



西安财经大学
XI'AN UNIVERSITY OF FINANCE AND ECONOMICS

信息学院



《数据库系统》实验指导书

信息学院

2019 年 1 月

目 录

第 1 章 前言	1
1.1 实验目标	1
1.2 实验内容	1
1.3 实验重点	1
第 2 章 SQL SERVER 2008	2
2.1 SQL Server 2008 简介	2
2.2 SQL Server 2008 安装过程	2
2.3 SQL Server 2008 配置	6
2.4 SQL Server 2008 管理工具	9
2.5 SQL Server 2008 系统数据库	11
2.6 数据库的组成	12
第 3 章 实验结果的备份与恢复	14
3.1 备份/还原数据库	14
3.2 分离/附加数据库	16
实验一 SQL 语句练习	21
4.1 实验目的	21
4.2 实验内容	21
4.3 实验任务	24
4.4 实验报告要求	25
实验二 SQL 数据库定义与操作	26
5.1 实验目的	26
5.2 实验内容	26
5.3 实验任务	29
5.4 实验报告要求	30
实验三 SQL 数据查询语句	31
6.1 实验目的	31
6.2 实验内容	31
6.3 实验任务	31
6.4 实验报告要求	33
实验四 SQL 数据更新语句	34
7.1 实验目的	34
7.2 实验内容	34
7.3 实验任务	34
7.4 实验报告要求	35
实验五 视图的定义与操作	36
8.1 实验目的	36
8.2 实验内容	36
8.3 实验任务	36
8.4 实验报告要求	36
实验六 数据库完整性的管理	37
9.1 实验目的	37
9.2 实验内容	37

9.3 实验任务.....	42
9.4 实验报告要求.....	42
实验七 数据库安全性的管理.....	43
10.1 实验目的.....	43
10.2 实验内容.....	43
10.3 实验任务.....	45
10.4 实验报告要求.....	45
实验八 存储过程实验.....	46
11.1 实验目的.....	46
11.2 实验内容.....	46
11.3 实验任务.....	46
11.4 实验报告要求.....	46
实验九 触发器实验*.....	47
12.1 实验目的.....	47
12.2 实验内容.....	47
12.3 实验任务.....	49
12.4 实验报告要求.....	49
实验十 数据库设计综合应用.....	50
13.1 实验目的.....	50
13.2 实验内容.....	50
13.3 实验任务.....	50
13.4 实验报告要求.....	50
13.5 参考题目.....	51

第 1 章 前言

根据教学计划安排,《数据库系统》课程总学时数为 72 学时,其中实验 36 学时。在实验中,包含有验证性实验、设计性实验,综合设计性实验。

本实验指导书,对于《数据库系统》的实验环境和实验所需的预备知识进行了详细的介绍,同时对每个实验都根据实验目的精心设计了实验内容,给出了实验步骤。实验中采用 Microsoft 公司的 SQL Server 2008 作为数据库平台。

1.1 实验目标

SQL 语言是关系数据库的标准语言,是本课程的一个重点。通过上机实验,可以使生加深对课堂讲授内容的理解,循序渐进地掌握 SQL 语言的使用;同时,使学生了解和熟悉 SQL SERVER 开发的环境,逐步掌握编辑、调试、运行程序的方法,初步积累编程经验。

1.2 实验内容

数据库、基本表、视图、索引的建立与数据的更新;关系数据库的查询,包括单表查询、连接查询、嵌套查询等;数据库系统的实现技术,包括事务的概念及并发控制、恢复、完整性和安全性实现机制。

1.3 实验重点

本课程的实验教学重点包括:

- (1) 数据库、基本表、视图、索引的建立与数据的更新;
- (2) SQL 的数据查询;
- (3) 恢复、完整性和安全性实现机制;
- (4) 设计一个小型的数据库管理系统。

第 2 章 SQL Server 2008

课程实验以 SQL Server 2008 作为数据库管理系统平台。本章将对 Microsoft SQL Server 2008 系统进行介绍, 使学生对该系统有整体的认识 and 了解, 为后面实验奠定坚实的基础。

2.1 SQL Server 2008 简介

SQL Server 是一个工作组级的关系数据库管理系统。它最初是由 Microsoft、Sybase 和 Ashton Tate 三家公司共同开发的, 于 1988 年推出了第一个 OS/2 版本。在 Windows NT 推出后, Microsoft 与 Sybase 在 SQL Server 的开发上就分道扬镳。Microsoft 将 SQL Server 移植到 Windows NT 系统上, 专注于开发推广 SQL Server 的 Windows NT 版本。Sybase 则较专注于 SQL Server 在 UNIX 操作系统上的应用。

1995 年, 微软公司成功地发布了 Microsoft SQL Server 6.0 系统, 这是微软公司完全独立开发和发布的第一个 SQL Server 版本。1996 年, 微软公司又发布了 Microsoft SQL Server 6.5 系统。这是微软公司独自发布的功能齐全、性能稳定的 SQL Server 系统, 该系统在数据库市场上占据了一席之地, 在我国的应用范围也开始逐渐扩大。

2008 年 8 月, 微软公司发布了 Microsoft SQL Server 2008 系统, 其代码名称是 Katmai。该系统在安全性、可用性、易管理性、可扩展性、商业智能等方面有了更多的改进和提高, 对企业的数据存储和应用需求提供了更强大的支持和便利。

2.2 SQL Server 2008 安装过程

根据安装机器软硬件的要求, 选择一个合适的版本, 以下以 SQL Server 2008 R2 为例。

1. 打开 SQL Server 2008 安装程序, 双击根文件夹中的 setup.exe。在自动运行的对话框中, 单击“运行 SQL Server 安装向导”。

2. 在“最终用户许可协议”页上, 阅读许可协议, 再选中相应的复选框以接受许可条款和条件。接受许可协议后即可激活“下一步”按钮。若要继续, 请单击“下一步”若要结束安装程序, 请单击“取消”。如图 2-1 所示:



图 2-1 “许可条款”对话框

3. 在“SQL Server 组件更新”页上, 安装程序将安装 SQL Server 2008 的必需软件。有关组件要求的详细信息, 请单击该页底部的“帮助”按钮。若要开始执行组件更新, 请单击“安装”, 如图 2-2 所示。更新完成之后若要继续, 请单击“下一步”。

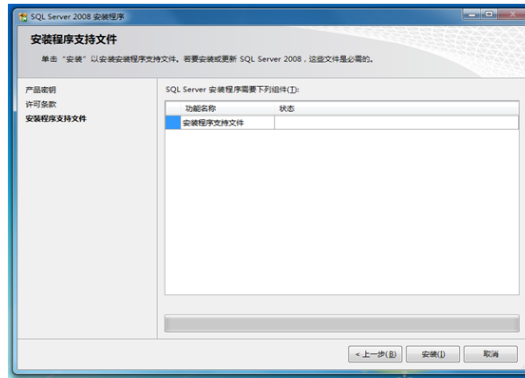


图 2-2 安装程序支持文件

4. 在“系统配置检查 (SCC)”页上，将扫描安装计算机，以检查是否存在可能妨碍安装程序的条件，如图 2-3 所示。



图 2-3: 安装程序支持规则

5. 在“要安装的组件”页上，请选择要安装的组件。选择各个组件组时，“要安装的组件”窗格中会显示相应的说明。可以选中任意一些复选框，建议全选。然后单击“下一步”继续，如图 2-4 所示。

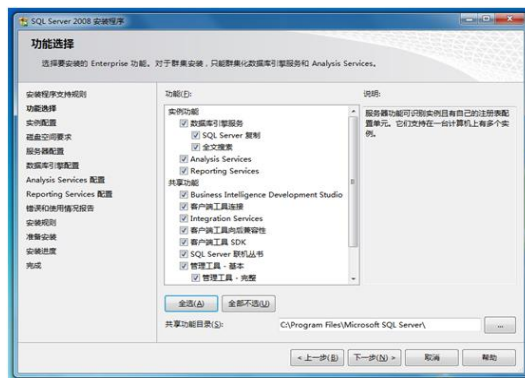


图 2-4 功能选择

6. 在“实例名”页上，请为安装的软件选择默认实例或已命名的实例。计算机上必须没有默认实例，才可以安装新的默认实例。若要安装新的命名实例，请单击“命名实例”，然后在提供的空白处键入一个唯一的实例名，如图 2-5 所示。

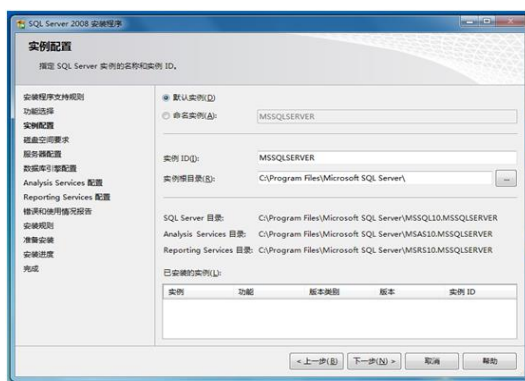


图 2-5 案例配置

7. 在弹出的“服务器配置”页面，选中“服务帐户”页，为 SQL Server 服务帐户指定用户名、密码和域名，可以对所有服务使用一个帐户，如图 2-6 所示。

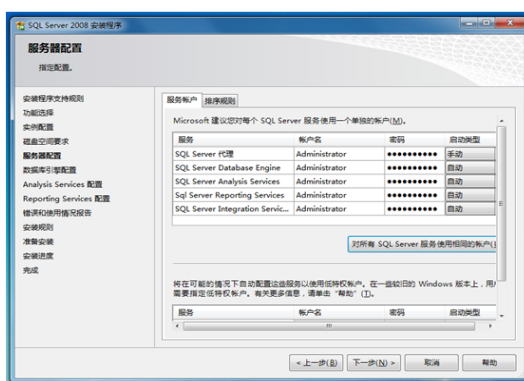


图 2-6 服务器配置

8. 在“身份验证模式”页上，选择要用于 SQL Server 安装的身份验证模式。如果选择 Windows 身份验证，安装程序会创建一个 sa 帐户，该帐户在默认情况下是被禁用的。选择“混合模式身份验证”时，请输入并确认系统管理员 sa 的登录密码。建议选择混合模式，并输入安全的密码，如图 2-7 所示。

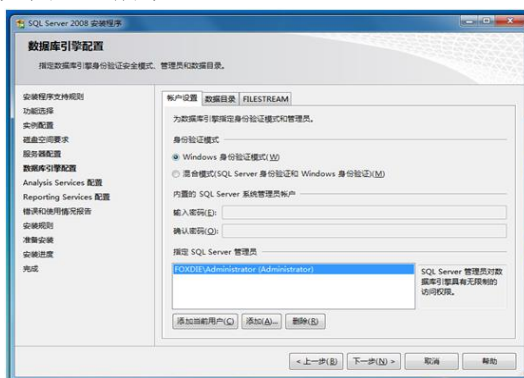


图 2-7 服务器引擎配置

9. 如果选择 Reporting Services 作为要安装的功能，将显示“报表服务器安装选项”页。使用单选按钮选择是否使用默认值配置报表服务器。如果没有满足在默认配置中安装 Reporting Services 的要求，则必须选择“安装但不配置服务器”安装选项。若要继续安装，请单击“下一步”，如图 2-8 所示：

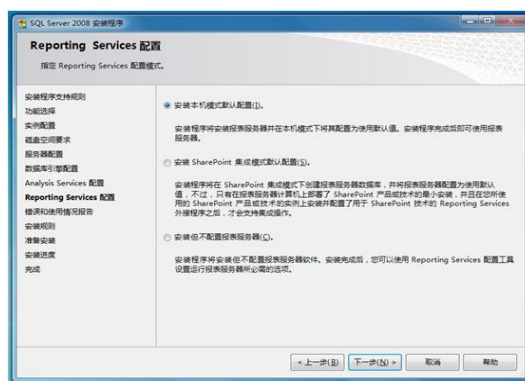


图 2- 8 Reporting Services 配置

10. 在“错误报告”页上，可以清除复选框以禁用错误报告。有关错误报告功能的详细信息，请单击该页底部的“帮助”。若要继续安装，请单击“下一步”，如图 2- 9 所示。

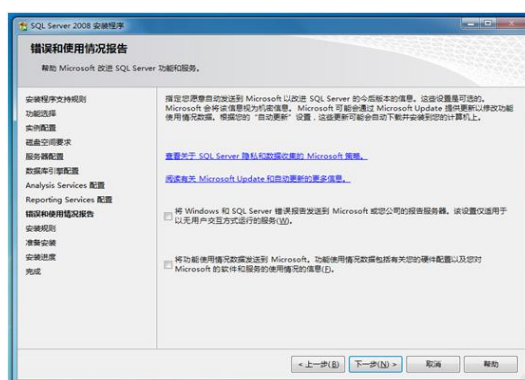


图 2- 9 错误和使用情况报告

11. 在“准备安装”页上，查看要安装的 SQL Server 功能和组件的摘要。若要继续安装，请单击“安装”，如图 2- 10 所示。

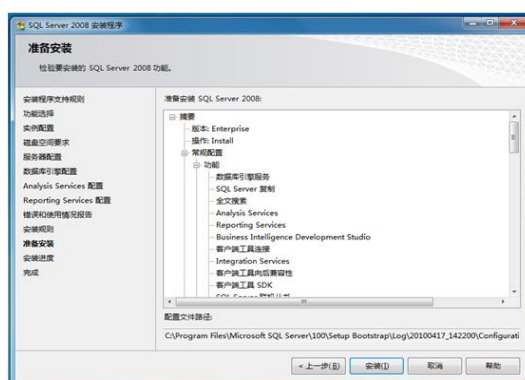


图 2- 10 准备安装

12. 在“安装进度”页上，可以在安装过程中监视安装进度。若要在安装期间查看某个组件的日志文件，请单击“安装进度”页上的产品或状态名称，如图 2- 11 所示。

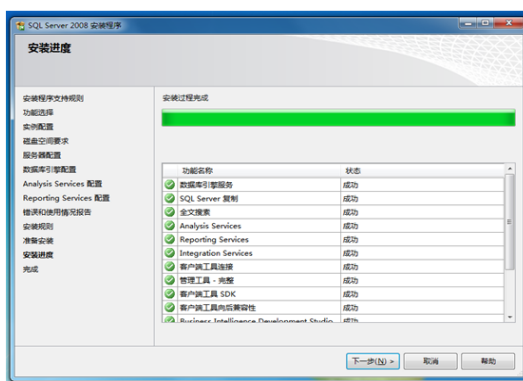


图 2- 11 安装进度

13. 在“完成 Microsoft SQL Server 安装向导”页上，可以通过单击此页上提供的链接查看安装摘要日志。若要退出 SQL Server 安装向导，请单击“关闭”，如图 2- 12 所示。

