



SQL Server 数据库 技术 与应用

主编 孔庆月 王彦新

 北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS

SQL Server 数据库 技术 与应用

主 编 孔庆月 王彦新
副主编 范丽萍 王 瑞
周 颖 吴小华

图书在版编目 (CIP) 数据

SQL Server 数据库技术与应用/孔庆月, 王彦新主编. —北京: 北京理工大学出版社, 2012. 8

ISBN 978 - 7 - 5640 - 6156 - 2

I. ①S… II. ①孔…②王… III. ①关系数据库系统 - 数据库管理系统 - 高等学校 - 教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字 (2012) 第 136569 号

出版发行 / 北京理工大学出版社

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010)68914775(办公室) 68944990(批销中心) 68911084(读者服务部)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 北京正合鼎业印刷技术有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 15

字 数 / 412 千字

版 次 / 2012 年 8 月第 1 版 2012 年 8 月第 1 次印刷

责任编辑 / 陈莉华

印 数 / 1 ~ 1500 册

责任校对 / 周瑞红

定 价 / 49.00 元

责任印制 / 王美丽

前言

Preface <<< <<<

我国“十二五”规划纲要中指出,全面提高信息化水平和大力发展新一代信息技术产业。提高信息化水平要进行海量的信息处理,在信息处理中数据库技术占据非常重要的位置。数据库技术是当前 IT 领域中发展最活跃,应用最广泛的计算机技术之一,它的应用已渗透到社会的各个领域,如各国家机关、企业、事业和商业单位的管理信息系统。因此,掌握好数据库技术可以大大提升个人在信息处理方面的能力。

为秉承高等教育指导思想和办学宗旨,配合教育部做好示范性和骨干高等院校建设,遵循高技能人才培养的特点和规律,通过对智联招聘网站(www.zhaopin.com)搜索“程序员”岗位统计显示:据不完全统计,60%是招聘 Web 程序员,任职要求中大都涉及了对数据库技术的要求(熟悉 SQL 语言,熟悉 SQL Server、MySQL、Oracle 等数据库的开发,熟悉存储过程的编写与调试),并经过大量的社会用人单位的调研和已就业毕业生的回访,我们总结近二十年的数据库技术教学经验编写了这一基于典型工作过程的单元化教学、任务驱动模式教材。教材由浅入深将一个简化的接近于学生实际学习生活的数据库应用系统实例——“学生管理系统”分解到各个教学单元中,使学生对关系数据库理论知识和技术应用的学习易于理解和掌握。教材最后一个教学单元以 Microsoft Visual Studio 中的 C#语言这一功能强大的编程利器为前台开发环境,以 Microsoft SQL Server 这一当前流行的数据管理系统软件建立并管理后台数据库介绍了数据库应用系统的典型开发过程。

数据库依据其所依赖的数据模型可分为关系数据库、面向对象的数据库、层次型数据库、网状数据库等,而其中的关系数据库是当前社会中应用最为广泛的数据库。教材主要讲解关系数据库理论知识和微软公司的 SQL Server 这一关系数据库管理系统软件的应用。教材内容组织形式基于单元化教学和任务驱动模式,尽量压缩理论知识(以够用为原则),加强数据技术知识的内容,注重学生数据库技术应用能力的培养,并配有和课程内容相配套的习题和实训。

本教材由孔庆月、王彦新担任主编,范丽萍、王瑞、周颖、吴小华担任副主编,这些教师均为一线教学老师,有着多年的丰富教学经验。编写分工为:孔庆月编写单元三、五、六以及附录 A、附录 B,王彦新编写单元十一,范丽萍编写单元八、九,王瑞编写单元四、十,周颖编写单元一、二,吴小华编写单元七。在本书编写过程中,得到了河北新龙科技有限公司张伟先生的技术指导和建议,在此表示衷心的感谢。

