

Microsoft®

ADO.NET 3.5

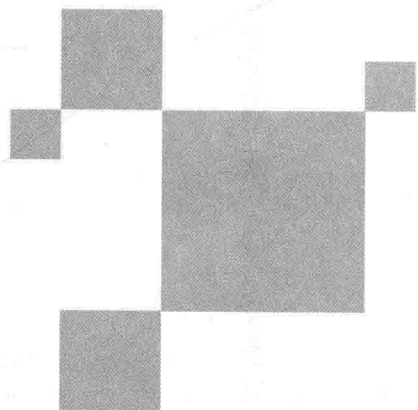
数据访问程序开发

微软公司 著



人民邮电出版社
POSTS & TELECOM PRESS





ADO.NET 3.5

数据访问程序开发

微软公司 著



人民邮电出版社

北京

图书在版编目 (C I P) 数据

ADO.NET 3.5数据访问程序开发 / 微软公司著. --
北京: 人民邮电出版社, 2010.10
ISBN 978-7-115-23346-2

I. ①A… II. ①微… III. ①软件工具—程序设计
IV. ①TP311.56

中国版本图书馆CIP数据核字(2010)第157537号

内 容 提 要

本书介绍 Microsoft 数据库开发技术的关键应用程序开发功能, 包括 Microsoft Visual Studio 2008 开发系统和 Microsoft ADO.NET 3.5, 以及如何开发可在连接场景和偶尔连接场景中插入、删除和更新数据库记录的应用程序。

全书共 8 章, 讲述了如何使用 ADO.NET 命令插入、删除和修改数据库记录; 在偶尔连接应用程序中使用数据集访问表; 使用语言集成查询 (LINQ) 插入、删除和修改数据库记录; 使用 ADO.NET 实体框架和实体 SQL 开发在概念模型中插入、删除和修改实体的代码; 使用同步服务在偶尔连接的应用程序中插入、删除和修改记录。

本书适合了解如何生成 Windows 窗体应用程序、如何使用基类库的常用功能, 以及对关系数据库管理系统数据库和事务结构化查询语言有基本理解的应用程序开发人员阅读。

ADO.NET 3.5 数据访问程序开发

-
- ◆ 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: 14 2010 年 10 月第 1 版
字数: 335 千字 2010 年 10 月北京第 1 次印刷

ISBN 978-7-115-23346-2

定价: 47.00 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 郭 健 李 进 张 充 宫 丽

前言

ADO.NET 是一组向 .NET Framework 程序员公开数据访问服务的类。ADO.NET 为创建分布式数据共享应用程序提供了一组丰富的组件，提供了对关系数据、XML 和应用程序数据的访问，因此是 .NET Framework 中不可缺少的一部分。ADO.NET 支持多种开发需求，包括创建由应用程序、工具、语言或 Internet 浏览器使用的前端数据库客户端和中间层业务对象。

通过学习本书，读者能够编写使用 ADO.NET 3.5 进行数据访问的应用程序。具体技能包括如下几点。

- 使用 ADO.NET 命令插入、删除和修改数据库记录。
- 在偶尔连接应用程序中，使用数据集访问表。
- 使用语言集成查询 (LINQ) 插入、删除和修改数据库记录。
- 使用 ADO.NET 实体框架和实体 SQL 开发在概念模型中插入、删除和修改实体的代码。
- 使用同步服务在偶尔连接的应用程序中插入、删除和修改记录。

本书适用对象

本书面向能够使用 VB.NET 和 C# 编写程序，且知道如何生成 Windows 窗体应用程序，以及如何使用基类库的常用功能的读者。本书假定读者对关系数据库管理系统数据库和事务结构化查询语言有基本的理解，但不必须具备高级 UI 线程和异步编程之类的概念。也不需读者具备生成 Visual Studio 加载项或自定义设计器的经验。

本书结构

本书共分为 8 章。

第 1 章 “数据库基础”。从最基本的数据库和表概念开始介绍，帮助没有数据库经验和概念的读者建立关于数据库方面的基础知识。

第 2 章 “ADO.NET 3.5 概述”。介绍数据访问模型和 ADO.NET 3.5 新特性，帮助读者对数据访问有一个基本的了解。

第 3 章 “ADO.NET 3.5 入门”。介绍 ADO.NET，描述了如何连接数据库以及如何使用 ADO.NET 命令来检索数据，并演示了如何使用简单命令和带参数的命令。