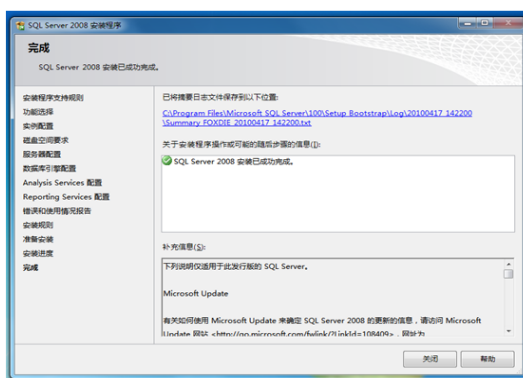


图 2- 12 安装完成

14. 如果提示您重新启动计算机，请立即重新启动。

15. 如果成功安装了 SQL Server 2008，则在开始菜单中添加了如下程序和相应的服务，如图 2- 13 所示：



图 2- 13 开始菜单中 SQL Server 2008 程序

2.3 SQL Server 2008 配置

2.3.1 SQL Server 2008 服务启动和停止

在 Microsoft SQL Server 2008 系统中，可以通过“计算机管理”工具或“SQL Server 配置管理器”查看和控制 SQL Server 的服务。

在桌面上，选择“我的电脑” - “管理”命令，可以看到如下图所示的“计算机管理”窗口。在该窗口中，可以通过“SQL Server 配置管理器”节点中的“SQL Server 服务”子节点查看到 Microsoft SQL Server 2008 系统的所有服务及其运行状态。图 2- 14 是 SQL Server 2008 系统的 7 个服务。

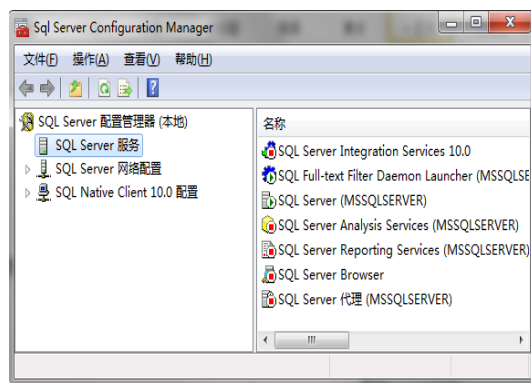


图 2- 14 SQL Server 2008 系统的 7 个服务

右击某个服务名称，可以查看该服务的属性，并且可以启动、停止、暂停和重新启动相应的服务。也可以使用操作系统“我的电脑”—“管理”选项，在【计算机管理】窗口中查看和启动、停止、暂停和重新启动相应的服务。

2.3.2 注册服务器

在安装 SSMS 之后首次启动它时，将自动注册 SQL Server 的本地实例。服务器只有在注册后才能被纳入 SSMS 的管理范围。如果需要在其他客户机上完成管理，就需要手工进行注册。

可以通过使用 SSMS 注册服务器，在注册服务器时必须指定以下参数：

- 服务器的名称
- 登录到服务器时使用的安全类型
- 如果需要，指定登录名和密码
- 注册了服务器后想将该服务器列入其中的组的名称

(1) 启动 SSMS，选择“视图/已注册的服务器”，显示“已注册的服务器”窗口，如图 2- 15 所示。

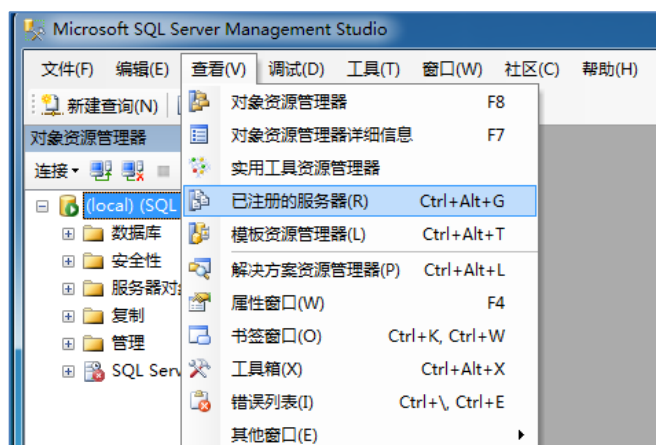


图 2- 15 已注册的服务器窗口

(2) 在“已注册的服务器”窗口中，显示了当前系统中的服务器组和所有已在 SSMS 注册的服务器，选择某一服务器组，用鼠标右键单击，在出现的快捷菜单中选择“新建

服务器注册”选项。

(3) 在“新建服务器注册”窗口中，设置要注册的服务器名称和登录到服务器时使用的安全类型，单击“测试”按钮，如果设置正确，则显示测试成功；否则，重新进行参数设置，如图 2-16 所示。

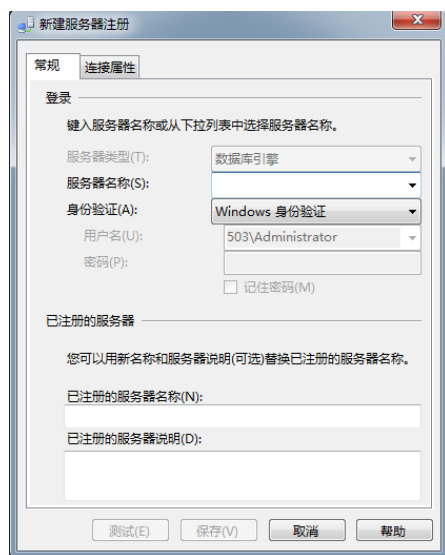


图 2-16 新建服务器注册窗口

(4) 单击“保存”按钮，则服务器注册成功。

成功注册的服务器就可以在 SSMS 中进行管理了。

2.3.3 创建服务器组

在一个网络系统中，可能存在多个 SQL Server 服务器分别保存不同的数据，可以对这些 SQL Server 服务器进行分组管理。分组的原则通常是依据组织结构原则，如将公司内一个部门的几台 SQL Server 服务器分为一组，这就是服务器组。服务器组便于对不同类型和用途的 SQL Server 服务器进行管理。

- 创建服务器组由 SSMS 来进行。

- (1) 启动 SSMS，选择“视图/已注册的服务器”，显示“已注册的服务器”窗口。
- (2) 在“已注册的服务器”窗口中，显示了当前系统中的服务器组和所有已在创建服务器，选择某一服务器组，用鼠标右键单击，在出现的快捷菜单中选择“新建服务器组”选项。
- (3) 在“新建服务器组”窗口中，设置要创建的服务器组名称和相关说明，单击“确定”按钮，则服务器组创建成功，如图 2-17 所示。

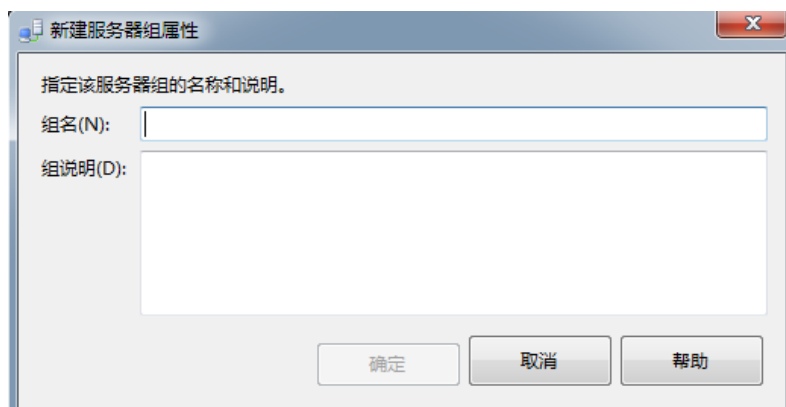


图 2-17 新建服务器组窗口

2.4 SQL Server 2008 管理工具

2.4.1 SQL Server Management Studio

Microsoft SQL Server Management Studio(SSMS)是 Microsoft SQL Server 2008 提供了一种集成环境,将各种图形化工具和多功能的脚本编辑器组合在一起,完成访问、配置、控制、管理和开发 SQL Server 的所有工作,大大方便了技术人员和数据库管理员对 SQL Server 系统的各种访问。Microsoft SQL Server Management Studio 启动后主窗口如图 1-6 所示。

Microsoft SQL Server 7 和 2000 版本中的 SQL Server Enterprise Manager 和 Query Analyzer 都被 Microsoft SQL Server 2005/2008 中的 Microsoft SQL Server Management Studio 工具替代了。使用 Microsoft SQL Server Management Studio 还可以管理 Microsoft SQL Server 7/2000/2005 实例。

Microsoft SQL Server Management Studio 由多个管理和开发工具组成的,主要包括“已注册的服务器”窗口、“对象资源管理器”窗口、“查询编辑器”窗口、“模板资源管理器”窗口、“解决方案资源管理器”窗口等。

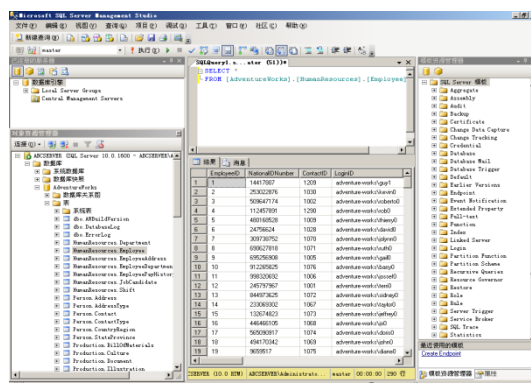


图 2-18 Microsoft SQL Server Management Studio 主窗口

“已注册的服务器”窗口位于图 2-18 中的左上角,可以完成注册服务器和将服务器组合成逻辑组的功能。通过该窗口可以选择数据库引擎服务器、分析服务器、报表服务器、集成服务器等。当选中某个服务器时,可以从右键的快捷菜单中选择执行查看服务器属性、启动和停止服务器、新建服务器组、导入导出服务器信息等操作。

“对象资源管理器”窗口位于图 2- 18 中的左下角，可以完成类似 SQL Server Enterprise Manager 工具的操作。具体地说，“对象资源管理器”窗口可以完成如下一些操作。

- 注册服务器
- 启动和停止服务器
- 配置服务器属性
- 创建数据库以及创建表、视图、存储过程等数据库对象
- 生成 Transact-SQL 对象创建脚本
- 创建登录账户
- 管理数据库对象权限
- 配置和管理复制
- 监视服务器活动、查看系统日志等

“查询编辑器”是以前版本中的 Query Analyzer 工具的替代物，它位于图 2- 18 的中部。用于编写和运行 Transact-SQL 脚本。与 Query Analyzer 工具总是工作在连接模式下，不同的是，“查询编辑器”既可以工作在连接模式下，也可以工作在断开模式下。另外，如同 Visual Studio 工具一样，“查询编辑器”支持彩色代码关键字、可视化地显示语法错误、允许开发人员运行和诊断代码等功能。因此，“查询编辑器”的集成性和灵活性大大提高了。

“模板资源管理器”窗口位于如图 2- 19 的右上部，该工具提供了执行常用操作的模板。用户可以在此模板的基础上编写符合自己要求的脚本。例如在“模板资源管理器”窗口中打开 Database 节点，如图 2- 19 所示，可以生成诸如 attach database、Bring Database Online、Create Database on Multiple File groups 等操作的模板。

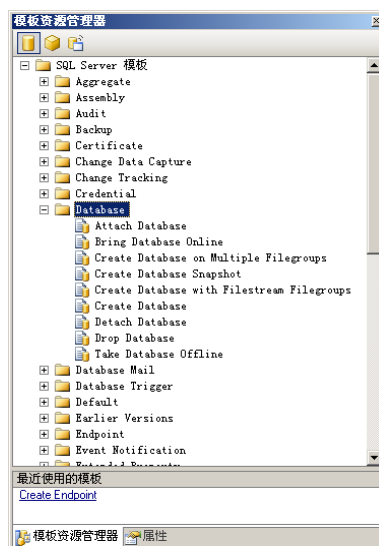


图 2- 19 模板资源管理器

2.4.2 SQL Server Profiler

使用摄像机可以记录一个场景的所有过程，以后可以反复地观看。能否对 Microsoft SQL Server 2008 系统的运行过程进行摄录呢？答案是肯定的。使用 SQL Server Profiler 工具可以

完成这种摄录操作。从 Microsoft SQL Server Management Studio 窗口的“工具”菜单中即可运行 SQL Server Profiler。SQL Server Profiler 的运行窗口如图 2- 20 所示。

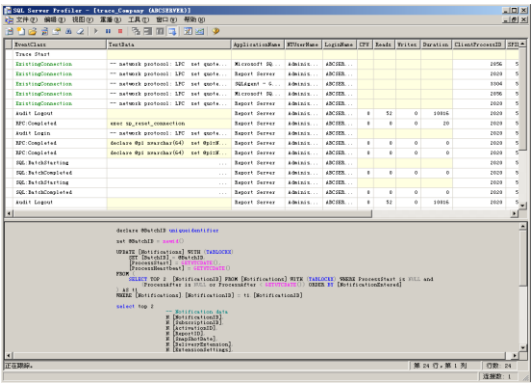


图 2- 20 SQL Server Profiler 的运行窗口

SQL Server Profiler 是用于从服务器中捕获 SQL Server 2008 事件的工具。这些事件可以是连接服务器、登录系统、执行 Transact-SQL 语句等操作。这些事件被保存在一个跟踪文件中，以便日后对该文件进行分析或用来重播指定的系列步骤，从而有效地发现系统中性能比较差的查询语句等相关问题。

