

Microsoft®

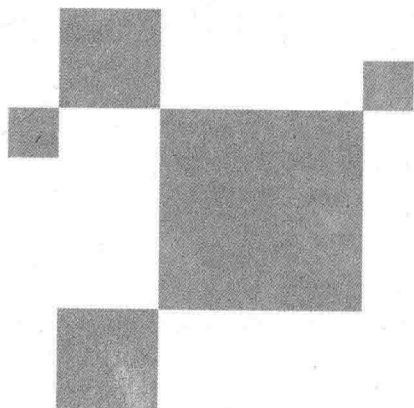
SQL Server 2008

数据库应用开发基础

微软公司 著

 **人民邮电出版社**
POSTS & TELECOM PRESS





SQL Server 2008

数据库应用开发基础

微软公司 著



人民邮电出版社

北京

图书在版编目 (CIP) 数据

SQL Server 2008数据库应用开发基础 / 微软公司著

— 北京 : 人民邮电出版社, 2010.11 (2011.1 重印)

ISBN 978-7-115-23342-4

I. ①S… II. ①微… III. ①关系数据库—数据库管理系统, SQL Server 2008 IV. ①TP311.138

中国版本图书馆CIP数据核字 (2010) 第174377号

版 权 声 明

本书的著作权归微软公司所有。未经微软公司书面许可, 本书的任何部分不得以任何形式或任何手段复制或传播。著作权人保留所有权利。

内 容 提 要

本书描述了实现 Microsoft SQL Server 2008 数据库所需要的知识和技能。全书共 14 章, 包括基本 T-SQL 查询语句的编写技能, 以及创建数据库和数据库文件、数据类型和表、索引、约束、触发器、XML 功能、视图、存储过程、函数、托管代码、管理事务和锁、使用 Service Broker 等知识。

本书适合需要熟练掌握 SQL Server 2008 数据库功能和技术来实施数据库的读者阅读。

SQL Server 2008 数据库应用开发基础

- ◆ 著 微软公司
责任编辑 刘 浩
- ◆ 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 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京隆昌伟业印刷有限公司印刷
- ◆ 开本: 787×1092 1/16
印张: 26.25 2010 年 11 月第 1 版
字数: 632 千字 2011 年 1 月北京第 2 次印刷

ISBN 978-7-115-23342-4

定价: 98.00 元 (附光盘)

读者服务热线: (010)67132692 印装质量热线: (010)67129223

反盗版热线: (010)67171154

编 委 会

编审和组织策划 Kyle Uphoff Foong Chee Ngiam 王 林

田本和 蒋 斌 梁 健

技术编审 蒋 斌 张 充 宫 丽 高 娟

前言

SQL Server 2008 是微软新一代的数据库管理系统，它提供一系列丰富的集成服务，可以对数据进行查询、搜索、同步、报告和分析等操作。SQL Server 2008 允许用户在使用 Microsoft .NET 和 Visual Studio 开发的自定义应用程序中使用数据，在面向服务的架构（SOA）和通过 Microsoft BizTalk Server 进行的业务流程中使用数据。信息工作人员可以通过他们日常使用的工具（如 2007 Microsoft Office 系统）直接访问数据。SQL Server 2008 提供一个可信的、高效率智能数据平台，可以满足您的所有数据需求。

通过学习本课程，读者可以掌握管理 SQL Server 2008 的实现技能包括：

- 常用 T-SQL 代码查询技能；
- 创建数据库和数据库文件；
- 创建数据类型和表；
- 规划、创建和优化索引；
- 使用约束在 Microsoft SQL Server 2008 数据库中实现数据完整性；
- 使用触发器在 Microsoft SQL Server 2008 中实现数据完整性；
- 在 Microsoft SQL Server 2008 中使用 XML 相关的功能；
- 实现视图；
- 实现存储过程；
- 实现函数；
- 在数据库中实现托管代码；
- 管理事务和锁；
- 使用 Service Broker 生成基于消息传递的解决方案。

本书适用对象

主要面向希望熟练掌握 SQL Server 2008 数据库功能和技术来实施数据库的读者。

本书结构

本书共分为 14 章，各章主要内容介绍如下。

第 1 章 数据库基础：介绍数据库的基础知识和 SQL Server 2008 的基础知识。

第 2 章 T-SQL 入门：介绍基本的 T-SQL 语句，以实例的方式帮助读者对 T-SQL 语言的基本用法建立感性认识。

第 3 章 创建数据库和数据库文件：介绍数据库开发人员必须执行的最基本的任务之一——创建数据库及其主要组件，如创建数据库、设置数据库选项、创建文件组、架构和数据库快照。

