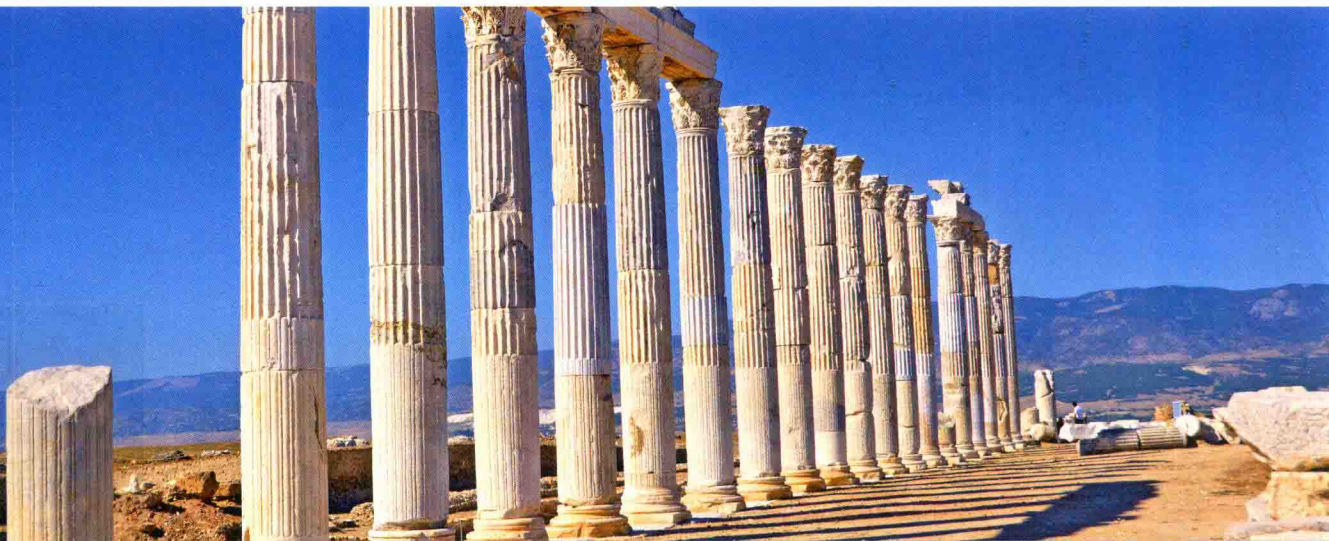


SQL专家近10年项目实施经验总结，深度解析SQL应用及常见使用误区。

结合大量实例全面剖析SQL知识。充分考虑兼容SQL Server数据库和Oracle数据库，
深入剖析索引及事务的原理。

SQL应用 及误区分析



张振磊 编著



机械工业出版社
China Machine Press

数据库 技术丛书

SQL应用 及误区分析



张振磊 编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

SQL 应用及误区分析 / 张振磊编著. —北京: 机械工业出版社, 2018.5
(数据库技术丛书)

ISBN 978-7-111-59730-8

I. S… II. 张… III. 关系数据库系统 IV. TP311.132.3

中国版本图书馆 CIP 数据核字 (2018) 第 080919 号

SQL 应用及误区分析

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 缪 杰

责任校对: 殷 虹

印 刷: 北京市兆成印刷有限责任公司

版 次: 2018 年 5 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 15.5

书 号: ISBN 978-7-111-59730-8

定 价: 59.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有 · 侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

作者简介



张振磊

现任创业软件股份有限公司技术总监。毕业于江南大学计算机科学与技术专业。先后获得软件设计师、系统架构师、Oracle专家等认证，并且满分通过了SQL认证。2009年加入创业软件，先后担任上海大区高级技术支持、山东区技术总监、北方大区技术总监、杭州区技术总监、浙闽大区技术总监，研发与实施医疗信息化项目百余次，包含上海市闵行区居民电子健康档案、杭州市社区卫生服务、滕州市人口健康档案等重大项目。他在近10年的一线项目实施过程中沉淀了许多宝贵的经验，勤劳务实的工作作风荣获滨江区“两新好人物”称号。

内容简介

本书对SQL相关知识由浅入深地进行了全面剖析，并且结合大量的实际操作对所有的知识点进行了举例论证，针对产品开发及项目实施中一些常见的使用误区也进行了深入的分析验证。本书既包含了基础的增删改查知识，又对视图、索引、约束、触发器、存储过程、函数及事务等知识进行了详细的讲解、举例及误区分析。出于对信息管理系统能适应多种关系型数据库的考虑，本书举例过程中同时兼容市场占有率比较高的关系型数据库SQL Server数据库和Oracle数据库，并且专门比较了SQL在SQL Server数据库与Oracle数据库中的差异。所有内容全部来自编者近10年的知识积累及数百次一线项目开发与实施经验的沉淀。

