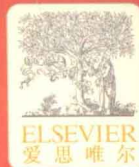


TURING

图灵程序设计丛书



[美] Joe Celko 著
王渊 钟鸣 朱巍 译

SQL 权威指南

(第4版)

Joe Celko's
SQL for
Smarties

Advanced SQL
Programming

Fourth Edition

- 世界级数据库专家Joe Celko经典力作
- 掌握高级技术，精通SQL编程的不二之选
- 揭示SQL标准背后鲜为人知的理论与实践考量



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

[美] Joe Celko 著
王渊 钟鸣 朱巍 译

SQL 权威指南

(第4版)

**Joe Celko's
SQL for
Smarties**

Advanced SQL
Programming

Fourth Edition

人民邮电出版社
北京

图书在版编目 (CIP) 数据

SQL 权威指南: 第 4 版 / (美) 塞科 (Celko, J.) 著;
王渊, 钟鸣, 朱巍译. -- 北京: 人民邮电出版社,
2013.1

(图灵程序设计丛书)

书名原文: Joe Celko's SQL for Smarties :
Advanced SQL Programming, Fourth Edition
ISBN 978-7-115-29663-4

I. ① S… II. ① 塞… ② 王… ③ 钟… ④ 朱… III. ①
关系数据库 - 数据库管理系统 - 指南 IV.
① TP311.138-62

中国版本图书馆 CIP 数据核字 (2012) 第 242360 号

内 容 提 要

本书为 SQL 名著中文版, 兼顾技术与实践, 全面细致介绍高级技术, 致力于打造 SQL 编程专家。本书阐释了数据库设计、优化和操作的各方面内容, 提供了成为 SQL 编程专业人士所需的技术与技巧、针对新旧挑战性难题的优秀解决方案、专业的思考方式 (以保证程序的正确性与高效性), 并涉及了数据库设计与规范化、SQL 数据类型、查询、分组、集合操作、优化等主题。另外, Joe Celko 以通俗易懂的语言叙述了一些关键问题, 比如避免使用过多 NULL 的原因及查询优化方式等。

本书适合中高级 SQL 编程人员学习参考。

图灵程序设计丛书 SQL 权威指南 (第 4 版)

-
- ◆ 著 [美] Joe Celko
译 王 渊 钟 鸣 朱 巍
责任编辑 毛倩倩
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鑫正大印刷有限公司印刷
- ◆ 开本: 800 × 1000 1/16
印张: 41.75
字数: 1039 千字
印数: 1-3 500 册
- 2013 年 1 月第 1 版
2013 年 1 月北京第 1 次印刷
- 著作权合同登记号 图字: 01-2011-1853 号
ISBN 978-7-115-29663-4
-

定价: 99.00 元

读者服务热线: (010)51095186 转 604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Joe Celko's SQL for Smarties: Advanced SQL Programming, Fourth Edition by Joe Celko, ISBN: 978-0-12-382022-8 .

Copyright © 2011 by Elsevier. All rights reserved.

Authorized Simplified Chinese translation edition published by the Proprietor.

Copyright © 2013 by Elsevier (Singapore) Pte Ltd.

All rights reserved.

Printed in China by POSTS & TELECOM PRESS under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in China only, excluding Hong Kong SAR, Macao SAR and Taiwan Province. Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书简体中文版由 Elsevier (Singapore) Pte Ltd. 授权人民邮电出版社在中华人民共和国境内（不包括香港特别行政区、澳门特别行政区和台湾地区）出版与发行。未经许可之出口，视为违反著作权法，将受法律之制裁。

本书贴有 Elsevier 防伪标签，无标签者不得销售。

第 4 版简介

与之前推出的第 1 版、第 2 版以及第 3 版一样，第 4 版也面向想要了解高级编程技巧的 SQL 编程人员。本书读者需要具有一年以上实际 SQL 编程经验。这不是一本入门书，所以我希望亚马逊网站的评论中不会出现前几版上市时的抱怨。

本书第 1 版出版于十年前，并被部分 SQL 程序员奉为经典。在我访问过的几乎所有软件公司中都能看到程序员的桌子上放着这本书。最棒的是，我发现有即时贴伸出书本：太棒了，看来他们常常要用到这本书，所以才会用即时贴当书签。

十年间的变化

层次数据库及网络数据库仍然在大公司的遗留系统中运行。财富 500 强公司中仍存在 IMS (Information Management System, 信息管理系统) 和传统文件，尽管 SQL 工作者并不愿意承认这点。不过 SQL 工作者依然可以感到自豪，这十年来，基于 SQL 的系统取得了很大进展。现在，我们拥有了几乎无所不包的应用程序，以及重要的、更小的数据库。