第 4 章 创建数据类型和表：介绍 SQL Server 2008 中系统提供的数据类型。学生将学习

如何定义自定义 Transact-SQL 数据类型，并描述如何创建表以及如何使用已分区表来将数据组织成多个分区。

- 第 5 章 使用 Transact-SQL 查询数据库：描述如何使用 T-SQL 语言实现数据库查询。
- 第 6 章 创建和优化索引：描述如何规划、创建和优化索引，以获得最佳性能益处。
- 第 7 章 使用约束和触发器实现数据完整性：说明如何使用约束在 SQL Server 2008 中实现数据完整性。他们还将使用触发器实现数据完整性。
- 第 8 章 使用 XML：描述如何使用 XML，包括使用 FOR XML 子句、OPENXML 函数、XQuery 表达式，以及 XML 本机数据类型。读者将学习在创建 XML 索引时必须考虑的注意事项，以及用来创建 XML 索引的语法；还将了解什么是 XML 架构和 XML 架构集合，以及如何使用它们来实现类型化 XML 数据。
- 第 9 章 实现视图：说明 Microsoft SQL Server 2008 中提供的不同类型的视图，这些视图提供了一种通过预定义查询访问数据的方法。
- 第 10 章 实现存储过程：描述如何设计和实现强制实施业务规则或数据一致性的存储过程，或者修改和维护其他开发人员编写的现有存储过程。
- 第 11 章 实现函数：讨论如何设计和实现强制实施业务规则或数据一致性的用户定义函数，或者修改和维护其他开发人员编写的现有函数。
- 第 12 章 在数据库中实现托管代码：描述如何使用托管代码实现数据库对象，如存储过程、用户定义数据类型、用户定义函数和触发器。
- 第 13 章 管理事务和锁：描述如何使用事务和 SQL Server 锁定机制来达到应用程序的性能和数据完整性要求。
- 第 14 章 使用 Service Broker：你将更多地了解 Service Broker、如何创建 Service Broker 对象，以及如何使用 Service Broker 来发送和接收消息。

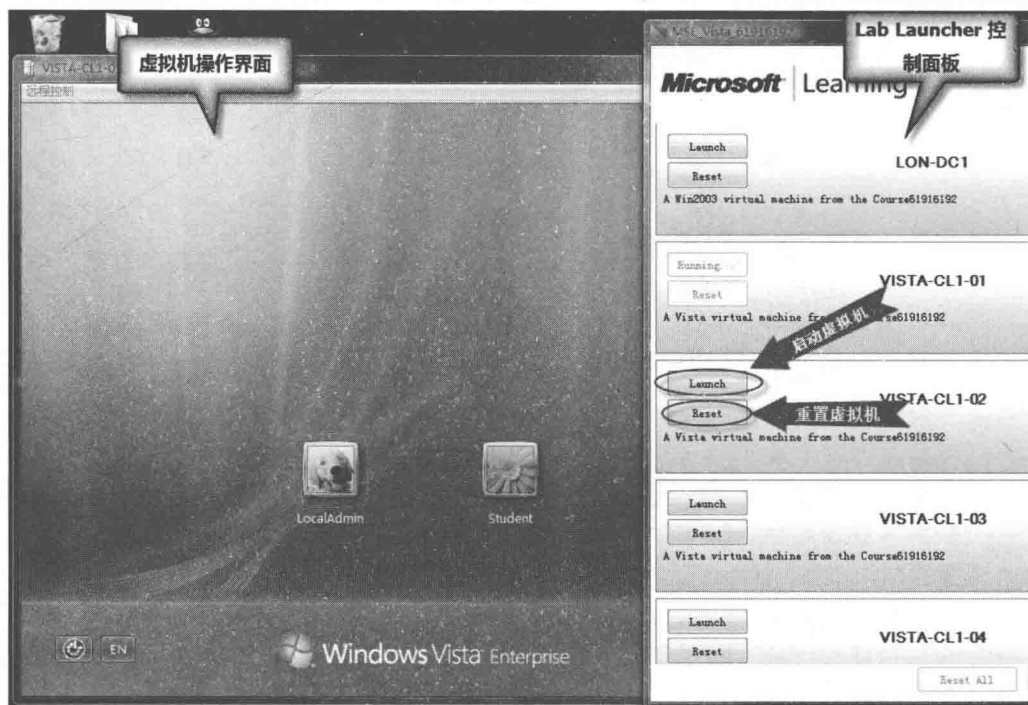
教学参考资料

获得微软授权使用该教材进行教学的教师，除了可以获得上述材料外，还可以从微软公司获得如下教学资料。

教参	章节教参与教学大纲
实验与习题答案	实验的具体操作步骤和习题答案
虚拟机	配置好的虚拟机实验环境，用于进行实验和课堂演示
实验室安装指南	包括在实验室部署虚拟机的指南
PowerPoint	用于进行课堂演示