第 4 章 “使用 ADO.NET 命令修改数据”。描述了如何使用 ADO.NET 命令插入、删除和修改数据库记录，并演示了如何异步执行命令，以及在事务中执行命令。

第 5 章 “使用数据集查询和维护数据”。描述了如何使用数据集访问和处理数据。并演示了如何创建类型化数据集，以及如何在偶尔连接场景中使用数据集。

第 6 章 “使用 LINQ 查询和维护数据”。描述了 LINQ 的功能，以及如何使用 LINQ 查询和修改数据。

第 7 章 “使用 ADO.NET 实体框架实现实体数据模型”。描述了数据的概念模型和逻辑模型，以及如何建立两者间的映射。说明了如何使用实体 SQL，通过实体模型来访问数据。

第 8 章 “使用同步服务构建偶尔连接解决方案”。描述了如何使用同步服务来使偶尔连接应用程序的本地数据库缓存与数据库服务器同步。

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。操作界面参见图 0-1。

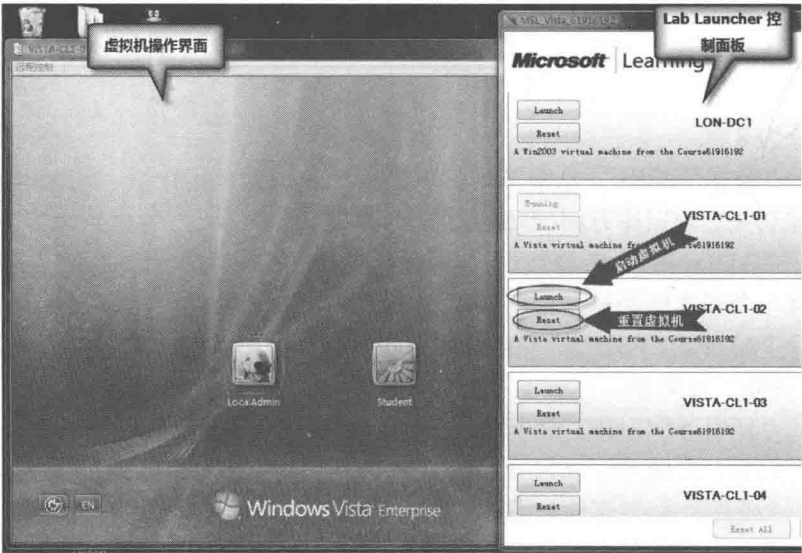


图 0-1 虚拟机操作界面

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。

在本书中包含一套虚拟机，分别对应不同的内容部分。

虚拟机	角色
10087A-LON-DEV-01	第 3 章实验所用的应用程序开发计算机
10087A-LON-DEV-02	第 4 章实验所用的应用程序开发计算机
10087A-LON-DEV-03	第 5 章实验所用的应用程序开发计算机
10087A-LON-DEV-04	第 6 章实验所用的应用程序开发计算机
10087A-LON-DEV-05	第 7 章实验所用的应用程序开发计算机
10087A-LON-DEV-06	第 8 章实验所用的应用程序开发计算机