由于编者水平有限,加之时间仓促,教材中难免有疏漏和不当之处,恳请广大读者和同仁批评指正。您所有的宝贵意见和建议请发往:kqy@netease.com。

编者

目录

Contents <<< <<<

单元一 创建关系数据库	1
任务 1 初步认识关系数据库	1
任务 2 启动 SQL Server 2005 数据库管理系统	3
任务 3 创建关系数据库	6
任务 4 创建数据表	10
实训一 创建数据库	17
实训二 认识 C/S 架构	17
单元二 关系数据库日常管理	24
任务 1 备份与还原数据库	24
任务 2 分离与附加数据库	29
任务 3 修改数据库名	32
任务 4 修改数据表	32
任务 5 使用对象资源管理器删除数据表中的数据行	35
任务 6 使用对象资源管理器删除数据表	35
任务 7 使用对象资源管理器删除数据库	36
任务 8 给表创建主键和索引	37
实训三 练习数据库的备份与还原操作	41
实训四 练习数据库的分离与附加操作	41
单元三 利用标准 SQL 语句对数据库操作	43
任务 1 认识标准 SQL 语言	43
任务 2 利用 SELECT 语句从数据表中提取数据	43
任务 3 利用 INSERT 语句向数据表中插入数据	53
任务 4 利用 DELETE 语句在数据表中删除数据	54
任务 5 利用 UPDATE 语句更新(修改)数据表中的数据	55
任务 6 利用数据定义语言创建和删除数据库	55
任务 7 认识数据控制语言	58
实训五 简单查询	59
实训六 多表连接查询	59
实训七 嵌套和相关查询	60
实训八 数据的插入、更新、删除	60
单元四 创建并利用视图操作数据库	63
任务 1 认识视图	63

任务 2 使用命令方式创建视图	64
任务 3 使用命令方式管理视图	65
任务 4 使用对象资源管理器管理视图	65
实训九 视图的创建及应用	67
单元五 Transact-SQL 语言	69
任务 1 认识 Transact-SQL 语言	69
任务 2 系统内置函数的认识及其应用	75
任务 3 程序控制语句及其应用	87
任务 4 用户自定义函数	93
任务 5 事务及其应用	100
任务 6 游标及其应用	106
实训十 Transact-SQL 程序设计	110
单元六 存储过程的设计及应用	114
任务 1 认识存储过程	114
任务 2 创建和执行存储过程	116
任务 3 存储过程的管理与维护	121
实训十一 存储过程设计	124
单元七 创建及应用触发器	128
任务 1 认识触发器	128
任务 2 创建和应用 DML 触发器	129
任务 3 触发器的管理与维护	134
任务 4 DDL 触发器及其应用	135
实训十二 触发器设计	138
单元八 关系数据库设计	140
任务 1 认识现实世界	140
任务 2 用 E-R 模型图描述现实客观世界	141
任务 3 E-R 模型图向关系模型的转化	144
任务 4 深入理解关系数据库	145
实训十三 成绩管理数据库设计	148
实训十四 网上论坛数据库设计	149
单元九 优化数据库	151
任务 1 理解函数依赖的概念	151
任务 2 理解范式的概念	152

单元十 SQL Server 2005 的安全管理	154
任务 1 认识 SQL Server 2005 的安全模型	154
任务 2 认识 SQL Server 访问控制	156
任务 3 管理 SQL Server 数据库引擎的身份验证模式	157
任务 4 管理 SQL Server 数据库引擎的安全连接	158
任务 5 数据库的安全性管理	170
任务 6 认知数据库用户角色	172
任务 7 权限管理	175
实训十五 数据库安全管理	180
单元十一 项目开发实例——学生管理系统	182
任务 1 项目引入	182
任务 2 数据库结构设计与实现	183
任务 3 创建项目	184
任务 4 设计与实现主窗体	185
任务 5 设计与实现课程信息管理窗体	186
任务 6 设计与实现学生信息管理窗体	193
任务 7 设计与实现成绩管理窗体	200
附录 A SQL Server 2005 的安装	209
任务 1 认识 SQL Server 2005	209
任务 2 认识 SQL Server 2005 的运行环境	210
任务 3 SQL Server 2005 的安装和配置	211
任务 4 认识 SQL Server 2005 的主要管理工具	216
任务 5 配置 SQL Server 2005	220
实训十六 SQL Server 2005 数据库服务器的安装与配置	224
附录 B 关系数据库的数学基础	227
任务 1 理解传统的集合运算	227
任务 2 专门的关系运算	230
参考文献	232

本单元初步学习有关关系数据库的基本概念和术语，使读者对关系数据库先有个初步的感性认知，并掌握利用 SQL Server 2005 数据库管理系统创建关系数据库的基本技能。

教学要求：通过本单元的学习，读者应该重点掌握以下内容：

- 关系数据库基本概念和术语。
- SQL Server 2005 数据库管理系统的基本操作。
- 关系数据库的创建。
- 数据表的创建。

任务 1 初步认识关系数据库

一、关系

要想了解关系数据库（Relation database），首先从感性认知的角度出发，对关系有个初步的认识，请来观察下面的一张描述有关学生信息情况的表格（Table）——学生表。