虚拟机

本书案例所提供虚拟机实验均采用微软的 Lab Launcher 进行操作，Microsoft Lab Launcher 是一种虚拟机操作界面，它让用户用一种更简易的操作界面进行虚拟机操作。该工具基于 Virtual Server R2 SP1 实现。虚拟机安装完毕后，用户即可使用此界面进行操作。操作界面如图 0-1 所示。



0-1 虚拟机操作界面

Lab Launcher 的运行环境要求如下。

- 操作系统为 Windows XP 简体中文版或 Windows Vista 简体中文版。
- 系统中安装 .NET Framework 2.0 简体中文版。
- 系统中安装 Microsoft Virtual Server 2005 R2 简体中文版。
- 2GB 物理内存。

在本书中，共包含一套虚拟机，以完成实验。

虚拟机	角色
10082A-NY-SQL-01	Adventure Works 的 SQL Server

目 录

第 1 章 数据库基础 1

- 1.1 数据库基础 1
 - 1.1.1 数据与数据库 1
 - 1.1.2 表、行、列和约束 2
 - 1.1.3 在 Microsoft SQL Server 2008 中创建表 4
 - 1.1.4 创建表的步骤小结 7
 - 1.1.5 设置列属性和约束 7
 - 1.1.6 关系 8
 - 1.1.7 其他 SQL Server 2008 数据库对象 14
- 1.2 SQL Server 2008 概述 15
 - 1.2.1 SQL Server 2008 简介 15
 - 1.2.2 SQL Server 2008 的应用方式 16
 - 1.2.3 SQL Server 2008 的版本 17
 - 1.2.4 SQL Server 2008 的组件 17
 - 1.2.5 SQL Server 2008 管理工具 17
 - 1.2.6 SQL Server 2008 数据库引擎组件 20
 - 1.2.7 SQL Server 2008 的软硬件需求 21

第 2 章 T-SQL 入门 22

- 2.1 T-SQL 基础 22
 - 2.1.1 T-SQL 语句的种类 22
 - 2.1.2 T-SQL 查询工具 23
 - 2.1.3 使用查询编辑器输入和运行 T-SQL 代码 24
- 2.2 T-SQL 快速入门 25
 - 2.2.1 创建数据库 25
 - 2.2.2 创建表 25

- 2.2.3 插入和更新数据 26
- 2.2.4 读取表中的数据 27
- 2.2.5 创建登录名 28
- 2.2.6 授予访问数据库的权限 29
- 2.2.7 创建视图和存储过程 29
- 2.2.8 授予访问数据库对象的权限 30
- 2.2.9 删除数据库对象 31
- 2.3 Transact-SQL 语法要素 32
 - 2.3.1 批处理命令 32
 - 2.3.2 注释语句 33
 - 2.3.3 标识符 34
 - 2.3.4 数据类型 35
 - 2.3.5 变量 35
 - 2.3.6 几个常用的系统函数 36
 - 2.3.7 运算符 41
 - 2.3.8 表达式 43
 - 2.3.9 流控制的语言元素 43
 - 2.3.10 保留的关键字 45

第 3 章 创建数据库和数据库文件 46

- 3.1 创建数据库 46
 - 3.1.1 SQL Server 数据库类型 46
 - 3.1.2 数据库的文件组成 47
 - 3.1.3 数据存储 47
 - 3.1.4 创建数据库的注意事项 48
 - 3.1.5 事务日志 49
 - 3.1.6 数据库选项 51
 - 3.1.7 数据压缩 51
 - 3.1.8 演示：创建数据库 52
- 3.2 创建文件组 53
 - 3.2.1 文件组 53
 - 3.2.2 使用文件组提高数据库性能 54