Foreword 推荐序

经过作者一年多的努力,《SQL 应用及误区分析》一书终于和广大读者见面了。

不知不觉,我们已经迈入大数据时代。当今,作为大数据源头,信息系统生产的数据量每年都在呈几何增长。由于 SQL 的不合理使用导致的信息系统性能问题困扰着众多 IT 从业者。本书正是基于这个背景来编写:不仅让读者学会 SQL 的使用,最大效能地用好 SQL 是本书的更进一步目的。

本书系统介绍了 SQL 的核心知识,既包含基础的增删改查,又包含视图、索引、约束、触发器、存储过程和函数等,同时详解了重要事务知识。本书有两大特色:一是对所有知识点进行了详细的举例论证;二是对常见的使用误区进行了深入解读。相信读者在以后的产品开发中能最大程度地避免不必要的认知误区和实践失误。

本书同时结合市场占有率较高的 SQL Server 数据库和 Oracle 数据库进行举例。对两种关系型数据库的差异性进行了专门和独到的比较,尤其让开发者在构建适合多种关系型数据库信息系统的过程中充分受益。

近些年随着大数据的崛起, NoSQL 数据库和 Hadoop 技术已经非常活跃,然而其主旨还是应对非结构化数据。SQL 技术在近 50 年的发展过程中已经深入人心,不仅没有没落,反而更受关注,更需要审视和进步。如今 NewSQL 的出现也标志着 NoSQL 与 SQL 已经从对立面走向融合体转变。SQL 作为关系型数据库的标准语言,应当被所有 IT 从业者所充分掌握。

本书作者张振磊是一个资深、严谨、钻研型的技术男,创新思维、推崇技术、善于分享、精益求精,这正是我推荐本书的原因。无论在数据库还是系统架构领域,作者都投入了很大的精力去探究实践,像他这样既能拿到 Oracle 认证专家又能拿到系统架构师认证、有着数百次一线项目开发和实施经验的年轻人,的确少数。本书完全就是 10 年间数百次项目实践经验沉淀的产物,着实宝贵。

希望本书能广受欢迎,给广大 IT 从业者带来兴奋愉悦、全新气息的学习资源。更期待这种学习资源能影响到更多的项目,体现更大的价值。

应晶

浙江大学教授

2018 年 3 月

前 言 *Preface*

在信息管理系统中，SQL 语句是非常重要的组成部分。虽然用户不会直接使用 SQL 语句操作信息管理系统，但是，信息管理系统必须使用 SQL 语句去响应用户的请求。作为信息管理系统的创造者和维护者，每一位相关 IT 人员都应该能熟练使用 SQL。随着社会的进步以及信息技术的革新，新的信息管理系统业务逻辑变得越来越复杂，业务数据量变得越来越庞大，SQL 语句应该引起信息技术从业者的足够重视。好的 SQL 语句能够帮助信息管理系统更稳健地运行，相反，差的 SQL 语句则将极大地降低信息管理系统运行的效率，从而影响用户的体验。

编者自 2008 年从江南大学计算机科学与技术专业毕业后，一直从事医院信息管理系统（HIS）的研发与技术支持工作。在近 10 年的项目实战中，遇到了无数次由于不合理地使用 SQL 语句而造成的各种问题。有些性能问题甚至直到系统运行数年，数据量达到一定程度，已经影响用户日常使用了才被发现。项目组中很多技术支持人员及工程人员对 SQL 语句没有足够的重视，以至于很多不合理的 SQL 语句年复一年地出现在软件产品中，降低了产品的质量。一个好的程序员不是熟练使用开发语言就可以了，还要对 SQL 语句乃至数据库知识有深入的了解。编者凭借多年的项目实战经验，并结合扎实的理论知识编写了此书，希望能够帮助更多的读者学会 SQL 语句，并能正确、高效地使用它，从而保障信息管理系统高质量、高效率地运行。