目 录

第 1 章 数据库基础	1	3.1.4 将 SQL Server 数据类型 映射为 .NET Framework 类型	49
1.1 数据存储	1	3.1.5 检索数据时处理异常	51
1.2 数据库基础知识	1	3.1.6 断开数据库连接	52
1.2.1 表、行、列和约束	2	3.1.7 演示：连接数据库并 检索数据	53
1.2.2 在 Microsoft SQL Server 2008 中创建表	3	3.2 管理连接和执行查询的最佳 做法	54
1.2.3 创建表的步骤小结	6	3.2.1 在配置文件中存储连接 字符串	54
1.2.4 设置列属性和约束	6	3.2.2 管理数据库连接	55
1.2.5 SQL 语言	7	3.2.3 异步执行长时间运行的 查询	56
1.2.6 关系	10	3.2.4 高效执行单值查询	59
1.2.7 存储过程	16	3.2.5 创建并执行参数化 命令	60
1.3 数据库应用程序	25	3.2.6 用 ADO.NET 编写提供 程序无关的代码	62
1.4 习题	25	3.3 实验：连接数据库并检索 数据	64
第 2 章 ADO.NET 3.5 概述	27	3.3.1 实验 3-1：连接到 数据库	65
2.1 数据访问模型	27	3.3.2 实验 3-2：执行简单 查询	67
2.2 数据访问模型发展史	35	3.3.3 实验 3-3：执行返回 结果集的查询	67
2.3 ADO.NET 概述	36	3.3.4 实验 3-4：执行需要 参数的查询	68
2.3.1 ADO.NET 对象模型	37	3.4 习题	70
2.3.2 数据提供程序	38	第 4 章 使用 ADO.NET 命令修改数据	74
2.4 ADO.NET 的特征	40	4.1 插入、更新和删除数据	74
2.4.1 断开连接方式的数据 访问	40	4.1.1 创建并执行修改数据的	
2.4.2 增强的可编程性	40		
2.4.3 内置 XML 支持	41		
2.5 ADO.NET 3.5 技术和技巧	41		
2.6 习题	42		
第 3 章 ADO.NET 3.5 入门	43		
3.1 连接数据库并检索数据	43		
3.1.1 连接数据库	43		
3.1.2 处理连接异常	45		
3.1.3 执行查询并检索结果	47		

命令	74	5.2.3 将修改的数据保存到数据库	104
4.1.2 修改数据时处理异常	75	5.2.4 解决开放式并发冲突	105
4.1.3 异步执行长时间运行的修改	76	5.2.5 演示: 使用数据集更新数据库	107
4.1.4 演示: 插入、更新和删除数据	78	5.3 在偶尔连接的环境中使用数据集	107
4.2 管理数据完整性和并发性	78	5.3.1 在偶尔连接的环境中使用数据集作为缓存	107
4.2.1 事务概述	78	5.3.2 将数据集作为 XML 序列化和反序列化	108
4.2.2 事务和并发性	80	5.4 实验: 使用数据集检索和修改数据	109
4.2.3 SQL Server 2005 隔离级别	81	5.4.1 实验 5-1: 创建类型化数据集	109
4.2.4 实现事务的最佳做法	82	5.4.2 实验 5-2: 将数据检索到数据集中	111
4.2.5 演示: 管理数据完整性和并发性	83	5.4.3 实验 5-3: 修改数据集集中的数据	114
4.3 实验: 使用 ADO.NET 命令修改数据	83	5.4.4 实验 5-4: 保存数据集并解决冲突	115
4.3.1 实验 4-1: 在数据库中插入、更新和删除数据	84	5.5 习题	118
4.3.2 实验 4-2: 实现事务性更新	87		
4.3.3 实验 4-3: 异步执行命令	89		
4.4 习题	90		
第 5 章 使用数据集查询和维护数据	96	第 6 章 使用 LINQ 查询和维护数据	123
5.1 创建并使用数据集检索数据	96	6.1 使用 LINQ 查询表达式查询内存中的数据	123
5.1.1 数据集概述	96	6.1.1 LINQ 简介	123
5.1.2 非类型化和类型化数据集	97	6.1.2 定义 LINQ 查询	124
5.1.3 创建非类型化数据集	97	6.1.3 执行 LINQ 查询并访问结果	126
5.1.4 演示: 定义类型化数据集类	98	6.1.4 使用 LINQ 查询数据集集中的数据	127
5.1.5 将数据检索到数据集中	98	6.1.5 演示: 定义并执行 LINQ 查询	129
5.1.6 访问数据集集中的数据	100	6.2 使用 LINQ to SQL 检索数据	130
5.2 使用数据集更新数据库	102	6.2.1 定义实体类	130
5.2.1 修改数据集集中的数据	102	6.2.2 使用实体类检索数据	133
5.2.2 验证数据集中修改的数据	103	6.2.3 通过定义关联联接实体类	134