3.2.3 演示: 创建文件组	57	4.3.4 可对分区数据执行的 操作	81
3.3 创建架构	57	4.3.5 演示: 创建已分区表	84
3.3.1 架构	57	4.4 实验: 创建数据类型和表	85
3.3.2 对象名称解析的 工作方式	59	4.4.1 实验 4-1: 创建数据类型	85
3.3.3 演示: 创建架构	60	4.4.2 实验 4-2: 使用新的 日期和时间数据类型	86
3.4 创建数据库快照	61	4.4.3 实验 4-3: 创建表	86
3.4.1 数据库快照的工作方式	61	4.4.4 实验 4-4: 创建已分 区表	87
3.4.2 何时使用数据库快照	62	4.4.5 关闭虚拟机	87
3.4.3 演示: 创建数据库快照	63	4.5 习题	88
3.5 实验: 创建数据库和数据库 文件	63	第 5 章 使用 Transact-SQL 查询数据库	91
3.5.1 实验 3-1: 创建数据库	64	5.1 检索数据	91
3.5.2 实验 3-2: 创建架构	64	5.1.1 使用 SELECT 语句检索 数据	91
3.5.3 实验 3-3: 创建数据库 快照	65	5.1.2 使用 WHERE 子句	92
3.5.4 关闭虚拟机	65	5.1.3 设置结果集格式	99
3.6 习题	66	5.2 数据分组与汇总	102
第 4 章 创建数据类型和表	67	5.2.1 使用 TOP n 列出前 n 个记录	102
4.1 创建数据类型	67	5.2.2 使用聚合函数	103
4.1.1 系统提供的数据类型	67	5.2.3 GROUP BY 的基础 知识	105
4.1.2 使用系统提供的数据 类型	68	5.2.4 联合使用 GROUP BY 子句和 HAVING 子句	107
4.1.3 别名数据类型	69	5.2.5 在结果集中生成汇总值	109
4.1.4 演示: 创建数据类型	70	5.2.6 使用 COMPUTE 和 COMPUTE BY 子句	112
4.2 创建表	71	5.2.7 推荐操作	114
4.2.1 SQL Server 在行中如何 组织数据	71	5.3 多表联接	114
4.2.2 SQL Server 如何组织大 数据值	72	5.3.1 使用表的别名	114
4.2.3 表的类型	73	5.3.2 组合多个表中的数据	115
4.2.4 创建表的注意事项	75	5.3.3 推荐操作	126
4.2.5 生成 Transact-SQL 脚本	76	5.4 子查询	126
4.2.6 演示: 创建表	77	5.4.1 子查询介绍	126
4.3 创建已分区表	78	5.4.2 把子查询用作派生表	127
4.3.1 已分区表	78	5.4.3 把子查询用作表达式	128
4.3.2 分区函数	79		
4.3.3 分区方案	81		

5.4.4 使用子查询关联数据	128	6.4.2 实验 6-2: 优化索引	164
5.4.5 推荐操作	133	6.4.3 关闭虚拟机	164
5.5 修改数据	133	6.5 习题	164
5.5.1 使用事务	133	第 7 章 使用约束和触发器实现	
5.5.2 插入数据	134	数据完整性	168
5.5.3 删除数据	137	7.1 数据完整性概述	168
5.5.4 更新数据	140	7.1.1 数据完整性类型	168
5.5.5 性能考虑	142	7.1.2 强制数据完整性的选项	169
5.5.6 推荐操作	142	7.2 实现约束	170
5.6 习题	142	7.2.1 约束	170
第 6 章 创建和优化索引	145	7.2.2 键约束	171
6.1 规划索引	145	7.2.3 其他约束	173
6.1.1 SQL Server 如何访问 数据	145	7.2.4 级联引用完整性	175
6.1.2 堆	146	7.2.5 约束检查注意事项	176
6.1.3 聚集索引	148	7.2.6 演示: 创建约束	178
6.1.4 非聚集索引	149	7.3 实现触发器	179
6.2 创建索引	151	7.3.1 触发器	179
6.2.1 创建索引概述	151	7.3.2 INSERT 触发器的 工作方式	181
6.2.2 唯一索引	153	7.3.3 DELETE 触发器的 工作方式	182
6.2.3 创建多列索引的注意 事项	154	7.3.4 UPDATE 触发器的 工作方式	183
6.2.4 何时创建计算列的 索引	155	7.3.5 INSTEAD OF 触发器的 工作方式	184
6.2.5 已分区索引	155	7.3.6 嵌套触发器工作方式	185
6.2.6 在索引中并入可用 空间的选项	156	7.3.7 递归触发器的注意事项	186
6.2.7 获取索引信息的方法	157	7.3.8 演示: 创建触发器	187
6.2.8 演示: 创建索引	158	7.4 实验: 使用约束和触发器实现 数据完整性	187
6.3 优化索引	158	7.4.1 实验 7-1: 创建约束	187
6.3.1 数据库引擎优化顾问	159	7.4.2 实验 7-2: 禁用约束	188
6.3.2 演示: 使用数据库 引擎优化顾问	159	7.4.3 实验 7-3: 创建触发器	189
6.3.3 索引碎片	160	7.4.4 关闭虚拟机	190
6.3.4 索引碎片整理选项	162	7.5 习题	190
6.3.5 演示: 索引碎片整理	163	第 8 章 使用 XML	193
6.4 实验: 创建和优化索引	163	8.1 使用 XML 数据类型	193
6.4.1 实验 6-1: 创建索引	163		