不同于一些纯理论书，本书在编写过程中一直秉承理论结合实践的原则来介绍 SQL 语句的应用。并且，用切身体会来分析平常遇到的一些 SQL 语句的使用误区，以便读者更好地理解 SQL 语句，也希望能够帮助读者在以后的工作中避免陷入误区。本书所讲的知识完全来自于编者近年来数百次的项目实战。由于本书是一本介绍 SQL 语句的书籍，一些举例尽量使用标准 SQL。标准 SQL 以外的 SQL 相关知识，分别以 SQL Server 数据库的 T-SQL 和 Oracle 数据库的 PL/SQL 来讲解。之所以选择 SQL Server 数据库和 Oracle 数据库来讲解，是因为在信息管理系统领域，SQL Server 数据库和 Oracle 数据库具有非常高的市场占有率，而且很多读者平时接触最多的也是 SQL Server 数据库和 Oracle 数据库。

本书讲解过程中用到的是 Oracle 数据库管理系统中 SCOTT 模式下的关系模型。该关系模型主要包含 4 张表，分别是 EMP（员工表）、DEPT（部门表）、SALGRADE（月薪等级表）以及 BONUS（奖金表）。SCOTT 模型是 Oracle 数据库安装过程中默认安装的一个非常精简并且容易理解的关系模型，非常适合 SQL 语句的学习。本书分别给出了这 4 张表在 SQL Server 数据库与 Oracle 数据库中的建表语句，以及基础数据导入的 SQL 脚本，后续 SQL 语句的举例也基本围绕这 4 张表展开。

本书共分为 14 章，由易到难，逐步讲解 SQL 语句的应用。其中，第 1 章是 SQL 概述；第 2 章简单介绍 SCOTT 模式；第 3~6 章分别介绍 SQL 常用的增删改查功能；第 7~12 章分别介绍常用的数据库对象，包含视图、索引、约束、触发器、存储过程和函数；第 13 章介绍非常重要的事务知识；第 14 章通过示例比较 SQL Server 数据库和 Oracle 数据库的差异。

本书在编写过程中得到了创业软件股份有限公司多位领导和同事的支持和帮助，感谢高级副总裁沈建苗、人力资源总经理于瑶以及研发中心各位同事。

由于编者的精力和水平有限，书中错误和疏漏之处在所难免，敬请广大读者批评指正。也可将问题以邮件形式发送到编者邮箱：zhangzl@bsoft.com.cn。

目 录 Contents

推荐序	
前言	
第1章 SQL 概述	1
1.1 DML	1
1.2 DDL	1
1.3 TCL	2
1.4 DCL	2
1.5 总结	2
第2章 SCOTT 模式	4
2.1 DEPT	5
2.2 EMP	7
2.3 SALGRADE	10
2.4 BONUS	11
2.5 总结	12
第3章 新增语句	13
3.1 单行新增	13
3.1.1 values 单行新增	14
3.1.2 select 单行新增	16
3.2 建表新增	18
3.3 查询结果新增	20
3.4 常见误区分析	21
3.4.1 历史数据转移引起的问题	21
3.4.2 values 单行新增不要省略列名	24
3.5 总结	25
第4章 删除语句	26
4.1 delete 语法	26
4.1.1 直接删除表中记录	26
4.1.2 基于其他表删除表中记录	27
4.2 truncate 语法	28
4.3 误删数据恢复	29
4.4 误删对象恢复	30
4.5 常见误区分析	31
4.5.1 慎用 delete	31
4.5.2 画蛇添足	32
4.6 总结	32
第5章 更新语句	33
5.1 update 语法	33
5.2 单表更新	33
5.3 表关联更新	35