尽管 OO 编程依然牢牢占据主导地位，但是在下一个十年里，函数式编程或许能够抢占 OO 编程的一些空间。尽管对象以及对象关系数据库找到了适合的市场，但仍没有机会占据主流地位。

2010 年后，XML 不再流行。从技术层面讲，XML 是一种用于描述数据以及将数据从某一平台移植到另一平台的语法，而它的支持工具也提供了搜索以及重新格式化的功能。INCITS H2 (前身为 ANSI X3H2 数据库标准委员会) 下设了一个 SQL/XML 委员会，以确保 XML 能与其支持工具一同工作。

数据仓库不再是只有大型企业才能使用的奢侈品。由于硬件和软件价格的下降，中型企业现在也能使用数据仓库了。编写 OLAP 查询与编写 OLTP 查询不同，也许需要专门写一本讲 OLAP 查询的“Smarties”书。

开源数据库做得很棒，也越来越符合标准。LAMP (Linux、Apache、MySQL 以及 Python/PHP) 平台占领了绝大多数网站。Ingres、Postgres、Firebird 以及其他数据库实现了 ANSI SQL-92 标准特性、大多数的 SQL-99 特性以及一些 SQL:2003 特性。

纵列数据库 (columnar database)、并行以及开放式并发机制开始出现在商业产品中，而不再局限于实验室内。SQL 标准总是在不断改变，但并不是每一次都变得更好。标准的某些部分

变得更趋于关系及集合，而其他部分则很明显地往使用过程式思想、处理非关系型数据的方向发展，这类标准建立在文件系统模型之上。引用 David McGoveran 的一句话：“委员会从未见过一个不喜欢的特性。”这句话，看来说得没错。

ANSI/ISO SQL-92 标准是一组公有的子集，通用于各类 SQL 产品，使这些产品能够为人所用。事实上，几年前美国政府将 SQL-99 标准描述为“仍在开发的标准”，要求联邦政府的合同必须与 SQL-92 标准兼容。

在开发兼容 SQL-92 标准的产品时，我们可以使用 FIPS-127 一致性测试套件对产品进行测试，这样所有供应商均可朝着同一个方向迈进。不过遗憾的是，克林顿政府中止了这个规定，而一致性问题又开始浮现。Whitemarsh 信息系统公司董事长 Michael M. Gorman 曾担任数据库标准委员会 INCITS H2 秘书长二十余年，他有一篇关于一致性问题的重要论文发表在 Wiscorp.com，该网站中其他关于 SQL 历史中政治方面的文章也值得一读。

今天，SQL-99 是编写在绝大多数平台上的可移植代码的标准。不过由于厂商支持 SQL:2003 特性的速度很快，我并不认为要局限于各个平台的最小交集。

第4版的新内容

在第2版中，我曾删除了书中的一些理论，并将这部分理论移到了 *Joe Celko's Data and Databases: Concepts in Practice* 一书中。我找不到任何理由将这部分理论移回到第4版。

由于树及层次技术相关内容足以编成一本书，因此我将这部分知识扩展并移到 *Joe Celko's Trees and Hierarchies in SQL for Smarties* 中。不过本书也简单提及了树及层次技术。

由于本书面向高级程序员，因此我将适合新手的编程技巧移到了 *Joe Celko's SQL Programming Style* 一书中讲述。本书适合那些编写真正 SQL 语句的读者，而非使用某些 SQL “方言”或伪装成 SQL 的原生语言的读者。实际上，将标准 SQL 语言翻译成他们所使用的 SQL “方言”不会太麻烦。

我尝试在方案中嵌入注解，以说明为什么该方案能够生效。我希望这种方式能帮助读者了解底层原理，以便将此原理应用到其他情形。

许多人为本书提供了素材，他们有些直接告诉我这些素材，有些则是通过新闻组提供了素材，而我无法对这些提供素材的朋友一一表示感谢，但我会尽量在他们提供的代码后面贴上他们的名字。为了避免遗漏某些朋友，我在此列出向我提供了素材或思路的朋友名单：Aaron Bertrand、Alejandro Mesa、Anith Sen、Craig Mullins（为本书的多个版本进行了校对）、Daniel A. Morgan、David Portas、David Cressey、Dawn M. Wolthuis、Don Burleson、Erland Sommarskog、Itzak Ben-Gan、John Gilson、Knut Stolze、Ken Henderson、Louis Davidson、Dan Guzman、Hugo Kornelis、Richard Romley、Serge Rielau、Steve Kass、Tom Moreau、Troels Arvin、Vadim Tropashko、Plamen Ratchev、Gert-Jan Strik。另外，还要感谢其他为我提供了素材，而我却忘记了名字的朋友们。