- 8.1.1 XML193
- 8.1.2 XML 数据类型194
- 8.1.3 Query、Value 和 Exist
方法195
- 8.1.4 Modify 方法196
- 8.1.5 Nodes 方法196
- 8.1.6 演示：使用 XML 数据
类型197
- 8.2 使用 FOR XML 检索 XML199
 - 8.2.1 FOR XML 子句简介199
 - 8.2.2 RAW 模式查询201
 - 8.2.3 AUTO 模式查询203
 - 8.2.4 EXPLICIT 模式查询205
 - 8.2.5 PATH 模式查询207
 - 8.2.6 检索嵌套 XML 的语法208
 - 8.2.7 演示：使用 FOR XML209
- 8.3 使用 OPENXML 分解 XML210
 - 8.3.1 分解 XML 数据概述210
 - 8.3.2 管理内存中节点树的
存储过程211
 - 8.3.3 OPENXML 语法213
 - 8.3.4 处理 XML 命名空间的
语法215
 - 8.3.5 演示：使用 OPENXML
分解 XML216
- 8.4 实验：使用 XML217
 - 8.4.1 实验 8-1：映射关系
数据和 XML217
 - 8.4.2 实验 8-2：将 XML 本机
存储在数据库中221
- 8.5 XQuery 简介223
 - 8.5.1 XQuery223
 - 8.5.2 XQuery 基础知识224
 - 8.5.3 XQuery 表达式225
 - 8.5.4 演示：使用 XQuery
表达式225
- 8.6 创建 XML 索引226
 - 8.6.1 XML 索引226
 - 8.6.2 XML 索引的优点226

- 8.6.3 XML 索引的类型 227
- 8.6.4 演示：创建 XML 索引 228
- 8.7 实现 XML 架构 229
 - 8.7.1 XML 架构 230
 - 8.7.2 XML 架构验证 232
 - 8.7.3 XML 架构集合 232
 - 8.7.4 类型化和非类型化
XML 234
 - 8.7.5 演示：使用类型化
XML 235
- 8.8 实验：使用 XML 235
 - 8.8.1 实验 8-3：结合 XML 方法
使用 XQuery 236
 - 8.8.2 实验 8-4：创建 XML
索引 237
 - 8.8.3 关闭虚拟机 238
- 8.9 习题 238

第 9 章 实现视图 243

- 9.1 视图简介 243
 - 9.1.1 视图 243
 - 9.1.2 视图类型 244
 - 9.1.3 视图的优点 246
- 9.2 创建和管理视图 246
 - 9.2.1 创建视图的语法 246
 - 9.2.2 演示：创建视图 248
 - 9.2.3 更改和删除视图的
语法 249
 - 9.2.4 视图加密 251
 - 9.2.5 所有权链如何影响
视图 252
 - 9.2.6 视图信息来源 253
 - 9.2.7 在视图中修改数据的
注意事项 254
- 9.3 使用视图优化性能 255
 - 9.3.1 视图的性能注意事项 255
 - 9.3.2 索引视图的性能注意
事项 256
 - 9.3.3 分区视图 257