表 1.1 学生表

学号	姓名	性别	出生日期	身份证号	班级名	系别	QQ	总学分
2009010101	张三多	男	1981-02-19	130102198102191179	网络 0801	计算机系	66666666	20
2009010102	李四龙	男	1981-06-20	120102198106201176	网络 0801	计算机系	99999999	19
2009010103	王建岭	男	1981-08-26	130112198108261178	网络 0801	计算机系		21

表 1.1 有以下 6 个特征。

（1）表格的列（Column）数和行（Row）数确定，表中每一列都必须是不可分解的，列具有原子性（即不可分解性）。

（2）每列信息的性质是相同的，即每一列中的信息是同一类型的数据，它们来自同一个域。

（3）表中的任意两个行不能完全相同（否则数据会有重复）。

（4）表中不同的列来自不同的域，每一列有不同的列名（否则数据会有重复）。

（5）表中列的顺序可以任意互换，不会改变表的意义。

（6）表中行除首行外，其余行的次序也可以任意互换，不会改变表的意义。

像表 1.1 这样的一张表，就称为一个关系。非常接近于 Excel 工作簿文件中的工作表

(Sheet)。一张表只有同时具备了以上 6 个特征才能称为一个关系。

表 1.2 则不能称为一个关系，因为其行数和列数不确定，“收入”这一列又细分为“基本工资”“津贴”“其他”“水电费”和“实际收入”5 列，违反了前述的 6 个特征中第 (1) 条特征的要求。

表 1.2 工资表

职工号	姓名	收 入				
		基本工资	津贴	其他	水电费	实际收入
0101	张三	1 000.00	1 000.00	600.00	200.00	2 400.00
0102	李四	1 200.00	1 000.00	900.00	200.00	2 900.00
0103	王五	1 500.00	1 000.00	1 000.00	200.00	3 300.00

二、关系数据库

关系数据库主要是由若干个关系组成的相关的可被共享的数据集合。可以把关系数据库比喻为 Excel 应用软件创建的工作簿文件，一个关系相当于 Excel 工作簿文件中的一张工作表。当然这样比喻并不十分恰当，随着学习的不断深入，对关系数据库的认识也会越来越深入。

三、和关系数据库相关的术语

(1) 关系。这是一种特殊的二维表格，如前面所述，关系是同时具备上面 6 个特征的表格。

(2) 记录。除首行外（首行有时称为表的结构或表头），关系中的任意一行称为一条记录或一个元组 (Tuple)。

(3) 字段 (Field)。关系中任何一列，称为一个字段。列有名和值之分，如“姓名”是列名，“张三”是列值。

(4) 主键 (Primary Key)。这是一个列或多个列的组合，主键主要用来区分表中的每一行。

如学生表中的“学号”这一列就可以作为主键，而在表 1.3 所列的成绩表中，学号和课程号两个列组合起来才可以作为主键。在一个表中只能有一个主键。

表 1.3 成绩表

学 号	课 程 号	成 绩
2009010101	01	90
2009010101	02	85
2009010101	03	95
2009010102	01	95

续表

学 号	课 程 号	成 绩
2009010102	02	90
2009010103	02	80
2009010103	03	95

(5) **候选键**。这也是一个列或多个列的组合，候选键也可以用来区分表中的每一行。一个表中可以有多个候选键，如 QQ 号、身份证号、银行账号等都是候选键。一般是从多个候选键中选取一个作为主键。

(6) **主属性**，即包含在主键中的属性（列名）。主属性不同于主键，当主键由多个列组成时，一个数据表中可有多个主属性；而当主键由一个列组成时，主键和主属性可以理解为是等价的。

提示：初学者很容易误把一个关系中的多个主属性看成多个主键，这是不正确的，因为一个关系中只能有一个主键。

(7) **关系数据库**。与一个特定组织的各项应用有关的全部数据表（关系）的集合，主要由关系、视图和存储过程等数据库对象组成（注：视图和存储过程等内容将在后续的单元中陆续学习）。例如，学生表（参见表 1.1）、成绩表（参见表 1.3）和下面的课程表（参见表 1.4），就组成一个简单的教学管理系统数据库。

表 1.4 课程表

课 程 号	课 程 名	学 分
01	英语	5
02	数学	4
03	法律	2
04	体育	2

(8) **关系数据库管理系统**（Relation DataBase Management System，RDBMS）。这是对关系数据库进行创建和管理的一类系统软件的总称。数据库管理系统软件产品有很多种，常用的数据库管理系统软件有 SQL Server、Oracle、MySQL、Access、Visual FoxPro 等。在计算机应用基础课程中学习过的电子表格应用软件——Excel，就具备了关系数据库管理系统的一些基本功能。