2.5 SQL Server 2008 系统数据库

从数据库的应用和管理角度上看，SQL Server 将数据库分为两大类：系统数据库和用户数据库。系统数据库是 SQL Server 数据库管理系统自动创建和维护的，这些数据库用于维护系统正常运行的信息。用户数据库保存的是与用户的业务有关的数据。我们通常所说的创建数据库指的都是创建用户数据库，对数据库的维护管理也是指的是对用户数据库的维护。一般用户对系统数据库只有查询权。系统数据库中保存的系统表用于系统的总体控制。系统数据库保存了系统运行及对用户数据的操作等基本信息。在安装好 SQL Server 2008 后，系统会自动安装四个用于维护系统正常运行的系统数据库，分别是 master、msdb、model 和 tempdb。这些系统数据库的文件存储在 SQL Server 默认安装目录（MSSQL）中的子目录 Data 文件夹中。

1. master 数据库

master 数据库是 SQL Server 系统最重要的数据库，它记录了 SQL Server 系统的所有系统信息。若 master 数据库被损坏，SQL Server 服务器将无法正常工作。这些系统信息包括所有的系统配置信息、登录信息、SQL Server 初始化信息和其他系统数据库和用户数据库的相关信息。因此，当创建一个数据库，更改系统的设置、添加个人登录账户等更改系统数据库 master 的操作之后，应当及时备份 master 数据库，这一点非常重要。确保备份 master 数据库是备份策略的一部分。

2. tempdb 数据库

tempdb 数据库是一个临时数据库，它为所有的临时表和其他临时存储需求提供存储空间，是一个由 SQL Server 中所有数据库共享使用的工作空间。也就是说，不管用户使用哪

个数据库，所建立的所有临时表和存储过程都存储在 tempdb 中。当用户与 SQL Server 断开连接或系统关机时，临时数据库中创建的临时表和存储过程被自动删除。在 tempdb 数据库中所做的操作不会被记录，因而在 tempdb 数据库中的表上进行数据操作要比在其他数据库中要快得多。而每次 SQL Server 启动时，tempdb 数据库都将被重建恢复到系统设定的初始状态，因此千万不要将 tempdb 数据库作为数据的最终存放处。同时，临时数据库是系统中负担较重的数据库，可以通过将其置于 RAM 中以提高数据库的性能。

3. model 数据库

model 数据库是创建所有用户数据库和 tempdb 数据库的模板文件。model 数据库中包含每个数据库所需的系统表格，是 SQL Server 2008 中的模板数据库。当创建一个用户数据库时，模板数据库中的内容会自动复制到所创建的用户数据库中，因此，用户创建的数据库不能小于 model 数据库的大小。同时也可以利用 model 数据库的模板特性，通过更改 model 数据库的设置，将经常使用的数据库对象复制到 model 数据库中，可以简化数据库及其对象的创建、设置工作，为用户节省大量的时间。

4. msdb 数据库

msdb 数据库存放服务器的任务列表，可以把定期调试执行的任务加到这个数据库中，它为报警、任务调试和记录操作员的操作提供存储空间。

2.6 数据库的组成

在 SQL Server 中数据库是由数据文件和事务日志文件组成的。一个数据库至少应包含一个数据文件和一个事务日志文件。

数据库文件是存放数据库数据和数据库对象的文件。SQL Server 数据库文件根据功能的不同，可以分为以下三种类型。

1. 主要数据文件（primary file）：用来存储数据库的数据和数据库的启动信息。每个数据库都必须有而且只能有一个主要数据文件，其扩展名为.mdf。主要数据文件中包含了其他数据库文件的信息。SQL Server 2008 要求主数据文件的大小不能小于3MB。

2. 次要数据文件（secondary file）：用来存储主要数据文件没有存储的其他数据，一个数据库可以没有次要数据文件，也可以有多个次要数据文件，而且这些次要数据文件可以建立在一个磁盘上，也可以分别建立在不同的磁盘上。次要数据文件的扩展名为.ndf。使用次要数据文件可以扩展存储空间。当数据库很大，可以建立一个主数据文件和多个次要数据文件，一个数据库最多可以有32 766 个次要数据文件。把数据库目录存储在主数据文件中，把数据和对象存储在次要数据文件中，这样的配置有助于减小磁盘访问竞争。如果数据库用一个主要数据文件和多个次要数据文件来存储数据，并将它们放在不同的物理磁盘中，数据库的总容量就是这几个磁盘容量之和。

3. 事务日志文件（transaction log）：事务日志文件是用来记录对数据库的操作信息的，它把对数据库的所有操作事件均记载下来。这就好比有人有写日记的习惯一样，他会把每天

做的所有事情都记录下来，日记就相当于这里的日志文件，而人每天所做的事件就相当于数据库的操作，当某个时刻，人忘记了之前所做的事情时就去查看他所写的日记，数据库也是如此。在数据库有问题的时候可以去看日志文件来分析出错的原因，当数据库被破坏时也可以利用事务日志文件恢复数据库的数据。日志文件的扩展名为.ldf。

需要注意的是，在SQL Server 数据库管理系统中，每个数据库必须至少有一个日志文件，但可以不止一个，事务日志文件也不属于任何文件组。在SQL Server 2008 中，某个数据库中的所有文件的位置都记录在master 数据库和该数据库的主文件中。

第3章 实验结果的备份与恢复

在每次实验完成后，为了能够将本次的实验结果加以保存防止意外丢失，需要进行备份工作。

3.1 备份/还原数据库

这种方法适合于将服务器上数据库文件备份到移动设备（如 U 盘）上，可以在实验结束离开实践中心之后在其它场合继续进行实验。

这种方法对移动设备的要求比较高，特别是对于病毒之类的破坏比较敏感，如果文件有任何损伤就不能进行还原操作，在使用时需要多加注意。

一、备份数据库

1. 启动软件“SSMS”，如

图 3-1 所示。

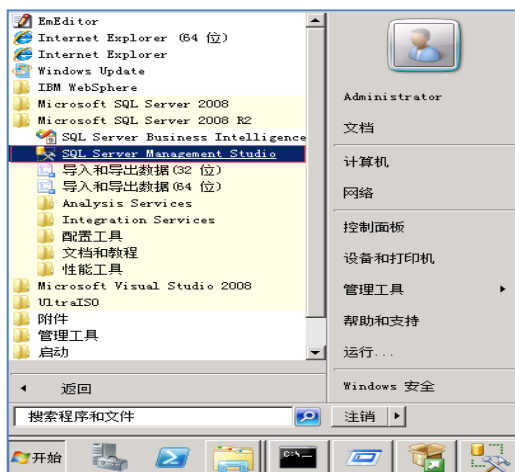


图 3-1 SQL Server Management Studio

2. 通过“Windows 身份验证”方式，“连接”到数据库服务器。



图 3-2 连接到服务器

3. 选中目标数据库，单击鼠标右键,然后选中“备份”菜单。

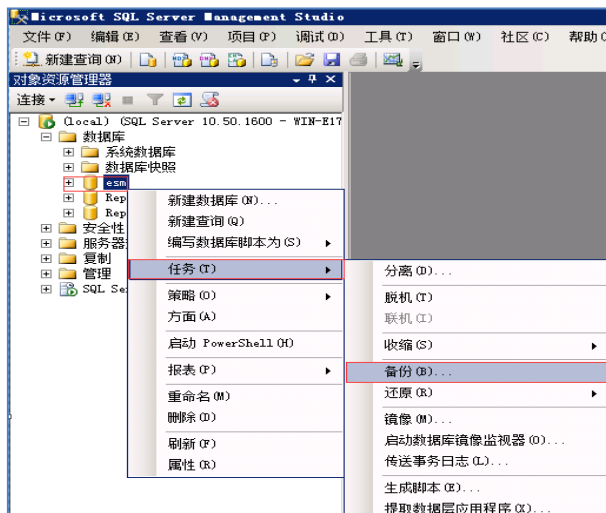


图 3-3 选择“备份”

4. 选择备份目的地，并输入备份数据的文件名

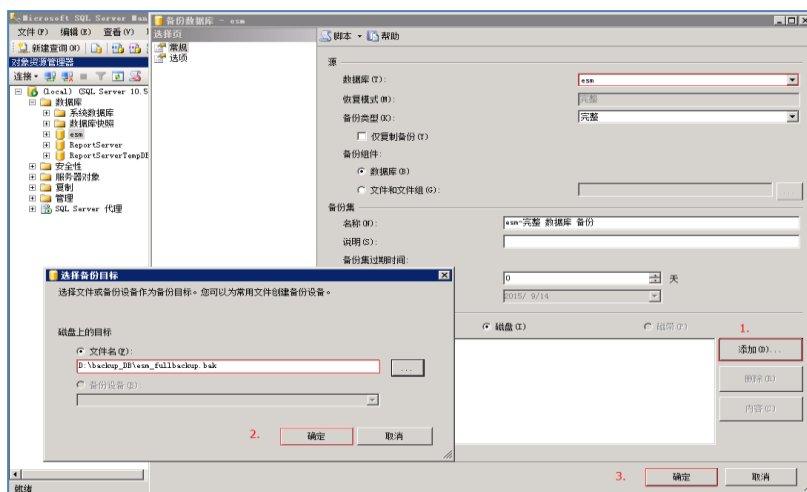


图 3-4 备份数据设置

5. 点击“确定”按钮，至此数据库备份完成。

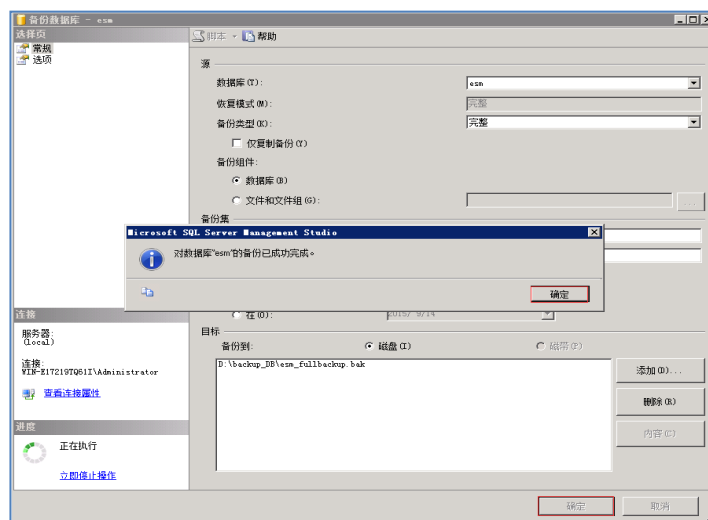


图 3-5 备份完成

二、数据库还原

1. 选中数据库，点击鼠标右键，选中“还原数据库”菜单。

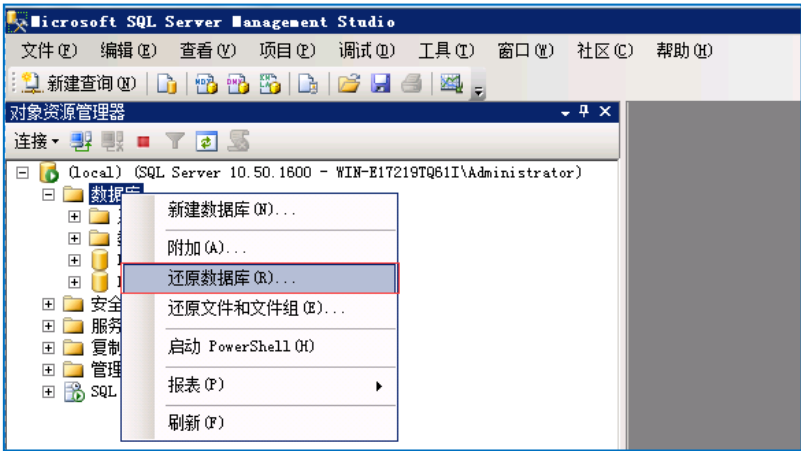


图 3-6 还原数据库

2. 找到数据库备份文件，输入目标数据库名称。

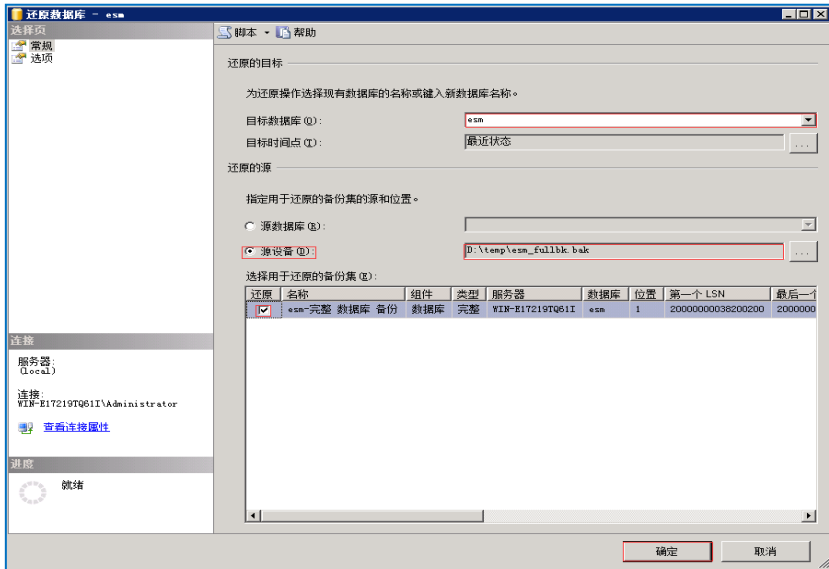


图 3-7 数据库名称

3. 如果只还原数据库完整备份，则在步骤 2 中直接点击“确定”按钮即可。
4. 至此数据还原结束。

3.2 分离/附加数据库

在实验教学过程中，同学们常常想把自己在学校实验室计算机中创建的数据库搬迁到自己的计算机中而不想重新创建该数据库或者有时需要把数据库复制到别的文件或电脑，即可使用分离/附加数据库的方法。