6.2.4	沿关联导航	137	7.2.3	在对象查询中使用 LINQ to Entities	169
6.2.5	演示: 通过编程方式 定义实体类	140	7.2.4	在对象查询中使用 实体 SQL	171
6.2.6	演示: 使用对象关系 设计器定义实体类	140	7.2.5	修改实体	176
6.2.7	微调查询	140	7.2.6	演示: 查询和修改 实体	179
6.3	使用 LINQ to SQL 修改数据	142	7.2.7	演示: 将实体绑定到 控件	179
6.3.1	插入、更新和删除实体 对象	142	7.3	实验: 使用 ADO.NET 实体 框架实现实体数据模型	179
6.3.2	将更改提交到数据库	145	7.3.1	实验 7-1: 创建实体 数据模型	179
6.3.3	处理开放式并发冲突	146	7.3.2	实验 7-2: 查询实体 数据模型中的实体	180
6.3.4	演示: 使用 LINQ to SQL 修改数据	149	7.3.3	实验 7-3: 修改实体 数据模型中的实体	182
6.4	实验: 使用 LINQ to SQL 查询和维护数据	149	7.4	习题	183
6.4.1	实验 6-1: 定义 实体类	150	第 8 章	使用同步服务构建偶尔 连接解决方案	188
6.4.2	实验 6-2: 使用 LINQ to SQL 检索数据	152	8.1	理解 Microsoft 同步服务	188
6.4.3	实验 6-3: 使用 LINQ to SQL 修改数据	154	8.1.1	同步服务概述	188
6.5	习题	157	8.1.2	同步服务支持的同步 模型	189
第 7 章	使用 ADO.NET 实体框架 实现实体数据模型	161	8.1.3	使用同步服务的常见 应用程序体系结构	189
7.1	使用 ADO.NET 实体框架创建 实体数据模型	161	8.1.4	同步服务与 SQL Server 合并复制	190
7.1.1	ADO.NET 实体框架 概述	161	8.2	使用同步服务下载数据	191
7.1.2	实体数据模型	162	8.2.1	配置服务器数据库以 支持同步	191
7.1.3	定义实体数据模型	165	8.2.2	同步服务 API 概述	193
7.1.4	演示: 创建实体数据 模型	166	8.2.3	在客户端数据库中 下载并缓存数据	194
7.1.5	演示: 修改实体数据 模型	166	8.2.4	演示: 使用同步服务 下载并缓存数据	197
7.2	使用 ADO.NET 实体框架查询和 修改数据	166	8.3	使用同步服务上载数据	197
7.2.1	对象服务概述	166	8.3.1	从客户端数据库上载	
7.2.2	定义对象查询	167			

修改过的数据	198	8.4.1	实验 8-1: 修改数据库	
8.3.2	检测并解决冲突	198	架构以支持同步	201
8.3.3	处理同步事件	199	8.4.2	实验 8-2: 将数据下载到
8.3.4	演示: 使用同步服务		客户端计算机	203
	上载数据	200	8.4.3	实验 8-3: 将数据更改
8.4	实验: 使用同步服务构建偶尔		上载到数据库	206
	连接解决方案	200	8.5	习题
				208

第1章 数据库基础

在现实生活中，人们无时无刻不在与数据打交道，小到通讯录中的一个电话号码，大到一个国家的国民生产总值。数据是关于事件的客观描述，例如电话号码就是通讯方式的客观描述，国民生产总值就是国家经济实力的客观描述。

单个零散的数据所表达的内涵往往很有限，而数据汇集在一起就可以看出事情的变化和发展。例如，历年员工平均工资额的数据放在一起，可以看出员工工资的走势。同时，在很多时候人们需要对这些数据进行处理。例如，计算班级所有学生成绩的平均分。面对如此纷繁复杂的数据，如果不建立一个规范的组织结构，数据的查找、修改和更新则会非常费时费力。数据库就是用来组织和存放数据的，至于数据的管理和处理，则由数据库管理系统完成。随着计算机技术与网络通信技术的发展，数据库技术已成为信息社会中对大量数据进行组织和管理的重要手段和软件技术，是网络信息化管理系统的基础，也是信息管理、办公自动化、计算机辅助设计等应用的主要软件工具之一。

本章将介绍如何在一个数据库中组织数据。为了让读者有更切身的体会，我们将创建一个简单的数据库。为什么第1章要着重讲解数据库的理论和创建？因为如果数据库未能在第一时间设计正确，则将会导致以数据为中心的应用程序出现问题。