9.4 实验: 实现视图	258	10.4.3 演示: 处理异常	291
9.4.1 实验 9-1: 创建视图	258	10.5 实验: 实现存储过程	292
9.4.2 实验 9-2: 创建索引 视图	259	10.5.1 实验 10-1: 创建存储 过程	292
9.4.3 实验 9-3: 创建分区 视图	260	10.5.2 实验 10-2: 处理执行 计划	294
9.4.4 关闭虚拟机	261	10.5.3 关闭虚拟机	294
9.5 习题	261	10.6 习题	295
第 10 章 实现存储过程	265	第 11 章 实现函数	297
10.1 使用存储过程	265	11.1 函数简介	297
10.1.1 什么是存储过程	265	11.1.1 函数类型	297
10.1.2 创建存储过程的语法	267	11.1.2 标量函数	298
10.1.3 创建存储过程的准则	271	11.1.3 内联表值函数	299
10.1.4 演示: 创建存储过程	272	11.1.4 多语句表值函数	300
10.1.5 更改存储过程的语法	273	11.1.5 演示: 创建函数	302
10.1.6 删除存储过程的语法	274	11.2 使用函数	302
10.1.7 演示: 更改和删除 存储过程	275	11.2.1 确定性函数和非确定性 函数	303
10.1.8 演示: 切换执行上 下文	275	11.2.2 创建函数的准则	304
10.2 创建带参数的存储过程	277	11.2.3 将存储过程重写为 函数	304
10.2.1 存储过程参数	277	11.3 控制执行上下文	305
10.2.2 演示: 创建带参数的 存储过程	280	11.3.1 执行上下文	305
10.2.3 表值参数	281	11.3.2 EXECUTE AS 子句	307
10.2.4 演示: 使用表值参数	282	11.3.3 扩展模拟上下文	308
10.3 处理执行计划	283	11.4 实验: 实现函数	310
10.3.1 执行计划	283	11.4.1 实验 11-1: 创建函数	310
10.3.2 查看执行计划	284	11.4.2 实验 11-2: 控制执行 上下文	313
10.3.3 执行计划缓存	285	11.4.3 关闭虚拟机	316
10.3.4 查询编译	287	11.5 习题	316
10.3.5 强制存储过程重新 编译	288	第 12 章 在数据库中实现托管代码	318
10.3.6 演示: 使用执行计划	288	12.1 SQL Server 公共语言运行时 简介	318
10.4 处理异常	289	12.1.1 .NET Framework 简介	318
10.4.1 结构化异常处理的 语法	289	12.1.2 .NET 公共语言运行时	324
10.4.2 异常处理准则	290	12.1.3 托管代码的优点	325

12.1.4	CLR 托管代码数据库对象	325	13.2.6	受限语句	349
12.1.5	托管代码和 Transact-SQL	326	13.3	理解 SQL Server 锁定体系结构	350
12.2	导入和配置程序集	326	13.3.1	可通过锁防止的并发问题	350
12.2.1	程序集	327	13.3.2	可锁定资源	351
12.2.2	程序集信任级别	327	13.3.3	锁类型	352
12.2.3	导入程序集的语法	328	13.3.4	锁兼容性	354
12.2.4	演示: 导入和配置程序集	328	13.4	管理锁	355
12.3	创建托管数据库对象	329	13.4.1	会话级锁定选项	355
12.3.1	托管数据库对象概述	329	13.4.2	锁升级	357
12.3.2	托管存储过程、触发器和函数	330	13.4.3	动态锁定	357
12.3.3	托管聚合和用户定义类型	330	13.4.4	死锁	358
12.3.4	演示: 创建托管数据库对象	331	13.4.5	查看锁定信息的方法	359
12.4	实验: 在数据库中实现托管代码	332	13.4.6	演示: 查看锁定信息	360
12.4.1	实验 12-1: 导入程序集	333	13.5	实验: 管理事务和锁	361
12.4.2	实验 12-2: 创建托管数据库对象	333	13.5.1	实验 13-1: 使用事务	361
12.4.3	关闭虚拟机	335	13.5.2	实验 13-2: 管理锁	362
12.5	习题	335	13.5.3	关闭虚拟机	363
第 13 章	管理事务和锁	337	13.6	习题	363
13.1	事务和锁概述	337	第 14 章	使用 Service Broker	366
13.1.1	事务	337	14.1	Service Broker 概述	366
13.1.2	锁	338	14.1.1	Service Broker	366
13.1.3	并发控制	339	14.1.2	Service Broker 系统体系结构	367
13.1.4	演示: 介绍事务	340	14.1.3	Service Broker 会话体系结构	368
13.2	管理事务	341	14.1.4	Service Broker 会话过程	369
13.2.1	自动提交事务	341	14.1.5	Service Broker 会话优先级	371
13.2.2	显式事务	342	14.1.6	Service Broker 安全体系架构	374
13.2.3	隐式事务	345	14.1.7	Management Studio 增强功能	376
13.2.4	事务恢复	347	14.2	创建 Service Broker 对象	376
13.2.5	使用事务的注意事项	348	14.2.1	启用 Service Broker	377
			14.2.2	外部激活支持	378