更正和补充^①

请将所有更正、补充、建议、改进以及替代方案发给出版商或我本人，尤其是当你发现了某些更好的方法时，请一定告知。

出版商网站：www.mkp.com。

^① 读者也可免费注册图灵社区 (ituring.com.cn) 并为本书提交勘误。——编者注

目 录

第 1 章 数据库与文件系统	1	3.5.2 为约束使用主键和断言	23
1.1 实体表	3	3.6 字符集相关结构	25
1.2 关系表	3	3.6.1 创建字符集	25
1.3 行与记录	3	3.6.2 创建排序规则	26
1.4 列与字段	4	3.6.3 创建翻译	26
1.5 模式对象	5	第 4 章 定位数据和特殊数值	27
1.6 CREATE SCHEMA 语句	6	4.1 显式的物理定位器	27
第 2 章 事务与并发控制	8	4.1.1 ROWID 和物理磁盘地址	27
2.1 会话	8	4.1.2 标识列	27
2.2 事务与 ACID	9	4.2 生成的标识符	30
2.2.1 原子性	9	4.2.1 GUID	30
2.2.2 一致性	10	4.2.2 UUID	31
2.2.3 隔离性	10	4.3 序列生成函数	32
2.2.4 持久性	10	4.4 预分配值	33
2.3 并发控制	11	4.5 特殊序列	34
2.3.1 三种现象	11	4.5.1 Series 表	34
2.3.2 隔离级别	12	4.5.2 素数	35
2.4 保守式并发控制	13	4.5.3 随机顺序值	37
2.5 快照隔离与乐观式并发	14	4.5.4 其他序列	39
2.6 逻辑并发控制	16	第 5 章 基础表和相关元素	40
2.7 死锁与活锁	16	5.1 CREATE TABLE 语句	41
第 3 章 数据库模式对象	17	5.1.1 列约束	41
3.1 CREATE SCHEMA 语句	17	5.1.2 DEFAULT 子句	43
3.2 CREATE PROCEDURE、CREATE FUNCTION 以及 CREATE TRIGGER 语句	18	5.1.3 NOT NULL 约束	43
3.3 CREATE DOMAIN 语句	18	5.1.4 CHECK() 约束	44
3.4 创建序列	19	5.1.5 UNIQUE 以及 PRIMARY KEY 约束	46
3.5 创建断言	19	5.1.6 REFERENCES 子句	47
3.5.1 为模式级约束使用视图	20	5.2 嵌套 UNIQUE 约束	49
		5.2.1 重叠键	52

5.2.2 单列唯一性与多列唯一性	54	6.4.1 一个常见的错误	88
5.3 CREATE ASSERTION 约束	62	6.4.2 一处改进	89
5.4 临时表	62	6.5 重写技巧	94
5.5 表操作	63	6.5.1 数据表和生成器代码	95
5.5.1 DROP TABLE <表名>	64	6.5.2 用计算替代查找	96
5.5.2 ALTER TABLE	64	6.5.3 斐波那契数列	96
5.6 避免属性分割	65	6.6 谓词函数	97
5.6.1 表级属性分割	66	6.7 过程化分解和逻辑分解	98
5.6.2 行级属性分割	67	6.7.1 过程式分解方案	99
5.7 在 DDL 中表现类层次关系	68	6.7.2 逻辑分解方案	100
5.8 显式物理定位器	70	第 7 章 过程式结构	102
5.9 自增列	70	7.1 创建过程	102
5.9.1 ROWID 与物理磁盘地址	72	7.2 创建触发器	103
5.9.2 标识列	72	7.3 游标	106
5.9.3 对比标识列和序列	73	7.3.1 DECLARE CURSOR 语句	106
5.10 生成标识符	73	7.3.2 ORDER BY 子句	107
5.10.1 行业标准的唯一标识符	73	7.3.3 OPEN 语句	113
5.10.2 国防部的唯一标识符	74	7.3.4 FETCH 语句	113
5.10.3 序列生成函数	75	7.3.5 CLOSE 语句	114
5.10.4 唯一值生成器	75	7.3.6 DEALLOCATE 语句	114
5.10.5 验证源	76	7.3.7 如何使用游标	114
5.11 关于重复行	77	7.3.8 位置更新及删除语句	117
5.12 其他模式对象	78	7.4 序列	117
5.13 临时表	79	7.5 生成列	118
5.14 CREATE DOMAIN 语句	79	7.6 表函数	119
5.15 CREATE TRIGGER 语句	80	第 8 章 辅助表	121
5.16 CREATE PROCEDURE 语句	80	8.1 序列表	121
5.17 DECLARE CURSOR 语句	81	8.1.1 对列表进行枚举	122
5.17.1 如何使用游标	83	8.1.2 将序列映射为循环	124
5.17.2 位置更新及删除语句	84	8.1.3 取代迭代循环	125
第 6 章 过程式、半过程式以及 声明式编程	86	8.2 查找辅助表	127
6.1 软件工程基本原理	86	8.2.1 简单转换辅助表	128
6.2 内聚性	86	8.2.2 多转换值辅助表	128
6.3 耦合度	87	8.2.3 多参数辅助表	129
6.4 大跨越	88	8.2.4 范围辅助表	129