5.4 常见误区分析.....	36	6.8.4 单行多列子查询.....	73
5.4.1 注意表关联更新.....	36	6.8.5 内联视图.....	74
5.4.2 注意数据类型.....	38	6.8.6 关联子查询.....	74
5.5 总结.....	39	6.9 别名.....	76
第6章 查询语句	40	6.9.1 表别名.....	77
6.1 查询语句的语法.....	40	6.9.2 列别名.....	77
6.2 where 子句中常用的运算符.....	41	6.10 常见误区分析.....	79
6.2.1 算术运算符.....	42	6.10.1 count 争议.....	79
6.2.2 逻辑运算符.....	43	6.10.2 null 的比较.....	80
6.2.3 比较运算符.....	44	6.10.3 单行子查询返回多行.....	82
6.2.4 优先级.....	50	6.10.4 分组函数的嵌套.....	83
6.3 分组.....	51	6.10.5 not in.....	84
6.3.1 分组函数.....	51	6.10.6 with (nolock).....	86
6.3.2 创建组.....	52	6.10.7 with (readpast).....	88
6.4 排序.....	52	6.10.8 max 用于字符型属性.....	89
6.5 空值.....	56	6.11 总结.....	91
6.6 多表连接.....	56	第7章 视图	92
6.6.1 交叉连接.....	57	7.1 视图语法.....	92
6.6.2 非等值连接.....	58	7.1.1 创建语法.....	92
6.6.3 等值连接之内连接.....	59	7.1.2 修改语法.....	93
6.6.4 等值连接之外连接.....	60	7.1.3 删除语法.....	93
6.6.5 等值连接之自连接.....	64	7.2 视图举例.....	94
6.6.6 等值连接之自然连接.....	66	7.3 视图的作用.....	95
6.7 集合运算.....	67	7.3.1 定制用户数据.....	95
6.7.1 并集.....	67	7.3.2 复杂查询简单化.....	96
6.7.2 交集.....	68	7.4 简单视图.....	97
6.7.3 差集.....	69	7.5 复杂视图.....	97
6.8 子查询.....	70	7.6 键值保存表.....	97
6.8.1 多行单列子查询.....	70	7.7 只读视图.....	98
6.8.2 多行多列子查询.....	72	7.8 with check option.....	98
6.8.3 单行单列子查询.....	72	7.9 物化视图.....	101

7.9.1 创建时生成数据选项	101	第 9 章 约束	136
7.9.2 刷新方式	102	9.1 约束语法	136
7.9.3 数据刷新的时间	102	9.1.1 创建语法	136
7.9.4 物化视图索引	103	9.1.2 删除语法	137
7.9.5 物化视图举例	103	9.2 主键约束	137
7.10 索引视图	106	9.3 外键约束	138
7.11 常见误区分析	107	9.4 唯一性约束	141
7.11.1 单张表组成的视图可以更新	107	9.5 非空约束	142
7.11.2 多张表组成的视图不能更新	109	9.6 check 约束	143
7.12 总结	111	9.7 默认值约束	144
第 8 章 索引	112	9.8 常见误区分析	146
8.1 索引语法	113	9.8.1 是否有必要使用外键	146
8.1.1 创建语法	113	9.8.2 程序校验代替检查约束	146
8.1.2 删除语法	113	9.9 总结	148
8.2 B-Tree 索引	113	第 10 章 触发器	149
8.3 聚集索引	122	10.1 触发器语法	149
8.4 唯一索引	122	10.1.1 创建语法	149
8.5 非唯一索引	124	10.1.2 修改语法	150
8.6 组合索引	124	10.1.3 删除语法	151
8.7 反向键索引	125	10.2 变异表	151
8.8 函数索引	125	10.3 触发器内置对象	151
8.9 索引组织表	127	10.4 行级触发器	151
8.10 常见误区	128	10.5 语句级触发器	153
8.10.1 null 全表扫描	128	10.6 触发时间	156
8.10.2 <> 比较符引起全表扫描	129	10.7 instead of 触发器	160
8.10.3 <or> 引起全表扫描	131	10.8 常见误区分析	162
8.10.4 函数造成全表扫描	131	10.8.1 读变异表	162
8.10.5 慎用全表扫描	132	10.8.2 触发器死循环	162
8.10.6 组合索引如何进行索引	133	10.9 总结	164
8.11 总结	135		

第 11 章 存储过程.....165

- 11.1 存储过程语法.....165
 - 11.1.1 创建语法.....166
 - 11.1.2 修改语法.....166
 - 11.1.3 删除语法.....167
- 11.2 IN 模式参数.....167
- 11.3 OUT 模式参数.....169
- 11.4 删除存储过程.....171
- 11.5 常见误区分析.....171
 - 11.5.1 存储过程事务控制.....171
 - 11.5.2 参数名称引发事故.....179
- 11.6 总结.....181

第 12 章 函数.....183

- 12.1 系统函数.....183
 - 12.1.1 字符函数.....183
 - 12.1.2 数值函数.....186
 - 12.1.3 日期函数.....186
 - 12.1.4 null 相关的函数.....189
 - 12.1.5 聚合函数.....190
 - 12.1.6 其他常用函数.....190
- 12.2 自定义函数.....191
 - 12.2.1 自定义函数语法.....191
 - 12.2.2 SQL Server 标量值函数.....193
 - 12.2.3 SQL Server 内联表值函数.....193
 - 12.2.4 SQL Server 多语句表值函数.....194
 - 12.2.5 Oracle 标量值函数.....194
 - 12.2.6 Oracle 表值函数.....195
- 12.3 常见误区分析.....196
 - 12.3.1 SQL 函数必须有返回值.....196
 - 12.3.2 SQL 函数中不能进行 DML