14.2.3	创建消息类型的语法 ...	379	14.3.4	发送/接收消息的诊断工具	396
14.2.4	创建约定的语法	381	14.4	实验: 使用 Service Broker	397
14.2.5	创建队列的语法	383	14.4.1	实验 14-1: 创建 Service Broker 对象	397
14.2.6	创建服务的语法	387	14.4.2	实验 14-2: 实现起始服务	399
14.2.7	演示: 创建 Service Broker 对象	388	14.4.3	实验 14-3: 实现目标服务	400
14.3	发送和接收消息	389	14.4.4	关闭虚拟机	401
14.3.1	发送消息的语法	389	14.5	习题	401
14.3.2	接收消息的语法	392			
14.3.3	演示: 发送和接收消息	395			

第 1 章 数据库基础

在现今的世界中，数据无处不在。生活在互联网时代，人们已经习惯和各种各样的数据打交道。无论是在网上书店下一个订单然后收到发货通知，还是在电子地图中输入出发地和目的地得到一个最优的乘车路线。几乎一天中的大部分时间，人们都面对数据的输入和输出。在人们熟悉的网上书店，电子地图等系统的背后，数据库为其提供数据存取的支持，而管理诸多数据库的则是数据库管理系统。数据库管理系统已经成为各种系统不可或缺的一部分。本书所介绍的 SQL Server 2008 是微软公司发布的数据库管理系统。本章将从数据库基础知识出发逐步进入 SQL Server 2008 的介绍。

1.1 数据库基础

在正式进入 SQL Server 2008 数据库管理系统学习之前，首先需要对数据库基础知识有所了解。在下面的内容中，将帮助读者了解数据库及其组成，并在 SQL Server 2008 上实践建立一个数据库来进行快速入门。

1.1.1 数据与数据库

数据是对真实世界的抽象，也是信息的载体。数据“张三”代表的是某一个人的名字，而年龄“50”是他的岁数。单个的数据所表示的信息有限，很多数据组织在一起就能表达丰富的信息。例如“张三”，“男”，“1960/7/1”出生，职称“工程师”。通过这些数据集合在一起，就可以看出张三的大致情况。以上仅仅是张三一个人的数据。若要反映整个技术科科室成员的数据该如何组织数据呢？很简单，那就是制作一张表，如表 1-1 所示。

表 1-1 技术科成员信息

姓名	性别	出生日期	职称
张三	男	1960/7/1	工程师
李四	女	1977/6/20	工程师助理
王五	男	1986/3/1	技术员

这张表实在是非常简单的一张表，提供的信息有限，若是想要了解工资情况，就需要再加一栏；若是再要了解婚姻状况就还需要再增加一栏。需要的信息是相当多的，但表格不可能无限延展。解决这一问题的方法是使用多张表格进行联合查找。例如下面的这三张表格。

这样要知道李四的年龄，职称和工资，就可以先查表 1-2 获得姓名、年龄和职称代码以及工号，然后根据职称代码从表 1-3 获得职称名字，再根据工号查表 1-4 得李四的工资。如此，诸多零零总总的表汇集在一起，就好像仓库一样。数据库是按一定的结构和规则组织起

来的相关数据的集合，是综合各用户数据形成的数据集合，是存放数据的仓库。

表 1-2 员工基本信息表

工号	姓名	性别	出生日期	职称代码
001	张三	男	1960/7/1	Engineer
002	李四	女	1977/6/20	Engineer Assistant
003	王五	男	1986/3/1	Techonlogist
004	赵六	女	1975/11/20	Engineer

表 1-3 职称表

职称代码	职称
Engineer	工程师
Engineer Assistant	工程师助理
Techonlogist	技术员

表 1-4 工资表

工号	工资
001	5400
002	4800
003	2500
004	5500

自数据库这一概念一经提出，先后出现了几种数据模型。其中基本的数据模型系统有 3 种：层次模型系统、网络模型系统和关系模型系统。20 世纪 60 年代末提出的关系模型具有数据结构简单灵活、易学易懂且具备雄厚的数学基础等特点。关系模型从 20 世纪 70 年代开始流行，至今已成为数据库的标准。上面所举的例子中，数据库中有许多互相关联的表，这就是一种典型的关系数据库。关系数据库是将数据存储在多个互相关联的表中的数据库。通常关系数据库中的表拥有一多关系。下面将以 SQL Server 2008 为实例，对关系数据库中的一些基本概念进行阐述。