8.2.5 层次结构辅助表	130	10.5.1 NULLIF() 函数	187
8.2.6 “一个真正的查找表”	131	10.5.2 COALESCE() 函数	187
8.3 辅助函数表	133	10.6 数学函数	189
8.3.1 用辅助表求反函数	134	10.6.1 数学运算符	189
8.3.2 用辅助函数表进行插值	141	10.6.2 指数函数	191
8.4 全局常量表	143	10.6.3 标量函数	192
8.4.1 预分配值	143	10.6.4 将数值转换为文字	192
8.4.2 素数	144	10.7 唯一值生成器	193
8.4.3 斐波那契数列	144	10.7.1 存有间隙的序列	194
8.4.4 随机顺序值	145	10.7.2 预分配数值	194
8.5 把过程代码转换成表时的注意事项	147	10.8 IP 地址	195
第 9 章 规范化	152	10.8.1 CHAR(39) 存储	195
9.1 函数依赖和多值依赖	154	10.8.2 二进制存储	196
9.2 第一范式 (1NF)	154	10.8.3 使用多个单独的 SMALLINT	196
9.3 第二范式 (2NF)	158	第 11 章 SQL 中的时间数据类型	197
9.4 第三范式 (3NF)	159	11.1 关于日历标准的说明	197
9.5 基本关键字范式 (EKNF)	160	11.2 SQL 时间数据类型	199
9.6 Boyce-Codd 范式 (BCNF)	161	11.2.1 时间的内部表示	200
9.7 第四范式 (4NF)	162	11.2.2 日期格式标准	200
9.8 第五范式 (5NF)	163	11.2.3 处理时间戳	201
9.9 域-键范式 (DKNF)	164	11.2.4 处理时间	202
9.10 规范化的实用技巧	171	11.2.5 时区和夏令时	203
9.11 键类型	172	11.3 INTERVAL 数据类型	204
9.11.1 自然键	172	11.4 时间算术	206
9.11.2 人工键	172	11.5 时间数据模型的特性	207
9.11.3 对外暴露的物理定位器	173	11.5.1 为持续时间建模	207
9.12 非规范化的实用技巧	174	11.5.2 持续时间之间的关系	209
第 10 章 SQL 的数值数据	180	第 12 章 字符数据类型	211
10.1 数值类型	180	12.1 SQL 字符串问题	211
10.2 数值类型的转换	183	12.1.1 字符串相等问题	212
10.2.1 数值的舍入和截断	183	12.1.2 字符串排序问题	212
10.2.2 CAST() 函数	185	12.1.3 字符串分组问题	213
10.3 四则运算函数	185	12.2 标准字符串函数	213
10.4 算术运算和 NULL	186	12.3 常见的厂商扩展	214
10.5 值与 NULL 的相互转换	187		

