnes nebyl tím, čím jsem. Díky za všechno, mami. Jsem ti vděčný za vše, co jsi udělala pro mě a pro mého bratra. Být tvým synem je požehnání.

Mému bratrovi Joeovi: pokaždé, když jsem přijel z Baltimoru domů, abych si odpočinul od psaní, jsi tady byl pro mě, abys mi připomněl, že život není jenom práce, a že až tuto knihu dopíšu, měl bych se opět věnovat důležitějším věcem, které život přináší. Joe, jsi dobrý chlap a moc si tě vážím. Jsem na tebe hrdý, a jsem hrdý na to, že jsi mým bratrem.

Mé úžasné přítelkyni Georgii: bez tvé podpory bych nikdy nemohl napsat více než 600 stran této knihy. Byla jsi se mnou celou dobu, každý den. Víím, že to pro tebe bylo stejně náročné, jako pro mě. Pracoval jsem celé dny a celé noci psal, ale ty jsi byla stále skvělá. Měla jsi pro to pochopení a podporovala mě a za to ti budu navždy vděčný. Děkuji ti. Miluji tě.

Mému budoucímu tchánovi Georgovi a tchýni Kiki: děkuji vám za podporu, kterou jste mi celou dobu poskytovali. Vždy, když jsem vás přijel navštívit, cítil jsem se u vás jako doma a Kiki se vždy postarala o to, abychom se s Georgií dobře najedli. Chtěl bych poděkovat také svým švagrovým Anně a Kathy. Vždy jsem se s vámi dobře bavil a vždy jsme si u vás s Georgií skvěle odpočinuli od psaní a od Baltimoru.

Mému redaktorovi, Jonathanu Gennickovi, bez něj bych tuto knihu nikdy nenapsal. Jonathane, zasloužíš si obrovské uznání. Jako redaktor jsi dělal mnohem víc, než bylo tvou prací, a za to ti velice děkuji. Vymýšlel jsi témata jednotlivých návodů, provedl nespočet korektur, pomáhal mi udržet si dobrou náladu, i když už se blížil termín dokončení knihy. Bez tebe bych to nedokázal. Jsem vděčný za to, že jsi byl mým redaktorem a za příležitost, kterou jsi mi dal. Bylo mi ctí pracovat s tak zkušeným správcem databází a autorem knih, jakým jsi ty. Nevěřím tomu, že existuje o mnoho víc redaktorů, kteří mohou přestat být redaktory a začít pracovat jako správci databází, když se tak rozhodnou. Díky tomu jsi obvykle věděl, co chci říci, i když jsem měl potíže to vyjádřit. Nakladatelství O'Reilly má štěstí, že pro něj pracuješ jako redaktor, a já mám štěstí, že mým redaktorem jsi zrovna ty.

Rád bych poděkoval Alesi Speticovi a Jonathanu Gennickovi za knihu *Transact-SQL Cookbook*. Isaac Newton jednou řekl: „Jestli jsme někdy viděl o kousek dál než ostatní, bylo to proto, že jsem stál na ramenou obrům.“ V poděkování své knihy napsal Ales Spetic něco jako odkaz všem budoucím čtenářům a já si myslím, že by tento odkaz měl být v každé knize o SQL:

Doufám, že tato kniha doplní ostatní díla výborných autorů, jakými jsou Joe Celko, David Rozenshtein, Anatoly Abramovich, Eugene Berger, Iztik Ben-Gan, Richard Snodgrass a další. Strávil jsem mnoho nocí studiem jejich práce, a vše co umím jsem se naučil z jejich knih. Při psaní těchto řádků si uvědomuji, že každá noc, kterou jsem strávil objeováním jejich tajemství, je stála deset nocí, v nichž svoje vědomosti organizovali do konzistentní a čitelné podoby. Je mi ctí, že mám možnost tuto službu SQL komunitě nějak oplatit.

Dále bych rád poděkoval Sanjayi Mishrovi za jeho výbornou knihu *Mastering Oracle SQL* a za to, že mě dal dohromady s Jonathanem Gennickem. Jinak bych se s ním nikdy neseznámil a nikdy bych tuto knihu býval nenapsal. Je zvláštní, jak může obyčejný e-mail změnit váš život. Chci poděkovat Davidovi Rozenshteinovi, hlavně za jeho knihu *Essence of SQL*, díky které jsem se naučil, jak přemýšlet nad problémy s množinami dat a problémy v SQL, a řešit je. Chci také poděkovat Davidovi Rozenshteinovi, Anatolyi Abramovichovi a Eugene Bergerovi za jejich knihu *Optimizing Transact-SQL*, díky které jsem se naučil mnoho pokročilých technik práce s SQL, jež stále využívám.