1.1.2 表、行、列和约束

数据库是表的集合，而表是数据的集合。每个表包含了用户所感兴趣的对象的信息。如图 1-1 所示，一个企业的员工表中可以包含员工姓名、员工职位及员工性别，也可以保存员工出生日期、聘用日期等数据。

	员工编号	员工姓名	员工职位	员工出生日期	员工性别	员工聘用日期
	1	王海	经理	1970-2-26 0:00:00	男	1999-9-1 0:00:00
	2	刘虹	经理助理	1980-12-12 0:0...	女	2005-1-19 0:00:00
	3	张宇	职员	1975-6-28 0:00:00	男	2005-2-8 0:00:00
	4	陈东	职员	1972-1-1 0:00:00	女	2000-5-15 0:00:00

图 1-1 员工表中的行

表中的每一行称作记录或实体，表示一个具体的对象，如图 1-1 黑框中的内容所示。每个表包含若干条记录。在图 1-1 中，员工表中的每行只保存一名员工的详细信息，这些信息不会在表中的其他地方重复出现，因此当定位到某个特定的行时，就可以获得相应员工的所有信息。

表中的每一列称作属性或字段，表示对象的属性，如图 1-2 黑框中的内容所示。每列包含由列名所表示的一部分信息。如同表名一样，列名也应当表述明确。在图 1-2 的示例中，所有列名均以“员工”二字开头。比如使用“职位”作为列名，就不如“员工职位”含义明确。列名“员工职位”可以很明确地表示该列所包含的内容。

	员工编号	员工姓名	员工职位	员工出生日期	员工性别	员工聘用日期
	1	王海	经理	1970-2-26 0:00:00	男	1999-9-1 0:00:00
	2	刘虹	经理助理	1980-12-12 0:0:0...	女	2005-1-19 0:00:00
▶	3	张宇	职员	1975-6-28 0:00:00	男	2005-2-8 0:00:00
	4	陈东	职员	1972-1-1 0:00:00	女	2000-5-15 0:00:00

图 1-2 员工表中的列

为了能方便地从数据表中检索信息，每个表必须包含可唯一标识表中任何数据的单列或多列的组合。当构建数据库表时，将此列或列的组合标识为表的主键。例如，在员工表的例子中，因为员工编号是不可重复的，“员工编号”列就是主键。可以根据员工编号唯一地找到某位员工及其记录。一旦有了主键，只要知道表名、所在行的主键值以及列名，就可以很容易地访问数据库中的任何数据了。假设现在需要查找张宇的出生日期。为了获取该信息，首先找到员工表，然后根据员工编号查找张宇所在的行，最后就得到该行员工出生日期列的值，如图 1-3 所示。

	员工编号	员工姓名	员工职位	员工出生日期	员工性别	员工聘用日期
	1	王海	经理	1970-2-26 0:00:00	男	1999-9-1 0:00:00
	2	刘虹	经理助理	1980-12-12 0:0:0...	女	2005-1-19 0:00:00
▶	3	张宇	职员	1975-6-28 0:00:00	男	2005-2-8 0:00:00
	4	陈东	职员	1972-1-1 0:00:00	女	2000-5-15 0:00:00

图 1-3 员工张宇的出生日期

每个表都应包含一个主键。主键并不一定总是编号（虽然在简单的表中经常使用它），但是必须保证主键列的每个值都是唯一的。人员的姓氏或出生日期不适合用做主键，因为某些人可以有相同的姓氏或相同的出生日期。对于一张不包含主键的表，数据库服务器会因无法识别而返回错误的一行或多行。例如，如果信用卡号并非唯一但却被用做主键，客户可能收到错误的账单或为其他人的消费行为付费。因此，主键必须是唯一的。

也可以使用列的组合而非一列作为主键。如果主键仅是一列，则称为简单主键。如果主键由两列或更多列组成，则称为复合主键。例如，如果无法通过唱片名称唯一标识唱片集，则应设置表的主键为同时包含乐队名称和唱片名称。

数据库中还引入了约束这一机制来保证存储于数据库中的数据合法性、一致性和准确性。约束定义了列中允许值的设置规则，确保在列中输入有效的数据值，并且维护各个表之间的关联。