12.4 Cutter 表	222	执行删除	249
12.5 嵌套替换	223	15.1.4 在相同表内进行删除	250
第 13 章 NULL : SQL 中的缺失数据	224	15.1.5 不用声明引用完整性 在多个表中进行删除	252
13.1 空表和缺失表	225	15.2 INSERT INTO 语句	253
13.2 列中的缺失值	225	15.2.1 INSERT INTO 子句	253
13.3 上下文和缺失值	226	15.2.2 插入的性质	254
13.4 比较 NULL	227	15.2.3 批量装载和卸载实用程序	255
13.5 NULL 和逻辑	228	15.3 UPDATE 语句	255
13.5.1 子查询谓词中的 NULL	229	15.3.1 UPDATE 子句	255
13.5.2 逻辑值谓词	231	15.3.2 WHERE 子句	256
13.6 算术中的 NULL 值	231	15.3.3 SET 子句	256
13.7 函数中的 NULL 值	231	15.3.4 利用第二张表进行更新	257
13.8 NULL 和宿主语言	231	15.3.5 在 UPDATE 中使用 CASE 表达式	259
13.9 NULL 的设计忠告	232	15.4 常见厂商扩展的缺陷说明	261
13.10 关于多 NULL 值的说明	234	15.5 MERGE 语句	263
第 14 章 多列数据元素	237	第 16 章 比较或 theta 操作	266
14.1 距离函数	237	16.1 数据类型转换	266
14.2 在 SQL 中存储 IPv4 地址	239	16.1.1 日期显示格式	267
14.2.1 使用单个 VARCHAR(15) 列 表示 IPv4 地址	239	16.1.2 其他显示格式	268
14.2.2 使用一个 INTEGER 列表示 IPv4 地址	239	16.2 SQL 中的行比较	268
14.2.3 使用四个 SMALLINT 列表 示 IPv4 地址	240	16.3 IS [NOT] DISTINCT FROM 操作符	270
14.3 在 SQL 中存储 IPv6 地址	241	第 17 章 值化谓词	271
14.4 货币与其他单位的转换	242	17.1 IS NULL 谓词	271
14.5 社会安全号	242	17.2 IS [NOT] {TRUE FALSE UNKNOWN} 谓词	272
14.6 有理数	245	17.3 IS [NOT] NORMALIZED 谓词	273
第 15 章 表操作	246	第 18 章 CASE 表达式	275
15.1 DELETE FROM 语句	246	18.1 CASE 表达式	275
15.1.1 DELETE FROM 子句	246	18.1.1 COALESCE() 和 NULLIF() 函数	278
15.1.2 WHERE 子句	247	18.1.2 带 GROUP BY 的 CASE 表达式	278
15.1.3 根据辅助表中的数据			

18.1.3 CASE、CHECK() 子句和逻辑蕴涵	280	22.5 EXISTS() 和引用约束	320
18.2 子查询表达式和常量	283	22.6 EXISTS 和三值逻辑	320
18.3 Rozenshtein 特征函数	283	第 23 章 量子查询谓词	323
第 19 章 LIKE 与 SIMILAR TO 谓词	285	23.1 标量子查询比较	323
19.1 使用模式的技巧	285	23.2 量词和缺失数据	324
19.2 NULL 值和空字符串的谓词结果	287	23.3 ALL 谓词和极值函数	326
19.3 LIKE 并不是相等	287	23.4 UNIQUE 谓词	327
19.4 用联结消除 LIKE 谓词	287	23.5 DISTINCT 谓词	328
19.5 CASE 表达式和 LIKE 搜索条件	288	第 24 章 简单 SELECT 语句	329
19.6 SIMILAR TO 谓词	289	24.1 SELECT 语句执行顺序	329
19.7 字符串的有关技巧	291	24.2 单级 SELECT 语句	329
19.7.1 字符串的字符内容	291	第 25 章 高级 SELECT 语句	336
19.7.2 搜索与声明一个串	291	25.1 关联子查询	336
19.7.3 创建字符串中的索引	292	25.2 嵌入的 INNER JOIN	340
第 20 章 BETWEEN 和 OVERLAPS 谓词	293	25.3 OUTER JOIN	341
20.1 BETWEEN 谓词	293	25.3.1 OUTER JOIN 的一些历史	342
20.1.1 NULL 值的结果	294	25.3.2 NULL 和 OUTER JOIN	346
20.1.2 空集的结果	294	25.3.3 NATURAL JOIN 与搜索式	
20.1.3 程序设计技巧	295	OUTER JOIN	347
20.2 OVERLAPS 谓词	296	25.3.4 OUTER JOIN 自联结	348
第 21 章 [NOT] IN() 谓词	305	25.3.5 两次或多次 OUTER JOIN	349
21.1 优化 IN() 谓词	306	25.3.6 OUTER JOIN 和聚合函数	351
21.2 用 IN() 谓词替换 OR	309	25.3.7 FULL OUTER JOIN	351
21.3 NULL 和 IN() 谓词	309	25.4 UNION JOIN 操作符	352
21.4 IN() 谓词和引用约束	312	25.5 标量 SELECT 表达式	353
21.5 IN() 谓词和标量查询	313	25.6 旧 JOIN 语法与新 JOIN 语法	354
第 22 章 EXISTS() 谓词	315	25.7 受约束的 JOIN	355
22.1 EXISTS 和 NULL	316	25.7.1 库存和订单	355
22.2 EXISTS 和 INNER JOIN	318	25.7.2 稳定的婚姻	356
22.3 NOT EXISTS 和 OUTER JOIN	318	25.7.3 将球装入盒中	360
22.4 EXISTS() 和量词	319	25.8 Codd 博士的 T 联结	363
		25.8.1 Stobbs 方案	366
		25.8.2 Picere 方案	367