一、分离数据库

分离数据库的目的是将指定的数据库从 SQL Server 中卸载，就如同 SQL Server 中从来没有存在过该数据库一样。具体方法是：

1. 在启动 SSMS 并连接到数据库服务器后，在对象资源管理器中展开服务器节点。在

数据库对象下找到需要分离的数据库名称。右键单击数据库，在弹出的快捷菜单中选择属性项，如图 3- 8 所示，则数据库属性窗口被打开。

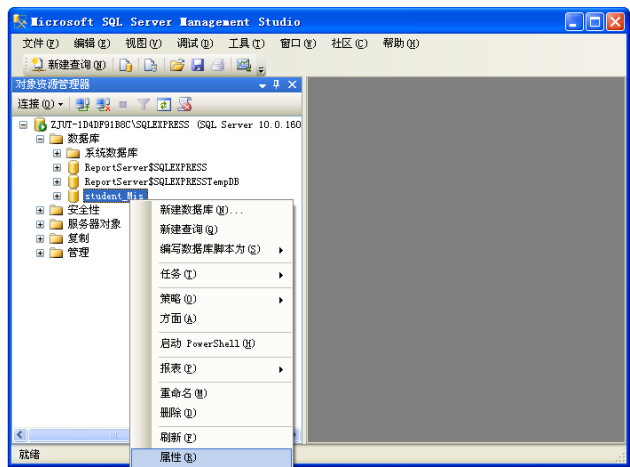


图 3- 8 打开数据库属性窗口

2. 在“数据库属性”窗口左边“选择页”下面区域中选定“选项”对象，然后右边区域的“其它选项”列表中找到“状态”项，单击“限制访问”文本框，在其下拉列表中选择“SINGLE_USER”。

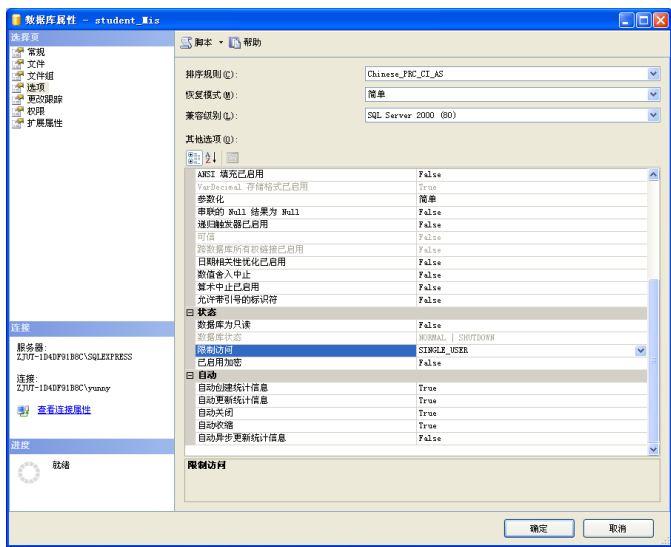


图 3- 9 数据库属性窗口

3. 在图 3- 9 中单击“确定”按钮后将出现一个消息框，通知我们此操作将关闭所有与这个数据库的连接，是否继续这个操作，如图 3- 10 所示。

注意：在大型数据库系统中，随意断开数据库的其它连接是有风险的，因为我们无法知道连接到数据库上的应用程序正在做什么，也许被断开的是一个正在对数据复杂更新操作、且已经运行较长时间的事务，所以此项操作须谨慎。

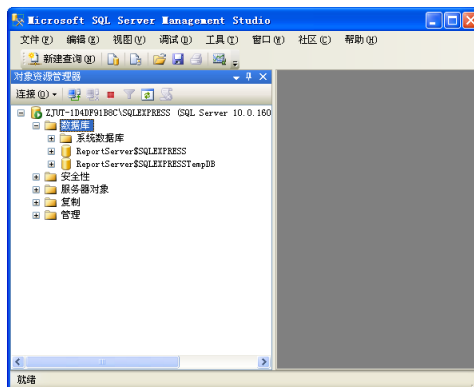


图 3-13 数据库被分离后的 SSMS 窗口

二、附加数据库

在需要使用刚才复制的数据库时,仅仅将移动存储器上的文件复制到相应的文件夹中还不够,还需要将该数据库附加到 SQL Server 中才能正常使用。具体方法是:

1. 将需要附加的数据库文件和日志文件拷贝到某个已经创建好的文件夹中。出于教学目的,我们将该文件拷贝到安装 SQL Server 时所生成的目录 DATA 文件夹中。
2. 在图 3-14 所示的窗口中,右击数据库对象,并在快捷菜单中选择“附加”命令,打开“附加数据库”窗口,如图 3-14 所示。

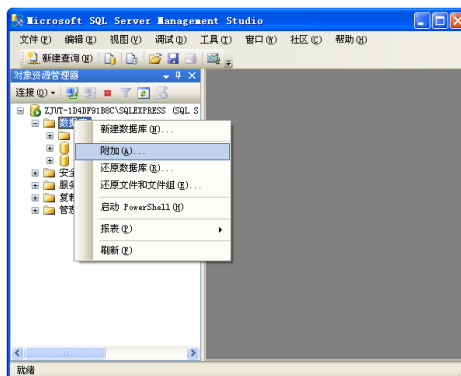


图 3-14 打开附加数据库窗口

3. 在“附加数据库”窗口中,单击页面中间的“添加”按钮,打开定位数据库文件的窗口,在此窗口中定位刚才拷贝到 SQL Server 的 DATA 文件夹中的数据库文件目录,选择要附加的数据库文件。

4. 单击“确定”按钮就完成了附加数据库文件的设置工作。这时,在附加数据库窗口中列出了需要附加数据库的信息,如图 3-16 所示。如果需要修改附加后的数据库名称,则修改“附加为”文本框中的数据库名称。我们这里均采用默认值,因此,单击确定按钮,完成数据库的附加任务。

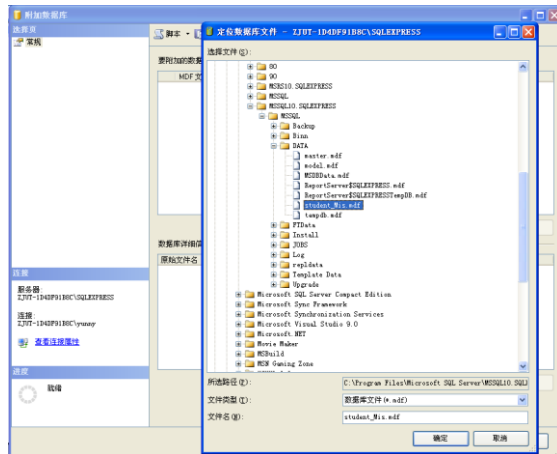


图 3- 15 定位数据库文件到附加数据库窗口中

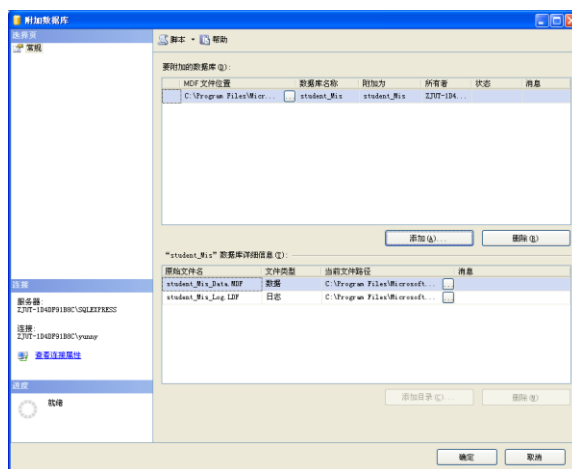


图 3- 16 添加附加的数据库后的附加数据库窗口

完成以上操作，我们在 SSMS 的对象资源管理器中就可以看到刚刚附加的数据库如图 3- 17 所示。

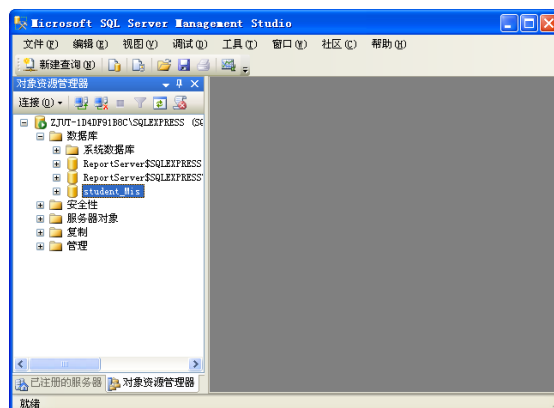


图 3- 17 已经附加了数据库的 SSMS 窗口

以上操作可以看出，如果要将某个数据库迁移到同一台计算机的不同 SQL Server 实例中或其它计算机的 SQL Server 系统中，分离和附加数据库的方法是很有用的。

实验一 SQL 语句练习

4.1 实验目的

1. 加深对表间关系的理解
2. 理解数据库中数据的查询方法和应用
3. 掌握各种查询的异同及相互之间的转换方法
4. 掌握各种查询要求的实现

4.2 实验内容

该实验以 SQL Server 2000 系统自带的 pubs 数据库为例, 以一个图书出版公司为模型。首先按前章介绍的方法将 pubs 数据库附加到当前的 DBMS 中, 再根据要求进行实验。

该数据库的基本表如下所示:

Authors:

属性名	数据类型	含义说明	可为空	检查	键/索引
au_id	Id	作者编号	否	是 1	主键
au_lname	varchar(40)	作者姓	否		
au_fname	varchar(20)	作者名	否		
phone	char (12)	电话	否		
address	varchar(40)	地址	是		
city	varchar(20)	所在城市	是		
state	char(2)	所在州	是		
zip	char(5)	邮编	是	是 2	
contract	Bit	是否签约	否		

(1) au_id CHECK 约束定义为 (au_id LIKE '[0-9][0-9][0-9]-[0-9][0-9]-[0-9][0-9][0-9]').

(2) zip CHECK 约束定义为 (zip LIKE '[0-9][0-9][0-9][0-9][0-9]').

Discounts:

属性名	数据类型	含义说明	可为空	检查	键/索引
discounttype	varchar(40)	折扣类型	否		
stor_id	char(4)	商店编号	是		外键 stores(stor_id)
lowqty	Smallint	数量下限	是		
highqty	Smallint	数量上限	是		
discount	Float	折扣	否		

Employee:

属性名	数据类型	含义说明	可为空	默认值	检查	键/索引
emp_id	Empid	职工编号	否		是 1	主键
Fname	varchar(20)	职工名	否			
Minit	char(1)		是			
Lname	varchar(30)	职工姓	否			
job_id	Smallint	工作编号	否	1		外键 jobs(job_id)
job_lvl	Tinyint		否	10		
pub_id	char(4)	出版社编	否	'9952'		外键
Hire_date	Datetime	工作日期	否	GETDATE()		

(1) CHECK 约束定义为:

(emp_id LIKE '[A-Z][A-Z][A-Z][1-9][0-9][0-9][0-9][FM]') OR

(emp_id LIKE '[A-Z]-[A-Z][1-9][0-9][0-9][0-9][FM]')。

Jobs:

属性名	数据类型	含义说明	可为空	检查	键/索引
Job_id	Smallint	工作编号	否		主键
job_desc	varchar (50)	工作描述	否		
min_lvl	Tinyint		否	是 1	
max_lvl	Tinyint		否	是 2	

(1) min_lvl CHECK 约束定义为 (min_lvl >= 10)。

(2) max_lvl CHECK 约束定义为 (max_lvl <= 250)。

pub_info:

属性名	数据类型	含义说明	可为空	检查	键/索引
pub_id	char(4)	出版社编	否		主键, 外键
Logo	Image	标志图	是		
pr_info	Text	出版信息	是		

Publishers:

属性名	数据类型	含义说明	可为空	检查	键/索引
pub_id	char(4)	出版社编号	否	是 1	主键
pub_name	varchar(40)	出版社名称	是		
city	varchar(20)	所在城市	是		

state	char(2)	所在州	是		
country	varchar(30)	所在国家	是		

(1) pub_id CHECK 约束定义为 (pub_id = '1756' OR (pub_id = '1622' OR (pub_id = '0877' OR (pub_id = '0736' OR (pub_id = '1389')))) OR (pub_id LIKE '99[0-9][0-0]'))。

Roysched:

属性名	数据类型	含义说明	可为空	检查	键/索引
title_id	Tid	书编号	否		外键 titles(title_id)
lorange	Int	低	是		
hirange	Int	高	是		
royalty	Int	版权	是		

Sales:

属性名	数据类型	含义说明	可为空	键/索引
stor_id	char(4)	商店编号	否	组合主键, 聚集索引, 外键
ord_num	varchar(20)	订单编码	否	组合主键, 聚集索引
ord_date	Datetime	订购日期	否	
qty	Smallint	数量	否	
payterms	varchar	付款方式	否	
title_id	Tid	书编号	否	组合主键, 聚集索引, 外键

Titles:

属性名	数据类型	含义说明	可为空	检查	键/索引
title_id	Tid	书编号	否		主键
title	varchar(80)	书名	否		
type	char (12)	类型	否		
pub_id	char(4)	出版社编号	是		外键 publishers (pub_id)
price	Money	价格	是		
advance	Money	预付款	是		
royalty	Int	版税	是		
Ytd_sales	Int	年销售量	是		
notes	varchar(200)	简介	是		
pubdate	Datetime	出版日期	否		

Stores:

属性名	数据类型	含义说明	可为空	检查	键/索引
stor_id	char(4)	商店编号	否		主键
stor_name	varchar(40)	商店名称	是		
stor_address	varchar(40)	商店地址	是		
city	varchar(20)	所在城市	是		
state	char(2)	所在州	是		
zip	char(5)	邮编	是		

Titleauthor:

属性名	数据类型	含义说明	可为	检	键/索引
au_id	id	作者编号	否		组合主键，聚集索引,外键
title_id	tid	书编号	否		组合主键，聚集索引,外键
au_ord	tinyint		是		
royaltyper	int	版权百分比	是		