本章将主要介绍数据库的基础知识和 SQL 语言，这是学习和掌握现代数据库技术的基础。

1.1 数据存储

在计算机尚未诞生的年代，人们用纸张记录数据，书写的时候不但极易出错，而且占用的资源和存放空间比较大，同时不便于查询。非专业记录人员往往要花费很长时间才能找到相关记录，大大降低了工作效率。某些非正常原因（如洪水、火灾等）往往容易导致数据大量丢失，使得其后的工作运行极为困难，甚至会引发重复性劳动。使用数据库之后，就可以高效且条理清晰地存储数据，它使人们能够更加迅速、方便地存储和管理数据。数据库的各级用户通过自己的权限进行相关查询，既不会破坏数据，又充分地利用了数据，将工作效率提高数倍。

例如，公司营销人员需要了解所在地区的人口数量、年龄层次及日常喜好，以有针对性地制定销售策略。这就需要市场调研部门的人员搜集相关信息，并将其存储到数据表中，再由营销人员针对特定需求对不同的数据表进行整合或区分，以查询相关的数据，去除冗余信息，从而做出正确的决策。

1.2 数据库基础知识

在介绍数据库之前，有必要先了解一下数据库的发展史。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三种：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。下面就来介绍关系模型中的数据是如何组织的。

1.2.1 表、行、列和约束

关系数据库是以一定方式将数据组织起来的集合。在关系数据库中，所有数据以表的形式进行存储。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，也可以保存员工的出生日期、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 0:0:0...	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

表中的每一行称为记录或实体，表示一个具体的对象，如图 1-1 黑框中的内容所示。每个表包含用户的若干条记录。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。

表中的每一列称为属性或字段，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 所示的表格中，所有列名均以“员工”二字开头。例如，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。

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。

	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 1-3 员工张宇的出生日期

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。

1.2.2 在 Microsoft SQL Server 2008 中创建表

数据库用来存储信息数据。当需要组织和管理存储在数据库中的数据时，通常会利用一些专用软件来完成，这类软件称为数据库管理系统。现在介绍如何在最常见的数据库管理系统 Microsoft SQL Server 2008 中用 SQL Server Management Studio 来创建一个如图 1-1 所示的员工表。创建表的步骤如下。

1. 启动 SQL Server Management Studio，出现“连接到服务器”界面，如图 1-4 所示。其中的服务器名称会因机器而异，单击“连接”按钮。
2. 在左窗体的“数据库”选项上单击鼠标右键，在随后弹出的快捷菜单中选择“新建数据库”命令，如图 1-5 所示。
3. 在“新建数据库”对话框中，在“数据库名称”文本框中输入“MySample”，如图 1-6 所示。



图 1-4 登录界面

如果要创建一个具有其他限制条件的数据库，可以在“选项”或“文件组”中进行具体操作。此处只是介绍基础操作，所以对其他内容不做深入的阐述。

4. 设置完毕后，单击“确定”按钮，可以看到已成功添加了 MySample 数据库，但是当你展开 MySample 时，可以看到数据库中没有数据。接下来，我们将带领你在 MySample 数据库中填充数据，并在此过程中温习数据库的基础知识。