25.8.3 参考文献	368	27.1.1 按范围分区	393
第 26 章 虚拟表：视图、派生表、CTE 及 MQT	369	27.1.2 单列范围表	394
26.1 查询中的视图	369	27.1.3 用函数进行分区	394
26.2 可更新视图和只读视图	370	27.1.4 按顺序分区	395
26.3 视图的类型	371	27.1.5 使用窗口函数进行分区	397
26.3.1 单表投影和限制	371	27.2 关系除法	398
26.3.2 计算列	371	27.2.1 带余除法	399
26.3.3 转换列	372	27.2.2 精确除法	400
26.3.4 分组视图	372	27.2.3 性能说明	400
26.3.5 联结视图	373	27.2.4 Todd 的除法	401
26.3.6 视图的联结	374	27.2.5 带 JOIN 的除法	403
26.3.7 嵌套视图	375	27.2.6 用集合操作符进行除法	403
26.4 数据库引擎如何处理视图	376	27.3 Romley 除法	404
26.4.1 视图列列表	376	27.4 RDBMS 中的布尔表达式	407
26.4.2 视图物化	376	27.5 FIFO 和 LIFO 子集	408
26.4.3 内嵌文本扩展	377	第 28 章 分组操作	411
26.4.4 指针结构	378	28.1 GROUP BY 子句	411
26.4.5 索引和视图	379	28.2 GROUP BY 和 HAVING	412
26.5 WITH CHECK OPTION 子句	379	28.3 多层次聚合	415
26.6 删除视图	383	28.3.1 多级聚合的分组视图	415
26.7 视图与临时表的使用提示	384	28.3.2 多层次聚合的子查询表达式	416
26.7.1 使用视图	384	28.3.3 多层聚合的 CASE 表达式	417
26.7.2 使用临时表	385	28.4 在计算列上分组	418
26.7.3 用视图扁平化表	385	28.5 成对分组	418
26.8 使用派生表	387	28.6 排序和 GROUP BY	420
26.8.1 FROM 子句中的派生表	387	第 29 章 简单聚合函数	422
26.8.2 包含 VALUES 构造器的派生表	388	29.1 COUNT() 函数	422
26.9 公用表表达式	389	29.2 SUM() 函数	426
26.10 递归公用表表达式	390	29.3 AVG() 函数	427
26.10.1 简单增量	391	29.3.1 空组的平均数	428
26.10.2 简单树遍历	391	29.3.2 多个列上的平均值	429
26.11 物化查询表	392	29.4 极值函数	430
第 27 章 在查询中分区数据	393	29.4.1 简单的极值函数	430
27.1 覆盖和分区	393	29.4.2 广义极值函数	432
		29.4.3 多条件极值函数	438

29.4.4	GREATEST() 和 LEAST() 函数	439
29.5	LIST() 聚合函数	442
29.5.1	使用递归 CTE 的 LIST 聚合函数	442
29.5.2	交叉表的 LIST() 函数	443
29.6	PRD() 聚合函数	443
29.6.1	通过表达式实现 PRD() 函数	444
29.6.2	通过对数实现 PRD() 聚合函数	445
29.7	位运算符聚合函数	447
29.7.1	OR 位运算符聚合函数	448
29.7.2	AND 位运算符聚合函数	449

第 30 章 高级分组、窗口聚合 以及 SQL 中的 OLAP 450

30.1	星模式	450
30.2	GROUPING 操作符	451
30.2.1	GROUP BY GROUPING SET	451
30.2.2	ROLLUP	452
30.2.3	CUBE	452
30.2.4	SQL 的 OLAP 示例	453
30.3	窗口子句	454
30.3.1	PARTITION BY 子句	454
30.3.2	ORDER BY 子句	454
30.3.3	窗口帧子句	455
30.4	窗口化聚合函数	456
30.5	序号函数	457
30.5.1	行号	457
30.5.2	RANK() 和 DENSE_RANK()	457
30.5.3	PERCENT_RANK() 和 CUME_DIST()	457
30.5.4	一些示例	458
30.6	厂商扩展	460
30.6.1	LEAD 和 LAG 函数	460
30.6.2	FIRST 和 LAST 函数	461
30.7	一点历史知识	462