4.3 实验任务

实验任务一

1. 查询所有作者的作者号、姓名信息
2. 查询所有作者的姓名、作者号信息，并在每个作者的作者号前面显示字符串“身份证号:”，表明显示的信息是身份证信息
3. 查询在 CA 州的作者姓名和城市
4. 查询出版日期在 2000. 1. 1-2000. 12. 31 之间的书名和出版日期
5. 查询每个出版社出版的书
6. 查询某店销售某书的数量
7. 查询有销售记录的所有书信息（书号、书名、类型和价格等）
8. 显示所有的书名（无销售记录的书也包括在内）
9. 查询已销售书的信息（书号、书名、作者等）
10. 查询所有出版商业（business）书籍的出版社的名称

实验任务二

在任务一的基础上，练习查询语句的使用，包括计算列、求和、最大、最小值、各类选择条件、字符匹配、分组和排序，体会各种查询的执行过程，为简单综合应用打下基础。

1. 查询书名以 T 开头或者出版社号为 0877，而且价格大于 16 的书的信息。
2. 按照类型的升序和价格的降序（在类型相同时）显示书的信息（书名、作者、出版社、类型、价格）

3. 查询销售量大于 30 的书名及销售数量
4. 查询在 1990.1.1 到 2000.12.31 间，每本书的销售总额
5. 查询所有作者的所在城市和州名，要求没有重复信息
6. 计算多少种书已被订价
7. 查询每本书的书名、作者及它的售书总量
8. 计算所有书的平均价格
9. 查询价格最高的书的书名、作者及价格

实验任务三

1. 参照以上各表给出的主键、外键设置的设置要求，在自己创建表中进行相应的设置。
2. 向 authors 表中插入一行作者信息（具体值自定）
3. 数量超过 100 的商店增加 10% 的折扣
4. 删除 1994.9.14 的订单
5. 删除 1 中所建立的索引
6. 建立 CA 州作者所著书的视图（包括作者号、姓名、所在州、书名、价格、出版日期）
7. 建立付款方式是现金（cash）的订单视图
8. 建立 CA 州的所有商店的视图

4.4 实验报告要求

1. 写出与上述任务相对应的 SQL 语句；
2. 并记录在实验过程中遇到的问题、解决办法及心得体会。

实验二 SQL 数据库定义与操作

5.1 实验目的

1. 熟悉 SQL Server 2008 中 SQL Server Management Studio 的环境
2. 了解 SQL Server 2008 数据库的逻辑结构和物理结构
3. 掌握使用向导方式和 SQL 方式来创建和删除数据库及数据表的方法

5.2 实验内容

在本练习中，将使用 SQL Server 2008 数据库管理系统创建“S+学号”数据库，并在数据库中创建表 Student，学习使用 SQL Server2008 数据库管理系统创建数据库和表的方法。

1. 启动 SQL Server2008:

依次单击开始→所有程序→SQL Server 2008→SQL Server Management Studio 启动 SQL Server 2008 数据库管理系统

2. 登录数据库服务器:

点击“连接到服务器”对话框中的连接按钮连接到 SQL Server 2008 数据库服务器，如图 5-1 所示。

说明：学生使用自己的机器或课堂的机器进行实验时，可以采用“Windows 身份验证”的方法进行登录，但在实验结束，需要把实验结果附加到信息学院实验室的服务器上，用户名、密码都是“S+学号”，数据库的名字也是“S+学号”，后续数据库登录与此相同。为后续实验中所有的表、视图等对象都创建在“S+学号”数据库里。



图 5-1 连接到服务器

3. 创建数据库, 数据库的名称统一为: “S+学号”，信息学院服务器上的数据库名称与用户在课堂或课后实验时使用的数据库名称要求相同，前者已建立完成，后者需要实验者自己完成。

在 SQL Server 2008 数据库管理系统的左边栏“对象资源管理器”中右击数据库对象，在弹出的快捷菜单中单击“新建数据库”命令。

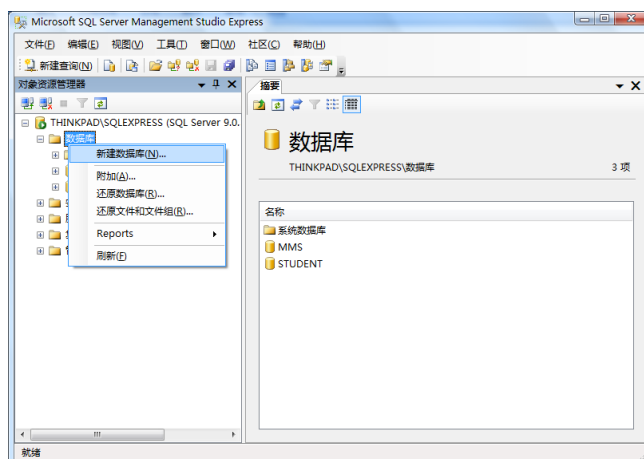


图 5-2 新建数据库

在如图 5-3 所示的“新建数据库”对话框右侧的数据库名称中输入数据库名称“S+学号”，然后单击确定。一般来说，系统会将数据库以文件的形式存放在默认的文件夹内，实验者要记住，以便实验结束时自己备份或分离。

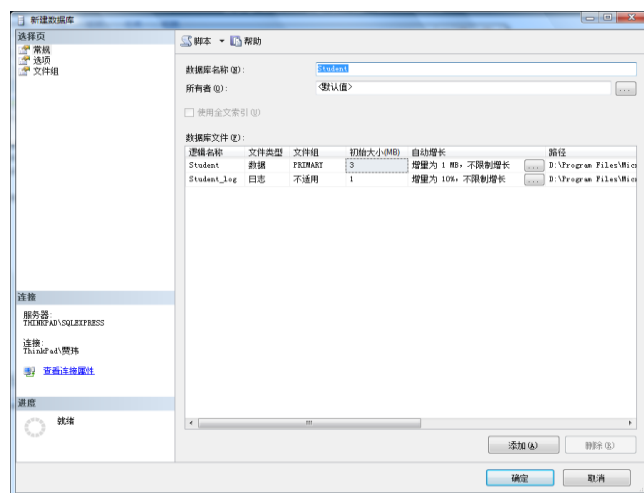



图 5-3 数据库名称

方法 1：在 SSMS 中创建表“Student”

单击 SQL Server 2008 数据库管理系统的左侧的“对象资源管理器”栏中的“刷新”按钮，以显示出已有的数据库“S+学号”。

依次展开左侧栏对象资源管理器中的“数据库”—>“S+学号”，并右击“S+学号”数据库中的表项目，在弹出的快捷菜单中单击“新建表”命令。

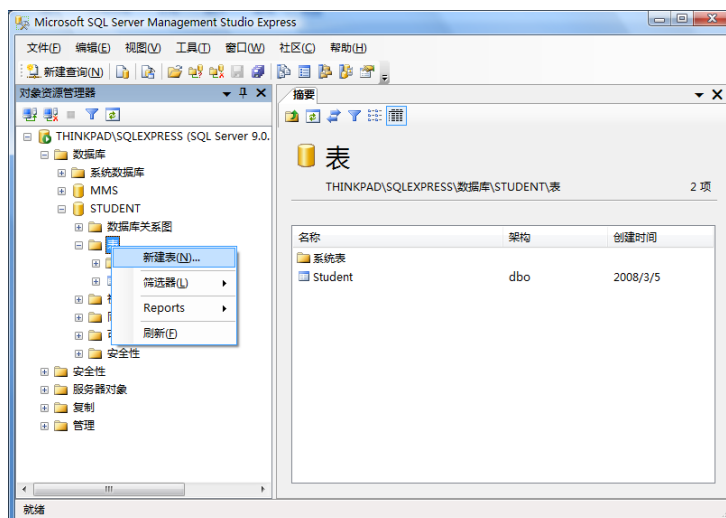


图 5-4 新建表

在右侧在工作区中输入“Student”表的信息，该表具有如下列：

列名	数据类型	完整性约束
Sno	CHAR(5)	主码
Sname	CHAR(10)	非空
Ssex	BIT	无
Sage	INT	无
Sdept	CHAR (15)	无

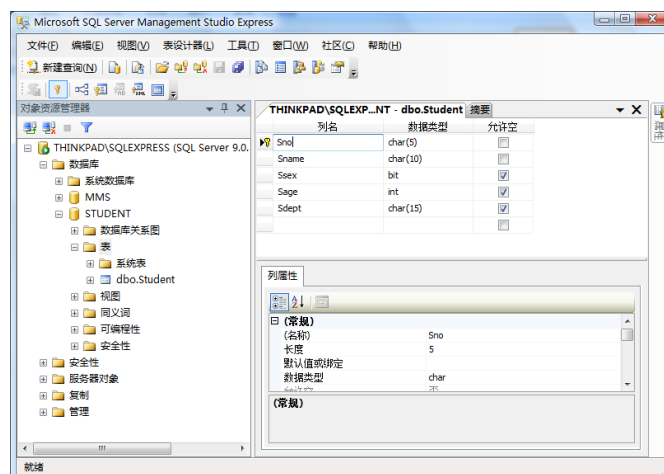


图 5-5 字段设置

单击文件菜单中的保存命令保存该表，并取名为“Student”。

方法 2：使用 SQL 语句创建表“Student”

1. 单击工具栏中的“新建查询”按钮，并在弹出的“连接到服务器”对话框中单击“连接”按钮，新建一个 SQL 脚本。

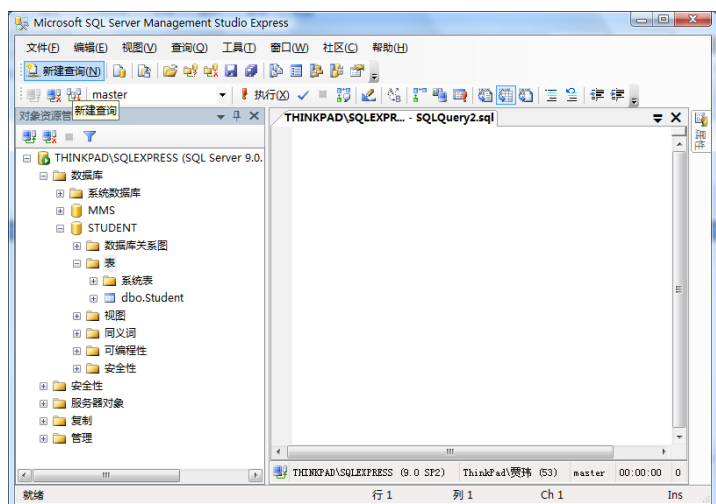



图 5-6 新建查询

2. 在右侧的 SQL 脚本输入框中输入如下 SQL 代码：

```
CREATE DATABASE S+学号 -- 创建“S+学号”数据库
GO
USE S+学号 -- 转到“S+学号”数据库
GO
CREATE TABLE Student -- 在“S+学号”数据库中创建表STUDENT
(
    Sno CHAR(5)PRIMARYKEY,
    Sname CHAR(10)NOTNULL,
    Ssex BIT,
    Sage INT,
    Sdept CHAR (15)
)
```

3. 单击工具栏中的  执行按钮，运行 SQL 语句，完成数据库与表的创建。
4. 使用同样的方法，创建教材中 Course 和 SC 数据表。

5.3 实验任务

1. 打开自己专属的数据库，建立如下所示的三个表：

Employee 表

列名	数据类型	长度	是否允许为空	说明
EmployeeID	Char	6	否	员工编号，主键
Name	Char	10	否	姓名
Birthday	Datetime	8	否	出生日期
Sex	Bit	1	否	性别
Address	Char	20	是	地址
Zip	Char	6	是	邮编

PhoneNumber	Char	12	是	电话号码
EmailAddress	Char	30	是	电子邮件地址
DepartmentID	Char	3	否	员工部门号，外键

Departments 表

列名	数据类型	长度	是否允许为空	说明
DepartmentID	Char	3	否	员工部门号，主键
DepartmentName	Char	20	否	部门名
Note	Text	16	是	备注

Salary 表

列名	数据类型	长度	是否允许为空	说明
EmployeeID	Char	6	否	员工编号，外键
Income	Float	8	否	收入
OutCome	Float	8	否	支出

5.4 实验报告要求


1. 要求给出三个表创建的 SQL 语句，要求建表时给出主外键约束、唯一约束、取空值约束、用户自定义的约束等；
2. 画出这三个表的关系图；
3. 举例说明如何操作会违背完整性（实体完整性、参照完整性）。

实验三 SQL 数据查询语句

6.1实验目的

- 1. 掌握 SELECT 语句的基本语法和查询条件表示方法
- 2. 掌握 GROUP BY 和 ORDER BY 子句的作用和使用方法
- 3. 掌握连接查询和子查询的使用方法

6.2 实验内容

- 1. 单击工具栏中的“新建查询”按钮，并在弹出的“连接到服务器”对话框中单击“连接”按钮，新建一个 SQL 脚本。
 - 2. 在 SQL 脚本中写数据查询语句执行。在查询分析器的查询窗口中输入 SQL 语句。
- 点击  执行(X) 按钮,执行该 SQL 语句,在查询窗口下部出现一个输出窗口,如图 6- 1 所示。