本教材主要阐述如何利用 SQL Server 2005 这一关系数据库管理系统软件进行数据库的创建、管理、高效运行、安全管理及其他各种相关操作技能等知识。

任务 2 启动 SQL Server 2005 数据库管理系统

一、认识 SQL Server 2005

SQL Server 2005 是微软公司推出的基于客户机/服务器（Client/Server，C/S）体系结

构的关系型数据库管理系统，它具有较强的可伸缩性、可用性和可管理性。既可用于二层的 C/S 架构，又可用于多层的 B/S 架构。本教材主要讲解 SQL Server 2005 的二层 C/S 架构功能。

最简单的 C/S 架构将软件分成两部分，即一台服务器和多个客户机。服务器通常采用高性能的 PC、工作站或小型机，并安装大型数据库管理系统，如 Oracle 或 SQL Server。客户端常采用性能一般的 PC，需要安装专用的客户端组件，这种结构如图 1.1 所示。

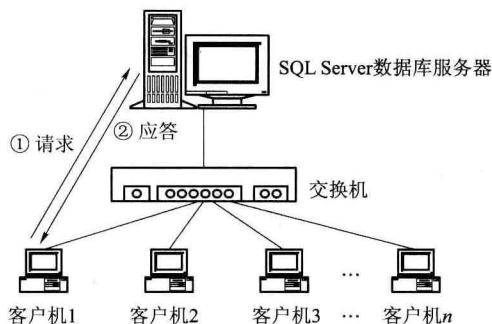


图 1.1 C/S 体系结构示意图

SQL Server 数据库服务器：指安装并启动运行了 SQL Server 2005 服务器端组件的计算机。

客户机：指安装并启动运行了 SQL Server 2005 客户端组件的计算机。

C/S 架构的工作机制有以下两点：

① 用户在客户机端使用 Transact-SQL 语句向 SQL Server 数据库服务器发出数据请求。

② SQL Server 数据库服务器响应用户的数据请求，并完成相应的数据操作后把结果信息发送给客户机。

二、启动运行 SQL Server 2005

1. 启动常用的 SQL Server 2005 数据库服务器端组件之一——数据库引擎

对数据库的创建、删除和管理等操作都是借助数据库管理系统软件来进行的，所以创建数据库之前必须先启动运行数据库管理系统软件。下面介绍如何启动 SQL Server 数据库管理系统软件。

常用的 Microsoft SQL Server 2005 数据库服务器端组件有数据库引擎、代理服务器、报表服务器、分析服务器等。数据库引擎（Microsoft SQL Server 2005 DataBase Engine）是最常用的服务器端组件之一，它是用于存储、处理和保护数据的核心后台服务程序。利用数据库引擎可控制访问权限，并快速处理事务，从而满足企业内要求极高而且需要处理大量数据的应用需要。数据库引擎还在保持高可用性方面提供了有力的支持。

启动运行 SQL Server 2005，必须先启动 Microsoft SQL Server 2005 数据库引擎，操作步骤如下：

单击“开始”→“所有程序”→Microsoft SQL Server 2005→“配置工具”菜单命令，选择 SQL Server Configuration Manager 选项，打开 SQL Server Configuration Manager 窗口，如图 1.2 所示。在此窗口的左侧栏中选择“SQL Server 2005 服务”选项，在右侧栏中右击 SQL Server (MSSQLSERVER)”选项，在弹出的快捷菜单中选择“启动”命令，如图 1.3 所示。启动完毕后，此时的计算机就是一台 SQL Server 数据库服务器了。

由于 SQL Server 数据库服务器运行于后台，SQL Server Configuration Manager (SQL Server 配置管理器) 窗口可以关闭，并不影响 SQL Server 数据库服务器的运行。