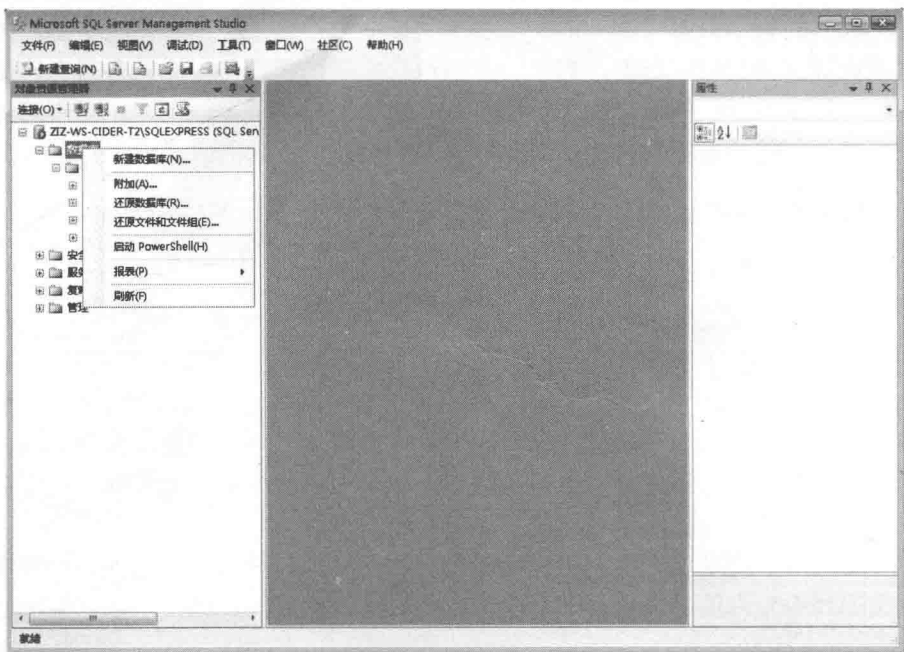


图 1-5 新建数据库

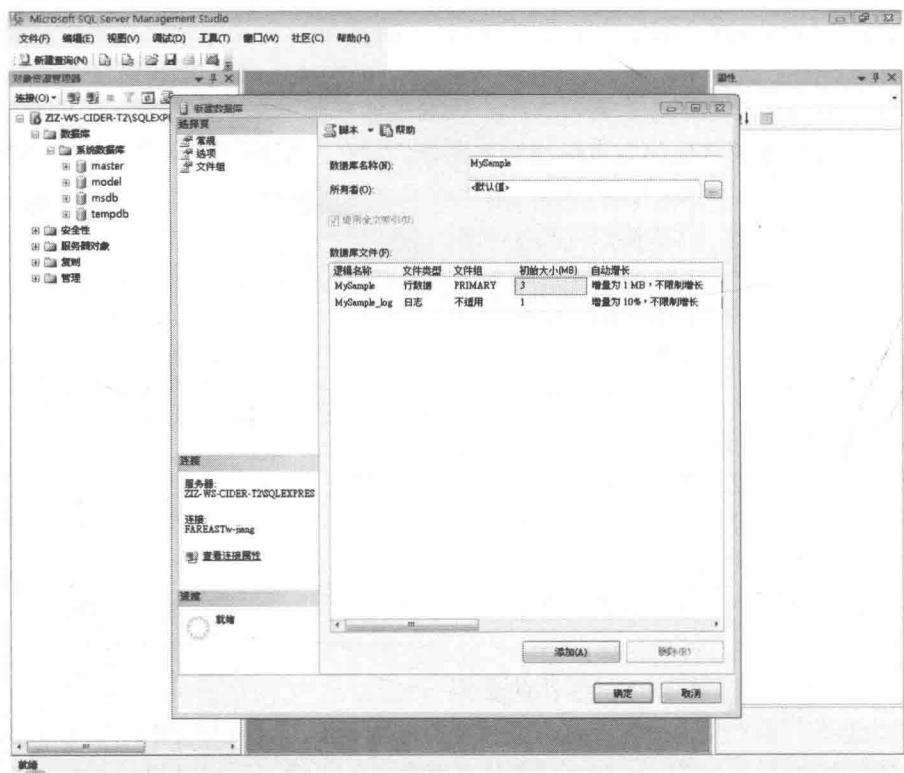


图 1-6 输入数据库名称

5. 在“对象资源管理器”窗口（如图 1-7 所示）中，展开 MySample 数据库，用鼠标右键单击表节点，选择“新建表”命令。

6. 首先新建“员工编号”列，并将其设置为主键，如图 1-8 所示。在列名中输入“员工编号”，并在“数据类型”下拉式列表中选择“int”（一旦将某列设置为主键，“允许空”那一列就默认设置为不能选择）。