第 31 章 SQL 中的描述性统计 463

31.1	众数	463
31.2	AVG() 函数	464
31.3	中值	464
31.3.1	中值编程问题	465
31.3.2	Celko 第一中值	466
31.3.3	Date 第二中值	467
31.3.4	Murchison 中值	468
31.3.5	Celko 第二中值	468
31.3.6	Vaughan 提出的应用视图 的中值	470
31.3.7	使用特征函数的中值	470
31.3.8	Celko 第三中值	473
31.3.9	Ken Henderson 的中值	475
31.3.10	OLAP 中值	476
31.4	方差和标准偏差	478
31.5	平均偏差	479
31.6	累积统计	479
31.6.1	运行差分	479
31.6.2	累积百分比	481
31.6.3	序号函数	483
31.6.4	五分位数和相关统计	486
31.7	交叉表	486
31.7.1	通过交叉联结建立交叉表	489
31.7.2	通过外联结建立交叉表	490
31.7.3	通过子查询建立交叉表	490
31.7.4	使用 CASE 表达式建立 交叉表	491
31.8	调和平均数和几何平均数	491
31.9	SQL 中的多变量描述统计数据	492
31.9.1	协方差	492
31.9.2	皮尔森相关系数 r	493
31.9.3	多变量描述统计中的 NULL 值	493
31.10	SQL:2006 中的统计函数	494
31.10.1	方差、标准偏差以及 描述统计	494
31.10.2	相关性	494

31.10.3 分布函数.....	495	34.2.1 没有 NULL 值和重复行时的 INTERSECT 和 EXCEPT 操作....	534
第 32 章 子序列、区域、 顺串、间隙及岛屿.....	496	34.2.2 存在 NULL 值和重复行时的 INTERSECT 和 EXCEPT 操作....	535
32.1 查找尺寸为 n 的子区域	496	34.3 关于 ALL 和 SELECT DISTINCT 的一个说明.....	536
32.2 为区域编号.....	497	34.4 相等子集和真子集.....	536
32.3 查找最大尺寸的区域.....	499	第 35 章 子集.....	538
32.4 界限查询.....	502	35.1 表中的每个第 n 项	538
32.5 顺串和序列查询.....	503	35.2 从表中选取随机行.....	539
32.6 数列的求和.....	506	35.3 CONTAINS 操作符.....	543
32.7 交换和平移列表值.....	509	35.3.1 真子集操作符.....	543
32.8 压缩一列数值.....	510	35.3.2 表的相等操作.....	544
32.9 折叠一列数值.....	510	35.4 序列间隙.....	547
32.10 覆盖.....	511	35.5 重叠区间的覆盖问题.....	549
第 33 章 SQL 中的矩阵.....	516	35.6 选取有代表性的子集.....	552
33.1 通过命名列进行访问的数组.....	516	第 36 章 SQL 中的树和层次结构.....	556
33.2 通过下标列进行访问的数组.....	519	36.1 邻接列表模型.....	557
33.3 SQL 的矩阵操作.....	520	36.1.1 复杂约束.....	557
33.3.1 矩阵等式.....	521	36.1.2 查询的过程遍历.....	559
33.3.2 矩阵加法.....	521	36.1.3 更改表.....	560
33.3.3 矩阵乘法.....	522	36.2 路径枚举模型.....	560
33.3.4 矩阵转置.....	523	36.2.1 查找子树和节点.....	561
33.3.5 行排序及列排序.....	524	36.2.2 找出层次和后代.....	561
33.3.6 其他矩阵操作.....	524	36.2.3 删除节点和子树.....	562
33.4 将表扁平化为数组.....	524	36.2.4 完整性约束.....	562
33.5 比较表格式中的数组.....	526	36.3 层次结构的嵌套集合模型.....	563
第 34 章 集合操作.....	528	36.3.1 计数特性.....	564
34.1 UNION 和 UNION ALL.....	528	36.3.2 包含特性.....	564
34.1.1 执行顺序.....	530	36.3.3 下级节点.....	565
34.1.2 混合使用 UNION 和 UNION ALL 操作符.....	531	36.3.4 层次聚合.....	566
34.1.3 对同一表中的列执行 UNION 操作.....	531	36.3.5 删除节点和子树.....	566
34.2 INTERSECT 和 EXCEPT.....	531	36.3.6 将邻接列表转换为嵌套集合 模型.....	567
		36.4 其他表现树和层次结构的模型.....	569