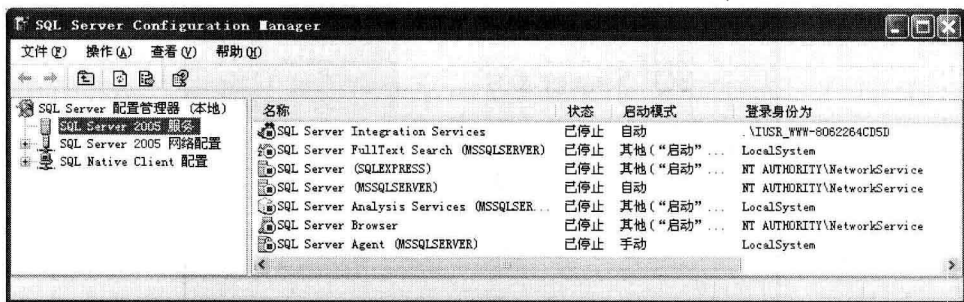


图 1.2 SQL Server 配置管理器窗口

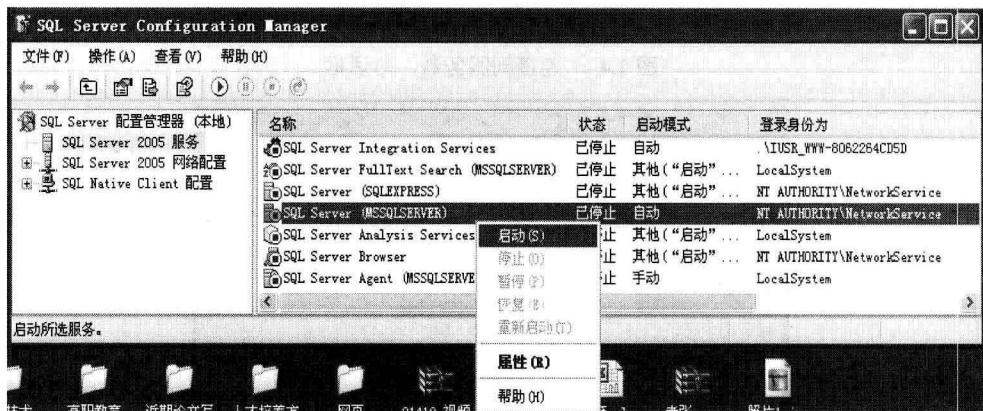


图 1.3 启动数据库服务器

提示：在 SQL Server 2005 安装完毕后，SQL Server 数据库引擎组件是自动启动的，不必手工启动。有时机房管理员为了提高计算机的运行速度，关闭了 SQL Server 数据库服务器端组件开机自动启动的功能，所以需要人工手动启动 SQL Server 数据库服务器端组件。另外，由于机房中的计算机一般都安装了还原卡，所以每次上机使用 SQL Server 2005 前，都要先手工启动相关的 SQL Server 数据库服务器端组件。

2. 启动 SQL Server 2005 客户端组件——SQL Server Management Studio

Microsoft SQL Server Management Studio 是 Microsoft SQL Server 2005 提供的一种新集成环境，用于访问、配置、控制、管理和开发 SQL Server 的所有其他组件。SQL Server Management Studio 将一组多样化的图形工具与多种功能齐全脚脚本编辑器组合在一起，可为各种技术级别的开发人员和管理员提供对 SQL Server 2005 数据库服务器的访问和管理。

启动 SQL Server 2005 客户端组件 SQL Server Management Studio 的操作步骤如下：

选择计算机的“开始”→“所有程序”→Microsoft SQL Server 2005 菜单命令，选择 SQL Server Management Studio 选项打开“连接到服务器”对话框，如图 1.4 所示。在此对话框的“服务器类型(T):”右侧的下拉列表框中选择“数据库引擎”选项，在“服务器名称(S):”右侧的下拉列表框中采用默认值，“在身份验证(A):”右侧的下拉列表框中选择“Windows 身份认证”选项，单击“连接”按钮，即可进入 Microsoft SQL Server Management Studio 窗口，如图 1.5 所示。



图 1.4 “连接到服务器”对话框

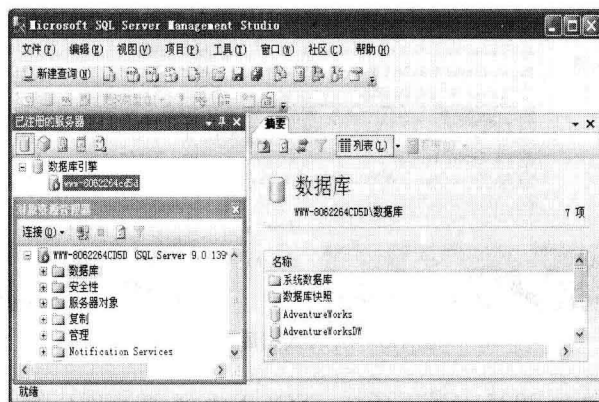


图 1.5 Microsoft SQL Server Management Studio 窗口

此时的计算机就是一台 SQL Server 客户机了。

Microsoft SQL Server Management Studio 窗口是今后操作 SQL Server 数据库的主要窗口，所有建立数据库、操作数据库和管理数据库等的操作都是在该窗口中进行的。

提示：必须先启动 SQL Server 2005 DataBase Engine 数据库引擎，然后启动 SQL Server 2005 客户端组件，顺序不能颠倒，因为数据库客户端组件启动时，要连接已经启动运行的数据库服务器端组件（如 SQL Server 2005 DataBase Engine 数据库引擎等）。