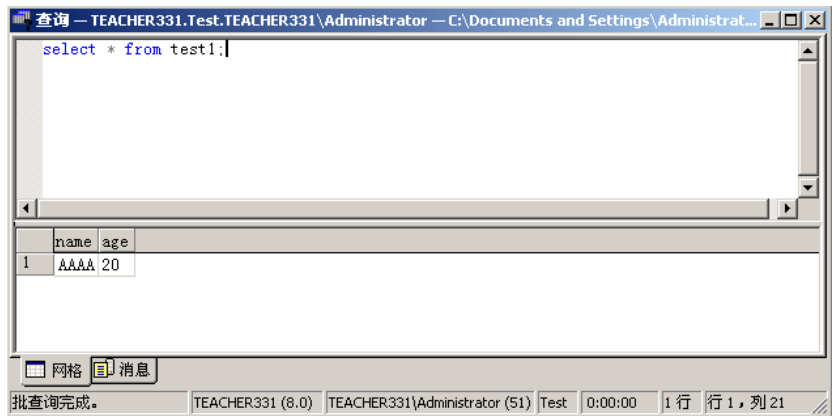


图 6- 1 执行查询

6.3 实验任务

- 1. 对上节建立的表输入数据：

Departments 表：

DepartmentID	DepartmentName	Note
1	财务部	财务部
2	研发部	研发部
3	人力资源部	人力资源部

Employee 表

EmployeeID	Name	Birthday	Sex	Address	Zip	PhoneNumber	EmailAddress	DepartmentID
1001	李勇	78-3-12	0	河南	475001	3880378	ly@henu.edu.cn	1
1002	王敏	80-11-2	1	河南	475002	0378311	wm@henu.edu.cn	1
1003	刘晨	78-6-22	0	河南	475003	0378322	lc@henu.edu.cn	1
1004	周宏	83-10-3	1	河北	475004	7865987	zh@ henu.edu.cn	1
2001	张立	78-8-1	0	河南	475005	0378333	zl@henu.edu.cn	2
2002	刘毅	82-1-23	0	河南	475006	0378344	ly@henu.edu.cn	2

2003	张玫	81-3-15	1	河南	475007	0378355	zm@henu.edu.cn	2
2004	王军	79-5-12	0	山东	475008	5687967	wj@henu.edu.cn	2
3001	徐静	76-8-12	1	河南	475009	0378366	xj@henu.edu.cn	3
3002	赵军	79-2-19	0	河南	475010	0378377	zj@henu.edu.cn	3
3003	王霞	82-8-18	1	湖南	475011	7556677	wx@henu.edu.cn	3

Salary 表

EmployeeID	Income	OutCome
1001	3600	1500
1002	3300	1000
1003	3700	1200
1004	4500	1600
2001	4000	1600
2002	3800	1800
2003	3800	1500
2004	5100	1800
3001	4200	2000
3002	4100	1800
3003	4600	1400

2. 练习下面简单的查询语句：

- (1) 查询每个雇员的所有信息
- (2) 查询每个雇员的地址和电话
- (3) 查询 EmployeeID 为 000001 的雇员的地址和电话。
- (4) 查询女雇员地址和电话，并用 AS 子句将结果中各列的标题分别指定为“地址”和“电话”。
- (5) 计算每个雇员的实际收入。
- (6) 找出所有姓王的雇员的部门号。
- (7) 找出所有地址中含有“中山”的雇员的号码和部门号。

3. 练习多表连接查询和嵌套查询：

- (1) 查询每个雇员的情况及工资情况（工资=Income - Outcome）
- (2) 查询财务部工资在 2200 元以上的雇员姓名及工资情况
- (3) 查询研发部在 1966 年以前出生的雇员姓名及其工资详情
- (4) 查询人力资源部雇员的最高和最低工资
- (5) 将各雇员的情况按工资由低到高排列

- (6) 求各部门的雇员数
- (7) 找出所有在财务部和人力资源部工作的雇员的编号
- (8) 统计人力资源部工资在 2500 以上雇员的人数
- (9) 求财务部雇员的总人数
- (10) 求财务部雇员的平均工资
- (11) 查找比所有财务部的雇员工资都高的雇员的姓名
- (12) 查找财务部年龄不低于研发部所有雇员年龄的雇员的姓名
- (13) 查找在财务部工作的雇员的情况

6.4 实验报告要求

1. 写出与上述查询任务相对应的 SQL 查询语句；
2. 并记录在实验过程中遇到的问题、解决办法及心得体会。

实验四 SQL 数据更新语句

7.1 实验目的

1. 加深对 SQL 数据更新（插入、修改及删除）语句的基本语法格式的掌握
2. 掌握单个元组及多个元组的插入、修改及删除操作的实现过程
3. 加深对更新操作时数据库中数据一致性问题的了解
4. 加深对约束条件在数据更新操作执行中的作用问题的了解

7.2 实验内容

1. 在 SSMS 中插入、删除及修改数据，
2. 在查询编辑器中写 SQL 插入、删除及修改数据


7.3 实验任务

打开自己专属的数据库，创建 S, P, J, SPJ 四个关系模式：


S (SNO, SNAME, STATUS, CITY)

表 - dbo.S	表 - dbo.P	表 - dbo.J	摘要
列名	数据类型	允许空	
 SNO	char(2)	<input type="checkbox"/>	
SNAME	varchar(20)	<input checked="" type="checkbox"/>	
STATUS	char(4)	<input checked="" type="checkbox"/>	
CITY	char(10)	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

P (PNO, PNAME, COLOR, WEIGHT)

列名	数据类型	允许空
 PNO	char(2)	<input type="checkbox"/>
PNAME	varchar(20)	<input checked="" type="checkbox"/>
COLOR	char(4)	<input checked="" type="checkbox"/>
WEIGHT	int	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

J (JNO, JNAME, CITY)

列名	数据类型	允许空
 JNO	char(2)	<input type="checkbox"/>
JNAME	varchar(20)	<input checked="" type="checkbox"/>
CITY	char(10)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

SPJ (SNO, PNO, JNO, QTY)

	列名	数据类型	允许空
主键	SNO	char(2)	<input type="checkbox"/>
	PNO	char(2)	<input type="checkbox"/>
	JNO	char(2)	<input type="checkbox"/>
	QTY	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

供应商表 S: 由供应商代码 (SNO)、供应商姓名 (SNAME)、供应商(STATUS)、供应商所在城市 (CITY) 组成;

零件表 P: 由零件代码 (PNO)、零件名 (PNAME)、颜色 (COLOR)、重量 (WEIGHT) 组成;

工程项目表 J: 由工程项目代码(JNO)、工程项目名(JNAME)、工程项目所在城市(CITY) 组成;

供应情况表 SPJ: 由供应商代码 (SNO)、零件代码 (PNO)、工程项目代码 (JNO)、供应数量 (QTY) 组成, 表示某供应商供应某种零件给某工程项目的数量为 QTY。试完成以下操作:

- (1) 在 S 表中插入元组“s6, 华誉, 40, 广州”
- (2) 在 J 表中插入元组“j8, 传感器厂”
- (3) 对每一个供应商, 求他为各种工程供应零件的总数量, 并将此结果存入数据库
- (4) 将 P 表中 PNO 值为 p6 的元组的 color 属性值改为绿, weight 属性值改为 60
- (5) 将 SPJ 表中前 4 个元组的 qty 属性值统一修改为 300
- (6) 将 S 表中 city 属性名含有“京”或“津”的相应 status 属性值增加 100
- (7) 将供应商 s2 为“一汽”工程项目所供应的零件数量修改为 2000
- (8) 将全部红色零件的颜色修改为浅红色
- (9) 将由 s5 供给 j4 的零件 p6 改为由 s3 供应
- (10) 在 SPJ 表中新增一名为 SDATE 的属性列, 对该表中的每一元组在 SDATE 属性列上填上实验当时的日期和时间
- (11) 删除所在城市为“广州”的供应商记录
- (12) 删除所有零件名称中第一个字为“螺”字的零件记录, 并在供应情况表中删除相应的记录
- (13) 删除 s3 和 s4 两供应商为“中建”工程供应“螺母”或“螺丝刀”零件的相应供应情况数据信息

7.4 实验报告要求

1. 写出与上述任务相对应的 SQL 更新语句;
2. 并记录在实验过程中遇到的问题、解决办法及心得体会。

实验五 视图的定义与操作

8.1 实验目的

1. 掌握创建视图的 SQL 语句的用法
2. 掌握修改视图的方法
3. 熟悉视图更新与基本表更新的区别与联系
4. 认识视图的作用

8.2 实验内容

1. 使用 SSMS 和 Transact-SQL 语句 CREATE VIEW 创建视图
2. 使用 Transact-SQL 语句 ALTER VIEW 修改视图
3. 学习删除视图的 Transact-SQL 语句 DROP VIEW 的用法

8.3 实验任务

1. 对应 HRM 数据库，参照前面实验中完成的查询，按如下要求自行设计视图：

(1) 基于单个表按投影操作定义视图。

举例：定义一个视图用以查看所有员工的编号、姓名和出生日期。

(2) 基于单个表按选择操作定义视图。

举例：定义一个满足 sex='true' 的员工的所有信息的视图。

(3) 基于单个表按选择和投影操作定义视图。

举例：定义一个视图用以查看部门号码为'2'的所有员工的姓名、电话和邮件地址。

(4) 基于多个表根据连接操作定义视图。

举例：定义一个视图用以查看所有员工的姓名、部门名及工资。

(5) 基于多个表根据嵌套查询定义视图。

举例：定义一个比所有财务部的雇员工资都高的雇员的信息的视图

(6) 定义含有虚字段（即基本表中原本不存在的字段）的视图。

举例：定义一个视图用以查看所有雇员的编号、姓名、年龄。

2. 分别在定义的视图上进行查询、插入、更新和删除操作，分情况（查询、更新）讨论哪些操作可以成功完成，哪些不能成功完成，并分析原因。

8.4 实验报告要求

1. 在实验报告中要给出上述任务的视图定义语句；
2. 分情况讨论哪些操作可以成功完成，哪些不能成功完成；
3. 查询操作：举出一个进行查询操作的例子；
4. 更新操作：更新操作包含插入、更新、删除，针对不能成功完成更新操作的情况举出两个例子，并分析原因。

实验六 数据库完整性的管理


9.1 实验目的

1. 掌握实体完整性的实现方法
2. 掌握域完整性的实现方法
3. 掌握参照完整性的实现方法
4. 掌握用户定义完整性的实现方法

9.2 实验内容

一、可视化界面的操作方法：

1. 实体完整性

(1) 将 Student 表的“Sno”字段设为主键：在表设计界面中，单击左边的行选定块，选定“Sno”字段，单击工具按钮 设置主键。如图 9-1 所示：

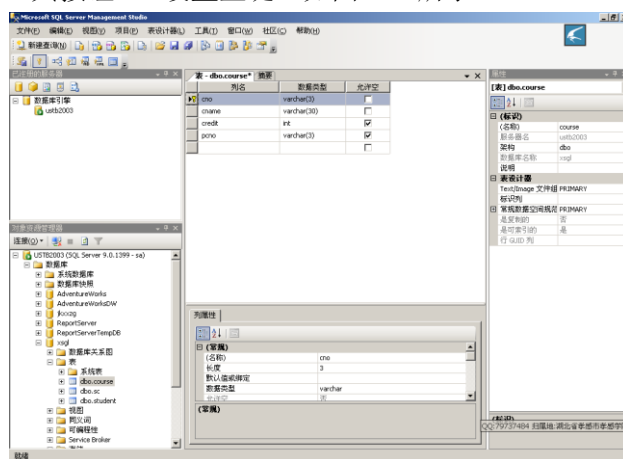



图 9-1 设置主键

(2) 将“SC”表的“Sno”和“Cno”设置为主键：在表设计界面中，单击并按住 Ctrl 键拖动左边的行选定块，选定 Sno 和 Cno 字段，单击工具按钮 设置主键。

2. 域完整性

(1) 将“Ssex”字段设置为只能取“男”，“女”两值：在表设计界面，点击鼠标右键—CHECK 约束—添加约束，添加 CK_Student_sex 名称，然后在约束表达式框中输入“Ssex in ('男','女')”。如图 9-2 所示。

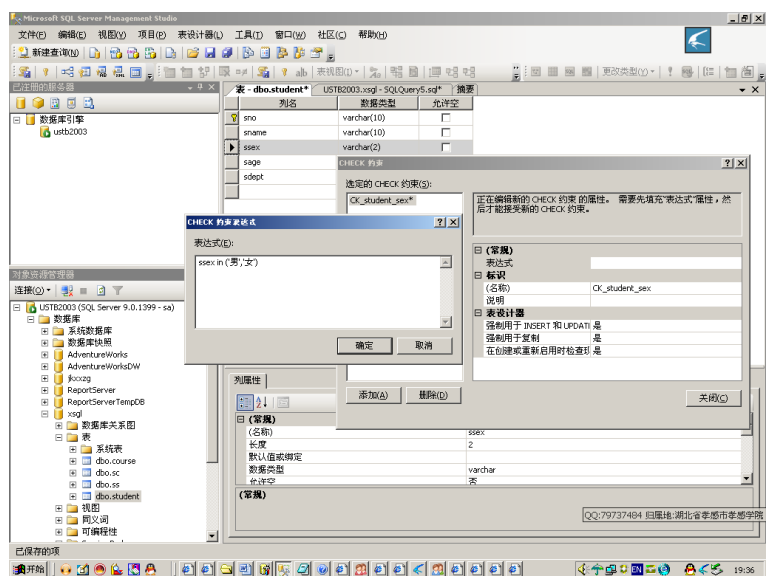



图 9- 2 CHECK 设置

3. 参照完整性

(1) “Student”表和“SC”表中的“Sno”字段设为参照: 打开“SC”表的设计界面, 点击工具栏按钮 , 在弹出的属性(properties)对话框中点击“新建”按钮, 在“主键表(Primary key table)”下拉框中选择“Student”表, 在其下的字段选择框中选择“Sno”, 在“外键表(Foreign key table)”下拉框中选择“SC”表, 在其下的字段选择框中选择“Sno”, 单击关闭即可, 如图 9- 3 所示。