第 37 章 SQL 中的图	570	38.9 2000 年问题	616
37.1 邻接列表模型图	570	38.9.1 零	616
37.1.1 SQL 和邻接列表模型	571	38.9.2 闰年	617
37.1.2 路径与 CTE	572	38.9.3 千年问题	618
37.1.3 环状图	577	38.9.4 旧数据中的怪异日期	619
37.1.4 邻接矩阵模型	579	38.9.5 后果	619
37.2 分割嵌套集合模型表示的图节点	580	第 39 章 优化 SQL	620
37.2.1 图中的所有节点	581	39.1 访问方法	621
37.2.2 路径端点	581	39.1.1 顺序访问	621
37.2.3 可达节点	582	39.1.2 索引访问	621
37.2.4 边	582	39.1.3 散列索引	622
37.2.5 入度和出度	582	39.1.4 位向量索引	622
37.2.6 源节点、汇聚节点、孤立节点 和内部节点	583	39.2 如何建立索引	622
37.2.7 将无环图转化为嵌套集合	584	39.2.1 使用简单查询条件	623
37.3 多边形中的点	586	39.2.2 简单字符串表达式	624
37.4 图论参考书目	588	39.2.3 简单时间表达式	625
第 38 章 时间查询	589	39.3 提供额外信息	626
38.1 时间数学	589	39.4 谨慎建立多列索引	627
38.2 个性化日历	591	39.5 考察 IN 谓词	627
38.3 时间序列	592	39.6 避免 UNION	629
38.3.1 时间序列中的间隙	593	39.7 联结胜于嵌套查询	629
38.3.2 连续时间段	595	39.8 使用更少的语句	630
38.3.3 相邻事件中缺失的时间	600	39.9 避免排序	631
38.3.4 查找日期	603	39.10 避免交叉联结	634
38.3.5 时间的起始点和结束点	604	39.11 了解优化器	635
38.3.6 开始时间和结束时间	605	39.12 在模式更改后重编译静态 SQL	636
38.4 儒略日	606	39.13 临时表有时能带来方便	637
38.5 其他时间函数	609	39.14 更新统计数据	639
38.6 星期	610	39.15 不要迷信较新的特性	639
38.7 在表中对时间建模	612	参考文献	642
38.8 日历辅助表	614		

第1章

数据库与文件系统



给我们带来麻烦的往往不是我们完全不知道的事情，而是那些我们自以为了解了，但其实不然的事情。

——阿蒂默斯·沃德（1834—1867），美国作家、幽默作家

数据库和 RDBMS 与那些伴随着 COBOL、FORTRAN、C、BASIC、PL/I、Java 等过程语言或面向对象语言一起出现的文件系统截然不同，SQL 语言本身没有 I/O 系统，因此我们通常把 SQL 解读为“几乎不能称为语言”（Scarcely Qualifies as a Language）。SQL 依赖于宿主语言，以便从终端用户获取数据或为其返回数据。

编程语言通常都以某一底层模型为基础。如果了解这个模型，就能更好地理解这门语言。例如，FORTRAN 语言建立在代数学理论之上，虽然这并不表明 FORTRAN 语言酷似代数学，但如果你了解代数学，那么对 FORTRAN 语言就不会完全陌生，你可以在赋值语言中写出表达式，也可以猜出那些没有见过的库函数所代表的意思。

在大多数其他编程语言中，程序员习惯于对文件进行处理。文件的设计源于纸质表格，这些文件都存储在计算机里并且非常依赖宿主机编程语言。FORTRAN 语言很难读取 COBOL 文件内容，反之亦然。事实上，在相同语言编写的程序之间共享文件也是非常困难的事情。

文件的最基本形式是一组记录，这些记录按照一定的次序记录在文件里，而我们能够通过物理位置引用记录。我们可以打开一个文件，读取第一条记录，之后依次读取后续的记录，直到读完最后一个记录。当读完最后一个记录后再次读取将会返回 EOF（End Of File）值。我们可以在这些记录中来回切换，并且一次操作一条记录，这些操作对于其他程序打开的其他文件没有任何影响，只有程序才能修改文件。

SQL 语言的模型是集合数据，而不是物理文件。SQL 语言的“工作单位”是整个模式（schema），而不是单个表格。

我们在学校里学过数学中集合的概念。集合是无序的，集合中的成员都属于同一类型。当我们对集合进行操作时，这些操作一次作用于全部成员。这就意味着，当我要从正整数集合中选择奇数子集时，返回的是所有奇数值组成的一个集合。我并没有一个个地对整数进行扫描，然后将满足条件的奇数组成集合，仅仅是定义了奇数的规则“如果将整数除以 2 时余数为 1，那么该数为奇数”，这个规则可用于对任何整数进行奇数判定。集合模型具有很多优点，并行处