Rád bych poděkoval celému týmu Wireless Generation. Je to skvělá firma a pracuji v ní skvělí lidé. Velký dík patří všem těmto lidem, kteří si udělali čas k recenzování či kritice knihy a již mi nabídli pomoc při jejím psaní: Jesse Davisovi, Joelovi Pattersonovi, Philipovi Zee, Kevinovi Marshallovi, Dougovi Danielsovi, Otisovi Gospodneticovi, Kenovi Gunnovi, Johnovi Stewartovi, Jimovi Abramsonovi, Adamovi Mayerovi, Susan Lauové, Alexis Le-Quocové a Paulovi Feuerovi. Chci poděkovat Maggie Hoové za její pečlivou recenzi a velmi užitečnou zpětnou vazbu při psaní úvodu do analytických funkcí. Chci poděkovat Chuckovi van Burenovi a Gillian Gutenbergové za jejich skvělou radu ohledně běhání. Ranní cvičení mi pomohlo vyčistit si hlavu a odpočinout si. Myslím, že bez chvíle oddechu bych tuto knihu nedokončil. Chci poděkovat Stevovi Kangovi a Chadovi Levinsonovi za to, že mě vždy poslouchali, když jsem nepřetržitě mluvil o různých technikách programování v SQL, i když jediné, po čem toužili po dlouhém dnu v práci, byl hamburger a pivo na Union Square v Hertland Brewery. Rád bych poděkoval Aaronovi Boydovi za všechnu podporu, jeho milá slova a hlavně za jeho cenné rady. Aaron je čestný, upřímný a velmi pracovitý chlapík; díky lidem jako je Aaron je Wireless Generation lepší firmou. Chci také poděkovat Olivierovi Pomelovi za jeho podporu při psaní této knihy a hlavně za pomoc, kterou mi poskytl při psaní sekce řešení pro DB2 v návodu věnovaném tvorbě dělených výpisů. Olivier navrhl uvedený způsob řešení daného problému, aniž by měl k dispozici systém DB2 a mohl ho otestovat! Pouze jsem mu vysvětlil, jak funguje klauzule WITH a on za několik minut přišel se způsobem řešení, který najdete v této knize.

Užitečnou recenzi manuskriptu této knihy také provedli Jonah Harris a David Rozenshtein. Arun Marathe, Nuno Pinto do Souto a Andrew Odewahn pak provedli zhodnocení návrhu této knihy a návodů, které jsou v ní uvedené, když byla kniha ve fázi přípravy. Všem vám velice děkuji.

Chtěl bych také poděkovat Johnovi Hayduovi a vývojovému týmu klauzule MODEL firmy Oracle Corporation, kteří si našli čas a provedli korekturu sekce věnované klauzuli MODEL, a umožnili mi lépe porozumět tomu, jak tato klauzule přesně funguje. Rád bych poděkoval Tomovi Kyteovi z firmy Oracle Corporation, který mi dovolil adaptovat jeho funkci TO_BASE tak, aby bylo možné ji využít v jakémkoliv systému. Bruno Denuit zodpověděl všechny moje otázky týkající se specifikace analytických funkcí představených v systému SQL Server 2005. Vývojář PostgreSQL Simon Riggs mě průběžně informoval o nových nástrojích PostgreSQL (velký dík mu patří zejména za to, že díky tomu, že jsem věděl, co a kdy bude do systému přidáno, jsem mohl do knihy zařadit informace o nových nástrojích PostgreSQL, jako je například skvělá funkce GENERATE_SERIES, díky které je podle mého názoru možné řešit mnoho problémů elegantnějším způsobem, než s pomocí kontingenčních tabulek).

V neposlední řadě bych chtěl poděkovat Kayovi Youngovi. Když máte talent a milujete svou práci, je skvělé pracovat s lidmi, kteří jsou stejně talentovaní a milují svou práci stejně jako vy. Mnoho z návodů, které v knize najdete, vzešlo ze spolupráce s Kayem a z našeho společného každodenního řešení problémů ve firmě Wireless Generation. Chci ti poděkovat, Kayi, a chci, abys věděl, že si hluboce vážím pomoci, kterou jsi mi při psaní knihy poskytl; počínaje různými radami, přes korektury jazykových problémů až po pomoc při psaní zdrojových kódů, jsi vždy byl nenahraditelnou součástí týmu, jenž se na vzniku této knihy podílel. Je skvělé, že s tebou mohu pracovat a díky Tobě je Wireless Generation lepší firmou.

V září 2005, Anthony Molinaro