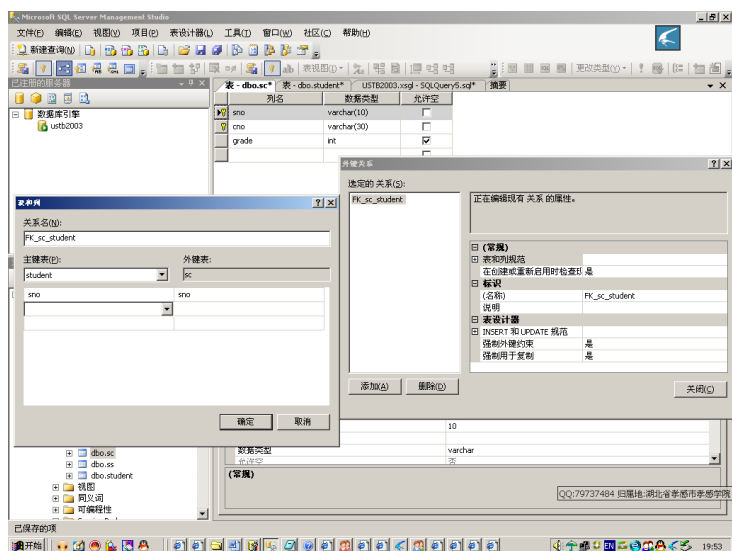


图 9- 3 参照完整性

命令方式操作方法:

实体完整性

1. 将“Student”表的“Sno”字段设为主键:

当“Student”表已存在则执行:

Alter table Student add constraint pk_Sno primary key (Sno)

当“Student”表不存在则执行:

```
Create table Student(Sno CHAR(5) primary key ,
Sname CHAR(10) NOT NULL,
Ssex CHAR(2),
Sage int,
Sdept CHAR(4))
```

注:可用命令“drop table Student”删除“Student”表

2. 添加一身份证号字段,设置其惟一性.(注: 操作前应删除表中的所有记录)

```
Alter table Student add id char(18) unique (id)
```

3. 将“SC”表的“Sno”和“Cno”设置为主键:

当“SC”表已存在则执行:

```
alter table SC add constraint PK_SnoCno primary key (Sno,Cno)
```

当“SC”表不存在则执行:

```
Create table SC(Sno CHAR(5),
Cno CHAR(2),
Grade INT NULL,
constraint PK_SnoCno primary key (Sno,Cno))
```

域完整性

4. 将“Ssex”字段设置为只能取“男”, “女”两值:

当“Student”表已存在则执行:

```
alter table Student add constraint CK_Sex check (Ssex in ('男','女'))
```

当“Student”表不存在则执行:

```
Create table Student(Sno CHAR(5) primary key ,
Sname CHAR(10),
Ssex CHAR(2) check (Ssex in ('男','女')),
Sage int, Sdept CHAR(4))
```

5. 设置学号字段只能输入数字:

```
alter table Student add constraint CK_Sno_Format check (Sno like
'[0-9][0-9][0-9][0-9][0-9]')
```

6. 设置身份证号的输入格式:

```
alter table Student add constraint CK_ID_Format check ((id like
'[0-9][0-9][0-9][0-9][0-9][0-9][1-2][0-9][0-9][0-9][0-1][0-9][0-3][0-9][0-9][0-9][0-9]_') OR (id
like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-1][0-9][0-3][0-9][0-9][0-9][0-9]'))
```

7. 设置 18 位身份证号的第 7 位到第 10 位为合法的年份(1900-2050)

alter table Student add constraint CK_ID_Format2 check (len(id)=18 and ((convert(smallint,substring(id,7,4))>=1900) and(convert(smallint,substring(id,7,4))<=2050)))

8. 设置男生的年龄必须大于 22, 女生的年龄必须大于 20.

Alter table Student add constraint CK_age check (Ssex='男' and Sage>=22 or Ssex='女' and Sage>=20)

参照完整性

9. 将“Student”表和“SC”表中的“Sno”字段设为参照:

当“SC”表已存在则执行:

alter table SC add constraint FP_Sno foreign key (Sno) references Student(Sno)

当“SC”表不存在则执行:

Create table SC(Sno CHAR(5) constraint FP_Sno

foreign key references Student(Sno),

Cno CHAR(2),Grade INT NULL,

constraint PK_SnoCno primary key (Sno,Cno))

完整性验证

1. 实体完整性: 在“Student”表数据浏览可视化界面中输入学号相同的两条记录将会出现错误如图 5 所示:

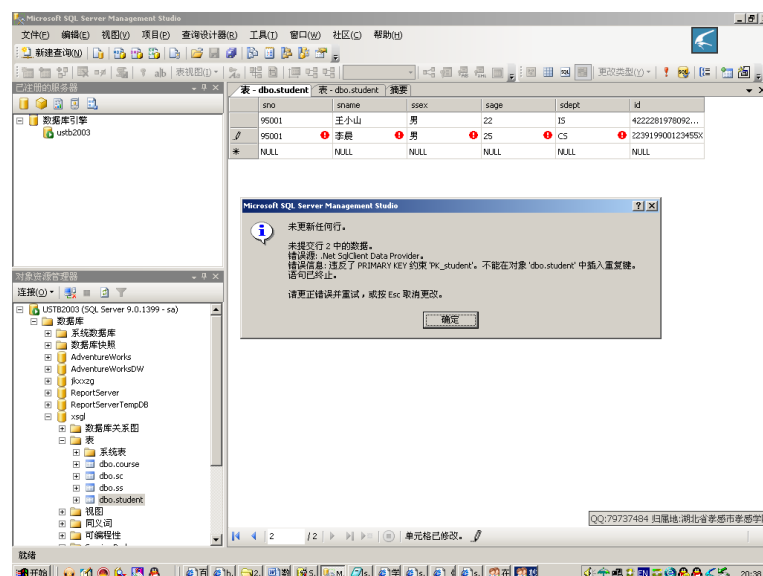


图 5

或者在命令窗口输入下面两条命令也会出现错误提示:

Insert into Student values('95003','张三','男',24,'CS','42222919901012903X')

insertinto Student values('95001','李四','女',21,'CS','422229199510129031')

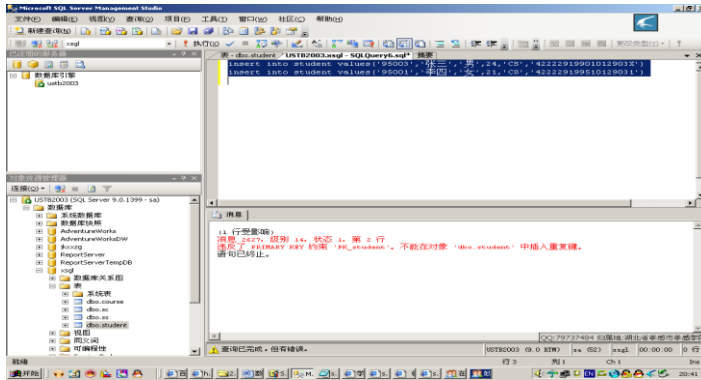


图 6

下面的语句用来验证“SC”表中的实体完整性:

insert into SC values('95002', '10',65)

insert into SC values('95002', '10',90)

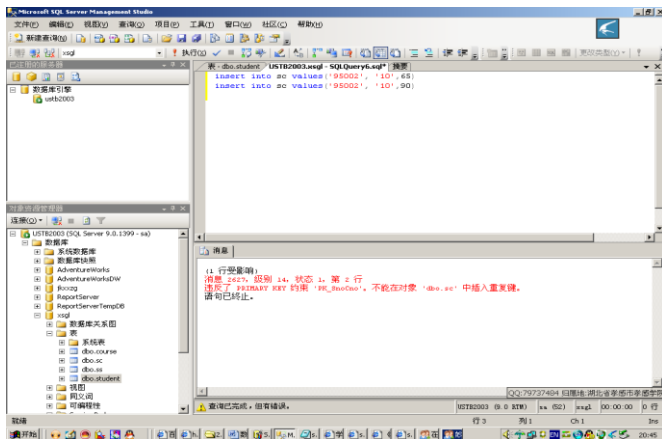
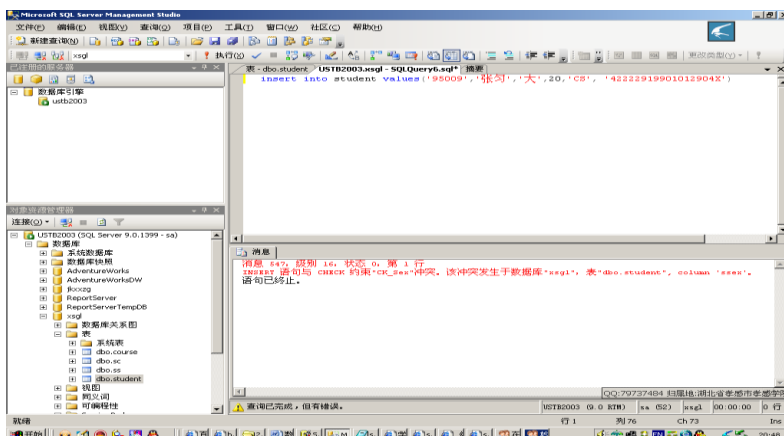


图 7

2. 域完整性:

使用下面的语句验证“Ssex”字段的域完整性:

insert into Student values('95009','张勾','大',20,'CS', '42222919901012904X')

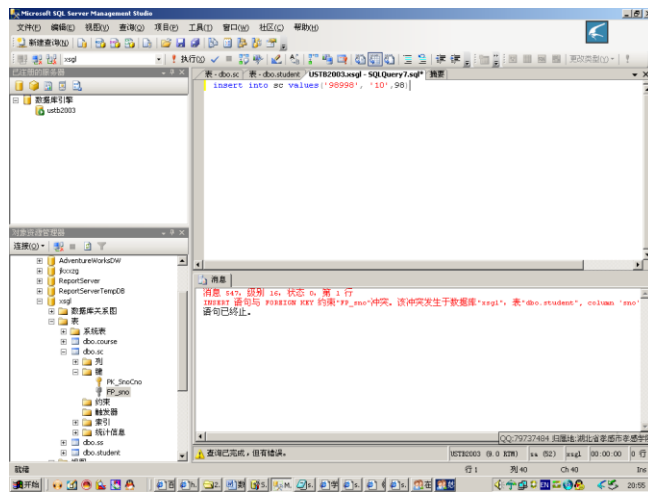


3. 参照完整性:

使用下面的语句“验证”SC 表中的“Sno”字段的域完整性(假设 Student 表中没有学号为

“95998”的学生记录):

insert into SC values('98998','10',98)



9.3 实验任务

1. 建立课程的实体完整性，和课程号 Cno 的参照完整性；
2. 对 HRM 数据库，练习建立三个表的主外键约束、唯一约束、取空值约束、用户自定义的约束（参考 HRM 数据库表定义图中说明）；
3. 建立 salary 表的 Income 字段限定在 0-9999 之间。

9.4 实验报告要求

1. 写出与上述任务相对应的 SQL 语句；
2. 并记录在实验过程中遇到的问题、解决办法及心得体会。

实验七 数据库安全性的管理

10.1 实验目的

1. 掌握 SQL Server 身份验证模式。
2. 掌握创建登录账户、数据库用户的方法。
3. 掌握使用角色实现数据库安全性的方法。
4. 掌握权限的分配。

10.2 实验内容

1. 设置身份验证模式

(1) 依次单击开始—>所有程序—>SQL Server 2008—>SQL Server Management Studio 启动 SQL Server 2008 数据库管理系统；

(2) 连接成功后，右键你的实例，选择“属性”；

(3) 在“属性”窗口中，转到“Security”(安全性)项，在“服务器身份验证”中设置为“SQL Server 和 Windows 身份验证模式”，确定，根据提示，重新启动 sql 服务；

(4) 然后执行下面的语句启用 sa 用户，同时清除 sa 的密码(能成功登陆后再根据你的需要设置)

```
EXEC sp_password null,null,'sa'
```

```
ALTER LOGIN sa ENABLE
```

(5) 语句执行完成后，再用 sa 连接实例，应该就没有问题了。

2. 设置登录账户

● 创建使用 Windows 身份验证 (SSMS) 的 SQL Server 登录名

(1) 在 SSMS 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。

(2) 右键单击“安全性”文件夹，指向“新建”，然后单击“登录名”。

(3) 在“常规”页上的“登录名”框中输入一个 Windows 用户名。

(4) 选择“Windows 身份验证”。

(5) 单击“确定”。

● 创建使用 SQL Server 身份验证 (SSMS) 的 SQL Server 登录名

(1) 在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。

(2) 右键单击“安全性”文件夹，指向“新建”，然后单击“登录名”。

(3) 在“常规”页上的“登录名”框中输入一个新登录名的名称。

(4) 选择“SQL Server 身份验证”。

(5) 输入登录名的密码。

(6) 选择应当应用于新登录名的密码策略选项。通常，强制密码策略是更安全的选

择。

- 通过 Transact-SQL 创建使用 Windows 身份验证的 SQL Server 登录名

(1) 在查询编辑器中，输入以下 Transact-SQL 命令：

```
CREATE LOGIN <name of Windows User>  
FROM WINDOWS;  
GO
```

- 通过 Transact-SQL 创建使用 SQL Server 身份验证的 SQL Server 登录名

(1) 在查询编辑器中，输入以下 Transact-SQL 命令：

```
CREATE LOGIN <login name>  
WITH PASSWORD = '<password>';  
GO
```

3. 设置数据库用户

- 创建数据库用户

(1) 在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。

(2) 依次展开数据库，某个具体数据库，“用户”

(3) 右键单击用户，选择“新建用户”

(4) 输入用户名，选择登录名，确定

- 通过 Transact-SQL 创建数据库用户 (create user):

(1) 输入 SQL 语句：

```
create user dba  
for login dba  
with default_schema=dbo
```

(2) 并指定数据库用户“dba”的默认 schema 是“dbo”。这意味着用户“dba”在执行“select * from t”，实际上执行的是“select * from dbo.t”。

4. 设置数据库角色

- 新建角色

(1) 在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中创建新登录名的服务器实例的文件夹。

(2) 依次展开数据库，某个具体数据库，“角色”

