<<SQL沉思录>>

图书基本信息

书名：<<SQL沉思录>>

13位ISBN编号：9787115213952

10位ISBN编号：711521395X

出版时间：2009-11

出版时间：人民邮电出版社

作者：Joe Celko

页数：270

译者：马树奇

前言

本书讨论的是使用表而不是过程式代码的各种SQL编程技术。
我一直跟大家说，学习SQL编程最重要的就是要摒弃原有的过程式编程，但对于那些一直都在以文件和过程代码的方式来思考的人们而言，仅仅这样说不行，还得具体说明如何采用声明式关系语言来完成相关工作。
因此我编写了本书，展示实际技术，阐述思想方式。

与我的其他作品相同，本书的读者对象是那些希望掌握良好SQL编程技术的专业SQL程序员。
本书假定读者对SQL语言足够了解，已经能够编写可以运行的代码，具有一年的SQL使用经验。

为什么需要具有一年的经验呢？
我通过几十年讲授SQL课程发现，大多数人在由过程式编程语言（如FORTRAN、Cobol、Pascal、C系列语言）、面向对象编程语言等转向SQL语言时，都要经过多个阶段，而这段时间基本为一年。
声明式语言与他们此前使用的所有编程语言都不相同。

学习一门新的编程语言就如同学习一门外语。
一开始，人们会念错单词，并且总想使用自己母语的词序和语法。
接下来，人们会通过努力，根据一种固定模式来正确地造句。
最后，人们才能轻松地以该语言来思考和表述，并且不需要再刻意关注语言本身。

SQL编程的初级阶段只不过是闭着眼睛从别人的程序中把现有的代码复制过来，这并不是真正的编程。
人们可能还会使用一种具有图形用户界面的工具来从文本文件中把SQL语句组合起来，甚至根本不需要显示自己的实际代码。

<<SQL沉思录>>

内容概要

《SQL沉思录》通过大量的实例，详细说明了为提高SQL编程技术而必须面对的思想方法上的根本转变——由以过程式编程方式思考转变为以数据集的方式来思考。
此外，《SQL沉思录》还讨论了关于SQL编程中查找表、视图、辅助表、虚拟表的应用，并独到地阐明了如何在SQL系统中正确地处理时间值以及SQL编程中的其他技术难点。
《SQL沉思录》适合广大数据库编程人员和SQL程序员学习参考。

作者简介

Joe Celko，世界著名的数据库专家，曾担任ANSI SQL标准委员会成员达10年之久，他也是世界上读者数量最多的SQL图书作者之一。
他曾撰写过一系列专栏，并通过他的新闻组支持和推动了数据库编程技术以及ANSI/ISO标准的发展。除本书外，他还撰写了多部SQL经典著作，包括《SQL编程风格》、《SQL解惑》和《SQL权威指南》，上述作品的中文版均已经或即将由人民邮电出版社出版。

<<SQL沉思录>>

书籍目录

章节摘录

第1章　SQL是声明式语言，不是过程式语言　　前言里，我谈到了一些FORTRAN程序员和一名LISP程序员的事，前者只会使用循环　　来解决问题，后者只会使用递归方式解决问题。
这种情况并不少见，因为人们都喜欢使用自己了解的工具。
下面讲一个笑话，不是真事：有人给一个数学家、一个物理学家和一个数据库程序员各发了一个橡皮球，并且让他们确定球的体积。
数学家认真地测量了直径，然后用球体积公式计算出了球的体积，或者认为这个球不很圆，就用三重积分计算了球的体积。
物理学家则在一个大烧杯中接满了水，把球放入水中，测量出排水量。
他并不关心这个球是什么形状。
数据库程序员呢，他到橡皮球生产商的在线数据库里查了这个球的型号和产品序列号，根本不关心这是不是球。
他获得了这个球的制造公差、设计形状和尺寸以及其他许多与整个橡皮球生产过程有关的参数。
这个故事说明：数学家知道如何计算，物理学家知道如何测量，而数据库技术人员知道如何查找数据。
每个人都采用自己的工具来解决问题。
现在我们把问题扩展到仓库中成千上万个橡皮球。
数学家和物理学家因此会花费大量的手工劳动完成任务，而数据库技术员只要下载一些信息，就能够得出橡皮球的工业标准（假设有这种标准）以及详尽得可以用于法庭辩论的文档。
1.1　不同的编程模型　　自我完善的过程就是在学习新知识的同时，忘记老的习惯。
——Edsgar Dijkstra①　　编程模型有多种。
过程式编程语言使用的是由流控制语句（WHILE—D0、IF—THEN—ELSESHBEGIN—END）控制的一系列过程步骤，借此把输入数据转换成输出数据。
这是对编程的一种传统认识，因为这是著名的数学家约翰·冯·诺伊曼归纳出来的，后来也常被称为冯·诺伊曼模型。
同样的源代码经相同的编译器编译之后，每次都生成相同的可执行模块。
该程序在每次调用时都以完全相同的方式工作。
这种模型中的关键字是可以预测和确定的。
由于这种模型具有可确定性，所以主要用于一些数学分析。
另外，还有一些变化。
一些语言使用了不同的流控制语句。
FORTRAN和HCOBOL会在程序一开始就为数据分配全部存储区。
后来的Algol系列编程语言会根据数据在程序块结构中的作用域动态地分配存储区。
EdsgarDijkstra（参见文献www.CS.utexas.edu／users／EWD／）发明了一种非确定性语言。
语句，又称为保护命令，既可以阻止语句的执行，也可以允许该语句的执行，而且在打开的语句之间没有确定的执行顺序。
这种模型没有在商业化产品中实现，但它表明人们原来在编程中认为必备的因素（确定性）可以被丢弃。
函数式编程语言的基础是用一系列嵌套的函数调用来解决问题。
在这些语言中，高阶函数可以转换自身的功能，这个概念非常重要。
导数变换和积分变换就是这种高阶函数在数学上应用的实例。
这种语言的目标之一是避免在程序中出现副作用，保证它们能够以代数的方式进行优化。
特别是，一旦某个表达式与另一个表达式相等（某种意义上的相等），它们就可以替换，而不会影响整个运算结果。

媒体关注与评论

“Joe Celko写的所有的书我都买了，这是其中写得最好的一本……强烈推荐I” ——Amazon com读者评论

<<SQL沉思录>>

编辑推荐

SQL是数据库的标准数据查询语言，在广大程序员的日常工作中已必不可少。
但是，大部分的SQL程序员都是由过程式编程人员和面向对象编程人员转变而来的，他们已经习惯了原有的思维方式，若要充分利用SQL必须学会以数据集的方式思考。
《SQL沉思录》中，世界级SQL专家、Joe Celko通过展示实用技术及思想方法，教会大家如何完成这个转变。
书中通过大量实例详细介绍了各种SQL编程技术，内容涉及查找表、辅助表、虚拟表、时态表、视图、SQL的思考方式等多个方面，值得所有SQL程序员仔细研读和思考。
世界级SQL专家经典著作　　深入揭示SQL编程本质　　教你不同于过程和OO编程的全新思考方式

版权说明