如果一台计算机既运行了 SQL Server 2005 服务器端组件，又运行了 SQL Server 2005 客户端组件，那么这台计算机就同时具备了数据库服务器端和数据库客户端双重功能。

提示：对安装 SQL Server 2005 软件感兴趣的同学可参考教材“附录 A SQL Server 2005 的安装”部分。

任务 3 创建关系数据库

一、数据库的组成

在 SQL Server 2005 中，数据库是用来存放表和索引的逻辑实体。数据库可以存放在一个

或多个物理磁盘上，一个磁盘也可以存放一个或多个数据库。一个最简单的数据库至少由一个扩展名为.MDF的主数据文件和一个扩展名为.LDF的事务日志文件组成。也可以根据实际需要，给数据库增加多个后缀为.NDF的次数据文件和其他扩展名为.LDF的事务日志文件，并可以将它们放在不同的磁盘上。

SQL Server 的每一个数据库都由关系图 (Relation Map)、表 (Table)、视图 (View)、存储过程 (Stored Procedures)、触发器 (Triggers)、用户自定义数据类型 (User-defined Data Types)、用户自定义函数 (User-defined Functions)、索引 (Indexes)、规则 (Constraints)、默认值 (Defaults) 等数据库对象组成。

二、系统数据库

在 SQL Server 2005 中，数据库分为系统数据库和用户数据库 2 种。

1. 系统数据库

一个新安装好的 SQL Server 2005 软件总是包括 4 个数据库，即 master、model、msdb 和 tempdb。

(1) master 数据库。

master 数据库由一些系统表组成。这些系统表跟踪作为整体的系统安装和随后创建的其他数据库。虽然每个数据库都有一组维护其包含对象信息的系统目录，但是 master 数据库中的一些系统目录还能够保存关于磁盘空间、文件分配和使用、系统层次的配置、端点、登录账号的信息，当前实例上的数据库信息和系统上其他 SQL Server 的存在信息（针对分布式操作）。

因为 master 数据库对系统来说至关重要，所以应保存它的一个当前备份。诸如创建另一个数据库、改变配置值和修改登录账号的操作都会修改 master 数据库，所以在执行过这些操作后应该备份 master 数据库。

(2) model 数据库。

model 数据库是一个模板数据库，它包含了用户数据库中应该包含的所有系统表及其他对象的结构。每当创建一个新的数据库时，SQL Server 都会复制 model 数据库作为新数据库的基础。如果希望每一个新的数据库在创建时都含有某些对象或者权限，可以把这些对象或权限放在 model 数据库中，此后，所有新建立的数据库都会继承它们。

(3) msdb 数据库。

msdb 数据库记录了有关 SQL Server Agent 代理服务的信息，如安排报警、作业，并记录操作员等信息。

SQL Server 代理服务会使用 msdb 数据库，它会执行一些如备份和复制等计划好的任务。Service Broker 也会用到 msdb 数据库，它为 SQL Server 提供队列和可靠消息传递。当不在该数据库上执行备份和维护任务时，通常可以忽略 msdb 数据库（不过也许需要查看备份历史和一些保存在那里的其他信息）。因为在 Microsoft SQL Server Management Studio 的对象资源管理器中可以访问 msdb 中的所有信息，所以通常不需要直接访问该数据库中的表。可以把 msdb 数据库中的表看做另一种形式的系统表，就像不能直接修改系统表，不应该直接在 msdb 的表中添加或删除数据，除非对自己的操作有着非常透彻的理解，或者确切知道怎样做。在 SQL Server 2005 以前，实际上是可以删除 msdb 数据库的，

之后 SQL Server 仍然可用，但不能再维护任何备份历史，并且不能再定义任务、警告、工作或者建立复制。在 SQL Server 2005 中，有一个没有文档记录的跟踪标记（Traceflag）允许删除 msdb 数据库，不过因为默认的 msdb 数据库非常小，建议即使用不到也不要删除它。

（4）tempdb 数据库。

tempdb 数据库是服务器的一个主要工作区。在执行复杂或者大型的查询操作时，如果 SQL Server 需要创建一些中间表来完成，那么它就在 tempdb 数据库中进行。在创建临时表时，即使你是在当前数据库中创建这些表的，但实际也是在 tempdb 数据库中创建的。只要需要临时保存数据，就是将数据保存在 tempdb 数据库中。

tempdb 数据库与其他任意数据库大相径庭。不仅数据库中的对象是临时的，连数据库本身也是临时的。每次启动 SQL Server 时，tempdb 数据库是系统中唯一完全重建的数据库。