操作.....198

12.4 总结.....200

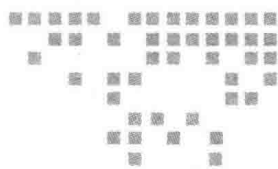
第 13 章 事务.....201

- 13.1 银行转账案例.....201
- 13.2 事务的 4 个属性.....204
 - 13.2.1 原子性.....205
 - 13.2.2 一致性.....205
 - 13.2.3 隔离性.....205
 - 13.2.4 持久性.....206
- 13.3 并发引起的问题.....207
 - 13.3.1 脏读.....207
 - 13.3.2 不可重复读.....207
 - 13.3.3 幻读.....207
- 13.4 事务隔离级别.....208
 - 13.4.1 读未提交.....208
 - 13.4.2 读提交.....208
 - 13.4.3 重复读.....208
 - 13.4.4 序列化.....209
- 13.5 事务保存点.....209
- 13.6 自治事务.....210
 - 13.6.1 自治事务用于存储过程.....210
 - 13.6.2 自治事务用于触发器.....212
- 13.7 常见误区分析.....214
 - 13.7.1 自治事务死锁.....214
 - 13.7.2 自治事务获取主事务的信息.....214
 - 13.7.3 主事务获取自治事务的信息.....216
- 13.8 总结.....217

第 14 章 SQL Server 与 Oracle 的差异.....219

- 14.1 前 N 行.....219

14.2	字符串拼接	220	14.10	对视图非键值保存表的更新	229
14.3	获取系统时间	221	14.11	分组函数嵌套	231
14.4	空字符串	221	14.12	内联视图	232
14.5	表别名	223	14.13	关联表删除	233
14.6	null 值排序	224	14.14	关联表更新	234
14.7	update 引起 select 阻塞	225	14.15	自增列	235
14.8	SQL、T-SQL 和 PL/SQL	227	14.16	总结	238
14.9	视图定义中出现排序	227			



SQL 概述

SQL 是结构化查询语言 (Structured Query Language) 的简称, 是一种关系型数据库操纵语言, 是所有关系型数据库都采用的标准语言。SQL 最早由 IBM 名士荣誉获得者 Don Chamberlin 于 1974 年提出。1979 年, 极具商业头脑的 Oracle 公司总裁 Larry Ellison 首先将 SQL 投入 Oracle 产品中, 从而使 SQL 具有了商业用途。1986 年, ANSI 将 SQL 作为关系型数据库的标准语言。1987 年, SQL 成为国际标准。至今, SQL 经久不衰。

SQL 语言可以分为 4 类, 分别是 DML (数据操纵语言)、DDL (数据定义语言)、TCL (事务控制语言) 和 DCL (数据控制语言)。接下来, 分别对这 4 类语言进行简单的介绍。

1.1 DML

对数据库中数据的操作无非以下几种: 新增数据到数据库中、删除数据库中不想继续保留的数据、修改数据库中已经存在的数据, 以及从数据库中检索数据。这些操作共同组成了数据操纵语言 (DML), 即我们平时所讲的增删改查 (CRUD)。DML 是使用频率最高的一类 SQL 语言。IT 从业者几乎天天跟 DML 打交道。这一部分内容也是与系统的业务逻辑关系最密切的, 所以 DML 语言是本书着重要讲的内容。

1.2 DDL

为了将数据有条不紊地保存到数据库中, 必须定义一定的规则。表、索引、视图、触

发器、存储过程、函数等对象共同组成了数据库中的数据结构，这些数据结构共同为数据服务。新建数据结构、修改已经存在的数据结构以及删除不再使用的数据结构几种操作共同组成了 DDL（数据定义语言）。由于现在常用的关系型数据库都自带管理工具，管理工具中包含了可视化操作界面，所以很多读者都是用可视化操作界面创建数据库对象的。一般地，对数据库操作熟练的 IT 人员都喜欢用 DDL 来创建数据库对象，毕竟命令行操作方式比可视化界面操作方式更快速、更方便。并且，可视化界面上的操作，最终还是要转换成 DDL 命令执行的。

当然了，这部分命令不是必须掌握的，因为前台可视化界面也可以进行操作，对于不擅长记忆命令的读者，前台可视化界面也是不错的选择。而且 DDL 远没有 DML 使用得频繁，很多工程人员甚至技术支持人员都很少使用 DDL 命令，但是研发人员接触 DDL 命令的机会还是很多的。