图 1-7 新建表

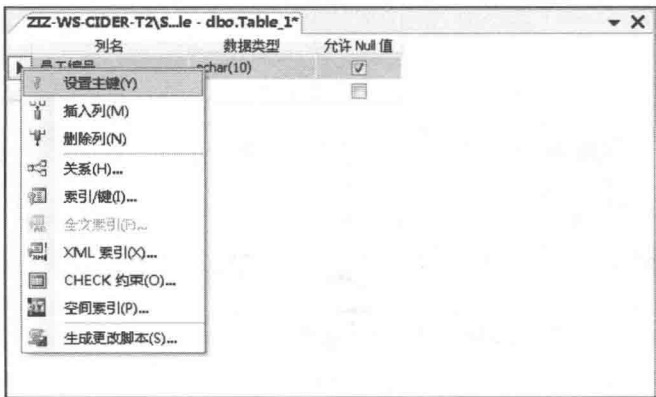


图 1-8 设置主键

7. 向员工表中添加其余几列，如图 1-9 所示。当向表中的某列填写详细信息时，一个新的空行会自动添加至列表的末端，允许为下一行添加详细信息。

8. 在窗体右侧的“属性”子窗口中，选择“员工编号”作为标识列，如图 1-10 所示。

9. 通过单击工具栏上的“保存”按钮来保存这张表。在“选择名称”对话框中输入“员工”，如图 1-11 所示，随后单击“确定”按钮。



图 1-9 员工表的列定义

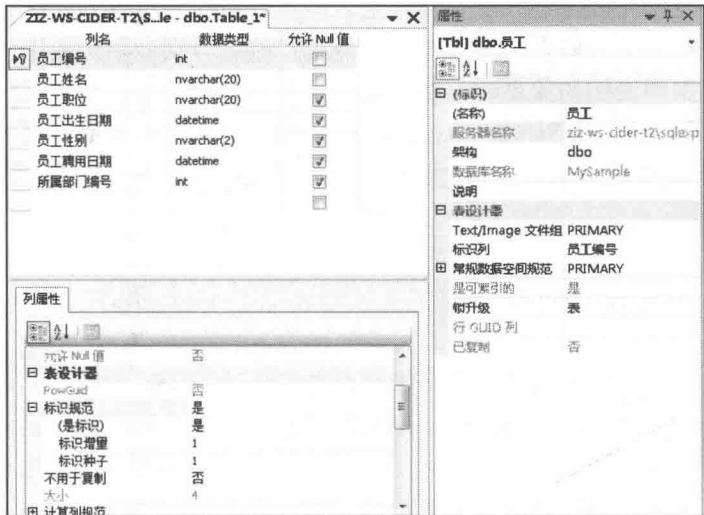


图 1-10 选择“员工编号”作为标识列

10. 为了确认数据表已经被创建，在“对象资源管理器”中展开表结点，将在列表树中看到新创建的员工表，如图 1-12 所示。

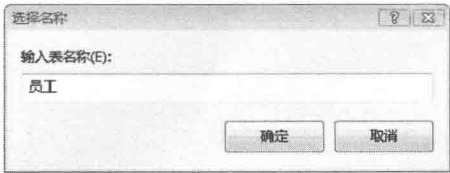


图 1-11 选择名称



图 1-12 新表创建完成

1.2.3 创建表的步骤小结

正如刚刚看到的，SQL Server Management Studio 所提供的工具使得在数据库中创建新表变得非常简单。在该示例中，实际完成了以下三项任务。

- 1. 创建新数据库；
- 2. 在数据库中创建新表；
- 3. 在表中创建列。

初次启动 SQL Server Management Studio 时，将首先连接到 SQL Server 2008 数据库服务器。一旦连接到数据库服务器，就可以很容易地创建新的数据库。由用户指定数据库名称，而由 SQL Server Management Studio 负责创建数据库并将其添加到可视的数据库列表中。

一旦创建数据库，就可以根据需要向其内添加新表。读者会发现创建列十分容易，只需指定列名及其数据类型并确定该列是否允许空值即可。通过使用列的快捷菜单，可以方便地在表中创建主键。正如前面所介绍的，主键使得用户能够分别识别出表中的每条记录。在本章后续小节探讨数据库表间的关系时，还将进一步介绍主键的重要性。最后一个问题是在表中设置标识列。实际上，SQL Server Management Studio 将用户创建数据库和表的意图看成用 SQL (Structured Query Language, 结构化查询语言) 所表达的查询，本章后面的“SQL 语言”部分将对此进行介绍。

1.2.4 设置列属性和约束

即便是创建简单表，也可以执行许多操作而非仅仅给表中的列命名。可以为各列赋以一组属性，然后进一步限制列的取值范围。这与在 C# 程序中为每个变量设置简单类型或复杂类型的方法类似。

以下是在员工表中的列属性和约束。

- 名称：列的名称。