注意：

从技术角度讲，可以在 tempdb 数据库中创建自己的对象，但强烈反对进行该操作。这是因为每次重启 SQL Server 数据库服务器时，SQL Server 系统都会删除并重新建立 tempdb 数据库，在其中创建的所有对象和权限在下次重启 SQL Server 时会全部丢失，不能长期保存。所以直接在 tempdb 数据库中创建对象没有任何意义。

2. 用户数据库

用户根据实际需求，为处理某一方面事务而自行建立的数据库，称为用户数据库。计算机用户主要创建并应用用户数据库。

三、创建用户数据库

使用 Microsoft SQL Server Management Studio 窗口创建数据库。

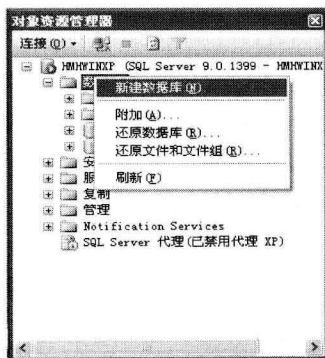


图 1.6 选择“新建数据库”快捷菜单命令

【例 1.1】 创建一个名为 XSGL 的学生信息管理数据库，主数据文件名为 XSGL_Data.MDF，初始文件大小为 4 MB，事务日志文件名为 XSGL_Log.LDF，初始文件大小为 4 MB，增长量为 1 MB。

（1）打开 Microsoft SQL Server Management Studio 窗口，在对象资源管理器树中展开服务器，右击“数据库”节点，在弹出的快捷菜单中选择“新建数据库”命令，如图 1.6 所示。

（2）打开“新建数据库”窗口，默认显示的是“常规”选择页。在“数据库名称”文本框中输入数据库的名字 XSGL，输入名称的同时，“数据库文件”列表框中的“逻辑名称”列将随之同步变化，如图 1.7 所示。