1.3 TCL

DML 只是用于数据的更改，真正管理数据更改的是 TCL，它决定了 DML 对数据的更改是提交到数据库还是回滚到修改之前的状态。而且，在大部分情况下，一个操作往往由多个 SQL 语句组成，并且，逻辑上要求这些 SQL 语句成为一个整体，要么同时提交，要么同时回滚。TCL 对数据更改的管理叫作事务控制。事务控制在复杂的业务逻辑中显得尤为重要。第 13 章将会详细介绍事务及 TCL 知识。

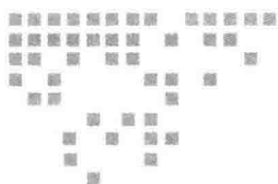
1.4 DCL

DCL 是用于权限控制的，负责对要访问数据库和数据库中数据对象的用户进行授权或回收权限。假如第三方公司要访问我们的数据库，出于安全考虑，我们一般会创建一个只有部分权限的用户给第三方公司使用。而且，随着需求的变更，我们也可能会收回部分原来分配给第三方公司的权限。像这种权限的分配、收回就由 DCL 操作。这部分命令使用得很少，但是，平时做接口的时候还是会用到的，所以，也需要掌握。如果有的读者从来没创建过用户，每次都是直接把自己的产品使用的用户名与密码提供给第三方公司使用，这就说明这些读者还没有意识到数据安全性的重要性。

1.5 总结

从 1974 年被 IBM 公司提出，SQL 至今已发展了 40 多年，作为关系型数据库采用的国际标准，它已被所有软件从业者所熟悉。程序员使用的开发语言可能不同，但是 SQL 却是一样的。

SQL 的关键字非常有限，语法也比较精简，很容易掌握，所以几乎所有程序员都会使用 SQL。随着社会的进步和信息技术的革新，数据量逐年呈指数级增长。很多旧系统性能变得越来越低，新开发的系统如果不加以注意，以后随着数据量的增长，势必也会引起系统性能的降低。当下，不是会用 SQL 就可以了，用好 SQL 已经成为程序员面临的新的挑战。



Chapter 2 第2章

SCOTT 模式

在 Oracle 数据库中，一个用户所拥有的所有对象的集合叫作一个模式。Oracle 数据库默认使用用户名作为该用户的模式名，所以 SCOTT 既是用户名又是模式名。Oracle 数据库中模式的概念对应于 SQL Server 数据库中的 DBO。SQL 语句的介绍肯定会用到数据库中的结构对象。本书在后续的讲解过程中选用了 Oracle 数据库的 SCOTT 模式作为举例的场景，该模式相对比较简单，而且容易理解。Oracle 数据库自带该模式，也是为了方便 Oracle 用户的学习与练习。

SCOTT 用户是 Oracle 数据库安装过程中默认安装的用户，该用户对应的 SCOTT 模式包含了 4 张表，分别是 EMP（员工表）、DEPT（部门表）、SALGRADE（月薪等级表）以及 BONUS（奖金表）。至于为什么取 SCOTT 这个名称，可能有的读者不知道原因。SCOTT 是 Oracle 公司成立之初的一位核心员工，他担任 Oracle 公司的分析师职务。Oracle 能够用 SCOTT 的名字来命名一个模式，说明在 Oracle 产品开发之初，SCOTT 充当了非常重要的角色。

记住名词的最好方法就是弄清楚它的含义。熟悉 Oracle 数据库的读者都知道，Oracle 数据库安装的时候默认使用 ORCL 作为服务名。可能一些比较懂 Oracle 数据库的读者也没有思考过，Oracle 数据库为什么选择 ORCL 而不是 Oracle 作为服务名。爱屋及乌，要熟悉一个产品，也要熟悉这个产品的生产厂家。ORCL 是 Oracle 公司的股票交易代码，知道这个原因，就不难理解 Oracle 公司为什么选用 ORCL 作为数据库的默认服务名了。

接下来分别介绍这 4 张表的含义，并将分别给出这 4 张表在 SQL Server 数据库和 Oracle 数据库中的建表语句（DDL）及表记录的导入脚本（DML）。导入脚本在本章不理解也可以，本章只是为了完成讲解环境的搭建。本章用到的插入命令在第 3 章中有详细的讲解，如果读者的计算机上安装过 Oracle 数据库的话，可以不执行本章的 SQL 命令，因为 Oracle 数据库已经默认安装了 SCOTT 模式，表结构有稍微变动的，可以只修改个别变动