（3）“数据库文件”列表框里每一个单元格的内容都可以编辑，用鼠标单击“初始大小 (MB)”单元格，把数据文件和日志文件的初始大小分别改为 4 MB。

（4）在“数据库文件”列表框的“自动增长”单元格位置，可单击  按钮更改数据库自动增长方式，如图 1.8 所示。

同样在“路径”单元格位置单击按钮，可以改变存放数据库文件的路径，如图 1.9 所示。这里都不做修改，采用系统默认值。

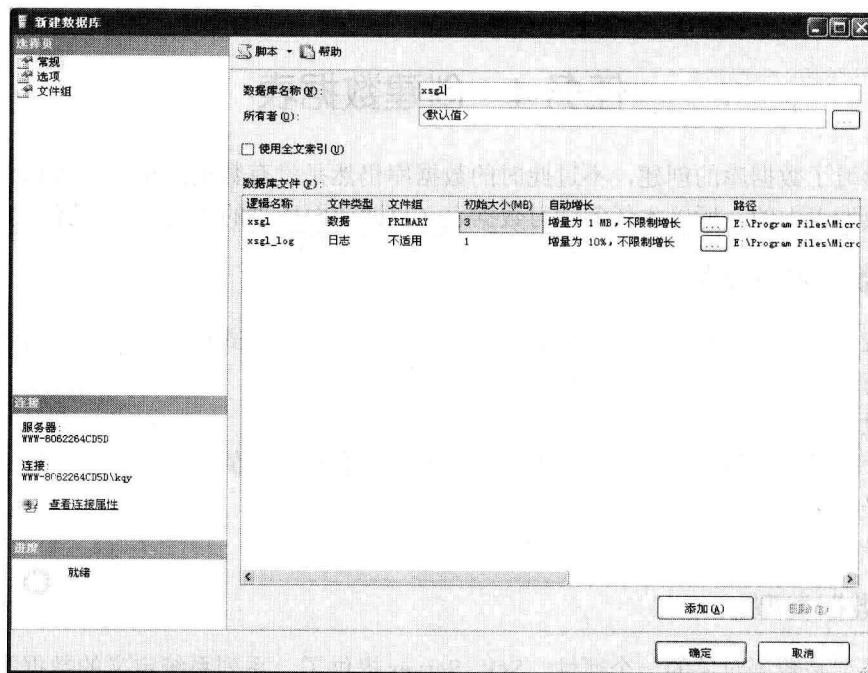


图 1.7 “新建数据库”窗口

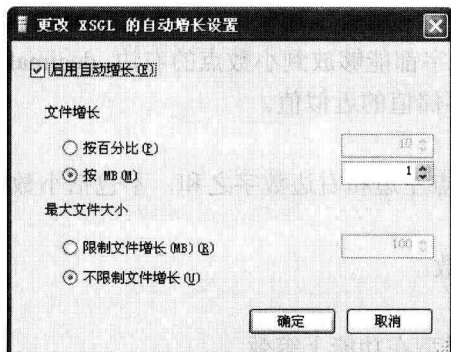


图 1.8 自动增长设置对话框

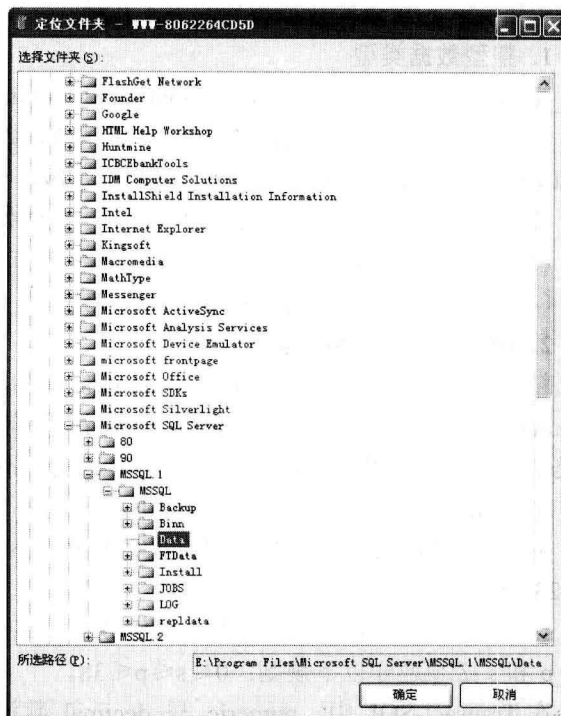


图 1.9 “定位文件夹”对话框



(5) 单击“确定”按钮, 创建进度显示在“新建数据库”窗口左下角的“进度”中。

创建成功后自动关闭“新建数据库”窗口, 并在“对象资源管理器”窗口的“数据库”节点下增加了名为 XSQL 的子节点。

任务 4 创建数据表

前面学习了数据库的创建, 不过此时的数据库仍然是没有数据的空库, 数据库中的数据是存放在数据表中的, 所以本任务学习数据表(也即前面所讲的“关系”)的创建。

一、数据表的概念

数据表定义为列的集合, 数据在表中是按行和列的组织形式排列的。每行代表唯一的一条记录, 而每列代表记录中的一个域。

具体设计表时一般应考虑以下几点:

- (1) 表的各列及每一列的数据类型, 列是否允许数据为空值。
- (2) 是否需要索引。
- (3) 是否使用约束、默认值或规则等。

二、数据类型

数据类型是数据对象的一个属性, SQL Server 提供了一系列系统定义的数据类型, 也可让用户根据需要在系统数据类型的基础上创建用户定义的数据类型。

SQL Server 提供的数据类型有以下几种:

1. 整型数据类型

整型数据类型有 bigint、int、smallint 和 tinyint 4 种类型。

(1) bigint 数据类型在数据库中占 8 个字节, 数值范围为 -2^{63} ($-9\ 223\ 372\ 036\ 854\ 775\ 808$) $\sim 2^{63}-1$ ($9\ 223\ 372\ 036\ 854\ 775\ 807$), 其精度为 19, 小数位数为 0。

(2) int 数据类型在数据库中占 4 个字节, 存储数据的范围是 $-2\ 147\ 483\ 648 \sim 2\ 147\ 483\ 647$ 。

(3) smallint 数据类型在数据库中占 2 个字节, 存储数据的范围为 $-32\ 768 \sim 32\ 767$ 。

(4) tinyint 数据类型在数据库中占 1 个字节, 存储数据的范围为 $0 \sim 255$ 。

2. 精确的数值型数据类型

精确的数值型数据有 decimal 和 numeric 两种。

decimal 数据类型最多可存储 38 位数字, 所有数字都能够放到小数点的右边。decimal 数据类型存储了一个准确(精确)的数字表达法; 不存储值的近似值。

定义 decimal 的列的两种特性如下:

p: 指定精度或对象能够控制的数字个数。小数点左边和右边数字之和, 不包括小数点。如 123.45, 则 $p=5$, $s=2$ 。

s: 指定可放到小数点右边的小数位数或数字个数。

p 和 s 必须遵守以下规则: $0 \leq s \leq p \leq 38$ 。

在 Transact-SQL 中, numeric 与 decimal 数据类型在功能上等效。

当数据值一定要按照指定精确存储时, 可以用带有小数的 decimal 数据类型来存储数字。