(3) 右击选中新建角色

- 通过 Transact-SQL 新建角色：

(1) CREATE ROLE role_name

- 在对象资源管理器中设置数据库角色

(1) 在 SQL Server Management Studio 中，打开对象资源管理器并展开要在其中

创建新登录名的服务器实例的文件夹。

(2) 依次展开数据库，某个具体数据库，“用户”选中某用户，右击“属性”，在展开的数据库用户属性对话框中进行设置

- 通过 Transact-SQL 设置数据库角色

(1) 通过加入数据库角色，赋予数据库用户“dba”权限：`exec sp_addrolemember 'db_owner', 'dba'`

(2) 此时，dba 就可以全权管理数据库 mydb 中的对象了。

(3) 如果想让 SQL Server 登陆帐户“dba”访问多个数据库，比如 mydb2。可以让 sa 执行下面的语句：

```
use mydb2
go
create user dba
for login dba
with default_schema=dbo
go
exec sp_addrolemember 'db_owner', 'dba' go
```

(4) 此时，dba 就可以有两个数据库 mydb, mydb2 的管理权限了！

5. 设置管理对象的权限

- 在对象资源管理器中授权：

(1) 选中需要授权的登录名、用户名或角色名

(2) 右键单击选择“属性”，在弹出的该对象的属性窗口中选择“安全对象”，进行权限分配

- 通过 Transact-SQL 分配权限：

(1) 通过 GRANT 和 REVOKE 语句

10.3 实验任务

1. 创建角色 ProgramerRole，拥有创建表，存储过程，视图权限，拥有对 Salary 表的查询、修改、插入权限
2. 创建一个登录账号 Testlogin
3. 创建对应于这个登录账号的数据库用户 TestUser
4. 将用户 TestUser 添加到 TestRole 角色中

10.4 实验报告要求

1. 写出与上述任务相对应的 SQL 语句。
2. 思考数据库服务器上的角色有哪些？权限类型有哪些？
3. 并记录在实验过程中遇到的问题、解决办法及心得体会。

实验八 存储过程实验

11.1 实验目的

1. 掌握用户存储过程的创建操作。
2. 掌握用户存储过程的执行操作。
3. 掌握用户存储过程的删除操作。

11.2 实验内容

1. 创建带输入参数的存储过程的存储过程。
2. 执行所创建的存储过程。
3. 删除所有新创建的存储过程。

11.3 实验任务

对应于“S+学号”数据库

- (1) 创建一个无参存储过程 `StuScoreInfo`，查询以下信息：班级，学号，姓名，性别，课程名称，考试成绩。
- (2) 创建一个带参数的存储过程 `stu_info`，该存储过程根据传入的学生编号在 `Student` 表中查询此学生的信息。
- (3) 创建一个带参数的存储过程 `StuScoreInfo2`，该存储过程根据传入的学生编号和课程名称查询以下信息：班级，学号，姓名，性别，课程名称，考试成绩。
- (4) 编写存储过程，统计离散数学的成绩分布情况，即按照各分数段统计人数。
- (5) 编写带参数的存储过程，根据传入的课程名称统计该课程的平均成绩。

11.4 实验报告要求

1. 写出与上述任务相对应的 SQL 语句
2. 并记录在实验过程中遇到的问题、解决办法及心得体会。

实验九 触发器实验*

12.1 实验目的

1. 掌握触发器的创建、修改和删除操作
2. 掌握触发器的触发执行
3. 掌握触发器与约束的不同

12.2 实验内容

1. 创建触发器

(1) 启动 SQL Server 查询编辑器，“S+学生学号”数据库。

(2) 在查询命令窗口中输入以下 CREATE TRIGGER 语句，创建触发器。

为 SC(学生选课)表创建一个基于 UPDATE 操作和 DELETE 操作的复合型触发器，当修改了该表中的成绩信息或者删除了成绩记录时，触发器被激活生效，显示相关的操作信息。

--创建触发器

```
CREATE TRIGGER tri_UPDATE_DELETE_sc
ON SC
FOR UPDATE, DELETE
AS
```

--检测成绩列表是否被更新

```
IF UPDATE(成绩)
BEGIN
```

--显示学号、课程号、原成绩和新成绩信息

```
SELECT INSERTED.课程号, DELETED.成绩 AS 原成绩,
INSERTED.成绩 AS 新成绩
FROM DELETED , INSERTED
WHERE DELETED.学号=INSERTED.学号
END
```

--检测是更新还是删除操作

```
ELSE IF COLUMNS_UPDATED()=0
BEGIN
```

--显示被删除的学号、课程号和成绩信号

```
SELECT 被删除的学号=DELETED.学号, DELETED.课程号,
DELETED.成绩 AS 原成绩
FROM DELETED
END
ELSE
```

--返回提示信息

```
PRINT '更新了非成绩列!'
```

(3) 点击快捷工具栏上的快捷按钮，完成触发器的创建。

2. 触发触发器

(1) 在查询命令窗口中输入以下 UPDATE SC 语句，修改成绩列，激发触发器。

```
UPDATE SC
SET 成绩=成绩+5
WHERE 课程号='101'
```

(2) 在查询命令窗口中输入以下 UPDATE SC 语句修改非成绩列，激发触发器。

```
UPDATE SC
SET 课程号='113'
WHERE 课程号='103'
```

(3) 在查询命令窗口中输入以下 DELETE SC 语句，删除成绩记录，激发触发器。

```
DELETE SC
WHERE 课程号='102'
```

3. 比较约束与触发器的不同作用期

(1) 在查询命令窗口中输入并执行以下 ALTER TABLE 语句，为 SC 表添加一个约束，使得成绩只能大于等于 0 且小于等于 100。

```
ALTER TABLE SC
ADD CONSTRAINT CK_成绩
CHECK(成绩>=0 AND 成绩<=100)
```

(2) 在查询命令窗口中输入并执行以下 UPDATE SC 语句，查看执行结果。

```
UPDATE SC
SET 成绩=120
WHERE 课程号='108'
```

(3) 在查询命令窗口中输入执行以下 UPDATE SC 语句，查看执行结果。

```
UPDATE SC
SET 成绩=90
WHERE 课程号='108'
```

4. 删除新创建的触发器

(1) 在查询命令窗口中输入 DROP TRIGGER 语句，删除新创建的触发器。

```
DROP TRIGGER tri_UPDATE_DELETE_sc
```

(2) 点击快捷工具栏上的快捷按钮，删除触发器。

12.3 实验任务

1. 在 Student 表中编写 insert 的触发器，假如每个班的学生不能超过 30 个，如果低于此数，添加可以完成；如果超过此数，则插入将不能实现。
2. 在 SC 表上编写 update 触发器，当修改 SC 表中的 Grade 字段时将其修改前后的信息保存在 SC_log 表中。

12.4 实验报告要求

1. 写出与上述任务相对应的 SQL 语句。
2. 触发器与约束有什么不同？

实验十 数据库设计综合应用

13.1 实验目的

1. 掌握数据库设计和实现的基本过程
2. 掌握数据库模式设计、分析和实现的方法
3. 理解数据库应用系统软件开发的一般过程
4. 掌握数据库设计工具软件 Power Designer 的使用

13.2 实验内容

要求学生根据周围的实际情况，自选一个小型的数据库应用项目，并深入到应用项目的现实世界中，进行系统分析和数据库设计。可以选择给定的五个叙述性题目或附录 1 的选题表中选择一个课题。学生 3 人一组，成员自由组合。提交课程设计报告，内容包括：

1. 课程设计说明（含设计题目、设计任务与要求、开发环境说明）
2. 功能需求分析
3. 数据库概念设计
4. 数据库逻辑设计
5. 应用系统功能模块
6. 源程序代码及运行结果（含运行界面屏幕截图）。
7. 总结

13.3 实验任务

主要完成数据库设计的六个阶段中的前四阶段：需求分析(分析用户要求)、概念设计(信息分析和定义)、逻辑设计(设计实现)和物理设计(物理数据库设计)进行。概念模型设计时采用自底向上的方法，即自顶向下地进行需求分析，然后再自底向上地设计概念结构。

1. 系统分析与数据库设计阶段

- (1) 通过社会调查，选择一个实际应用数据库系统的课题。
- (2) 进行系统需求分析和系统设计，写出系统分析和设计报告。
- (3) 设计数据模型并进行优化，确定数据库结构、功能结构和系统安全性和完整性要求。

2. 应用程序设计阶段

- (1) 完成数据库定义工作，实现系统数据的数据处理和数据录入。
- (2) 实现应用程序的设计、编程、优化功能，实现数据安全性、数据完整性和并发控制技术等功能，并针对具体课题问题提出解决方法。

3. 系统集成调试阶段

对系统的各个应用程序进行集成和调试，进一步优化系统性能，改善系统用户界面。

13.4 实验报告要求

1. 完成问题陈述中所提到的所有需求功能；

2. 要求撰写不少于 3000 个文字的文档，设计报告按照以下提纲书写：
 - (1) 概述（包括项目背景、编写目的、软件定义、开发环境等内容）
 - (2) 需求分析（问题陈述、需完成的功能。DD、DFD）
 - (3) 数据库概念结构设计（画出 E-R 模型图）
 - (4) 数据库逻辑结构设计（把 E-R 模型图转换为关系表。描述每一个基本表关系。定义视图、定义索引、主关键字、定义权限）
 - (5) 数据流图及程序结构框图
 - (6) 程序原代码及其说明
 - (7) 总结
 - (8) 参考文献
3. 主要内容及装订顺序：封面(统一格式)、目录、正文、参考文献；
4. 设计报告严禁抄袭，即使是同一小组也不允许雷同，否则按不及格论。

13.5 参考题目

请在给定的五个叙述性题目或附录 1 中任选一个题目完成本课程的设计任务。

1. 中小学智能排课系统

中小学校教务处作为中小学校所有教学事务的管理中心，是一所学校教学计划能够顺利进行的指挥和协调部门。其中排课室工作人员有限，排课的任务重，而引起课程变化的因素多，工作量大，迫切需要一套完善的计算机自动排课系统，以提高排课效率。

为此，要求软件主要完成的功能有：能根据教师要求（如某天不得排课）、课程约束（如体育不能排在上午第一节课）、班级约束（如某班星期五下午最后一节课不排课）、校级约束（如全校所有班级星期一下午第一节课都为班会）等信息自动为班级和教师生成课程表，要求主课尽量排在上午和下午一、二节课，副课尽量排在上午和下午的最后一节课，如体育课排在上午第一节课是不太合适的。对于软件不能安排的少数课程，教务工作者能够在自动排出的课程表上进行手工调课。

要求：

- (1) 系统可以进行两节连课处理，如作文课可以连课上；
- (2) 排出的课程表中不允许有教师冲突的情况，比如，一个教师同时给两个班级上课是不允许的；
- (3) 要求课程表中的课程要有所变化，比如一个班级的所有数学课总是排在上午第一节课是不好的课程表。
- (4) 每周上课天数可以是 5 天也可以是六天，每天上课节数可以是 7 节或是 8 节；
- (5) 每个年级所开课程是一样的，而且所开课程可以变化；
- (6) 一个教师可以教授多门课程；
- (7) 系统可以为每个班级和每位教师打印课程表；

(8) 系统可以把生成的课程表自动转化成网页在网上公布;

(9) 在课表生效后, 教师可以要求调课;

(10) 教师数量是动态的, 所开课程的数量也是动态的。

2. 学生学籍管理信息系统

学生学籍管理工作包括学生档案、学籍、成绩、升学等内容的管理; 面对大量的学生数据、报表, 手工处理方式已经很难跟上现代化的步伐。随着计算机技术及网络通讯技术的飞速发展, 许多学校已经有了较好的计算机应用甚至网络硬件建设基础。因此为提高学校管理工作的现代化、科学化水平, 保证信息处理的即时化、准确化, 开发一套对学生学籍进行管理的软件是极其重要的。

应完成的主要功能:

(1) 学生档案的管理, 即录入、修改、查询、输出学生档案信息, 这些信息包括学生基本情况、学生简历情况、学生奖励情况、学生处分情况、学生家庭信息、学生体检情况。

(2) 学生学籍管理, 录入、修改、查询、输出学生学籍信息, 这些信息包括学生奖贷学金情况、学生注册、学生异动情况、学生军训情况、学生毕业情况。

(3) 学生成绩管理, 录入修改、查询、输出学生入校成绩, 各学期、各门课程的成绩信息, 并支持按年级、班级等条件的统计、查询、报表输出。

3. 人事管理系统

系统功能的基本要求:

(1) 员工各种信息的输入, 包括员工的基本信息、学历信息、婚姻状况信息、职称等。

(2) 员工各种信息的修改;

(3) 对于转出、辞职、辞退、退休员工信息的删除;

(4) 按照一定的条件, 查询、统计符合条件的员工信息; 至少应该包括每个员工详细信息的查询、按婚姻状况查询、按学历查询、按工作岗位查询等, 至少应该包括按学历、婚姻状况、岗位、参加工作时间等统计各自的员工信息;

(5) 对查询、统计的结果打印输出。

数据库要求: 在数据库中至少应该包含下列数据表:

(1) 员工基本信息表;

(2) 员工婚姻情况表, 反映员工的配偶信息;

(3) 员工学历信息表, 反映员工的学历、专业、毕业时间、学校、外语情况等;

(4) 企业工作岗位表;

(5) 企业部门信息表。

4. 工资管理系统

系统功能的基本要求:

- (1) 员工每个工种基本工资的设置
- (2) 加班津贴管理, 根据加班时间和类型给予不同的加班津贴;
- (3) 按照不同工种的基本工资情况、员工的考勤情况产生员工的每月的月工资;
- (4) 员工年终奖金的生成, 员工的年终奖金计算公式= (员工本年度的工资总和+津贴的总和)/12;
- (5) 企业工资报表。能够查询单个员工的工资情况、每个部门的工资情况、按月的工资统计, 并能够打印;

数据库要求: 在数据库中至少应该包含下列数据表:

- (1) 员工考勤情况表;
- (2) 员工工种情况表, 反映员工的工种、等级, 基本工资等信息;
- (3) 员工津贴信息表, 反映员工的加班时间, 加班类别、加班天数、津贴情况等;
- (4) 员工基本信息表
- (5) 员工月工资表。

5. 高校图书管理系统

主要完成图书管理系统的设计。

用户有以下需求:

- (1) 建立读者档案;
- (2) 建立图书档案, 建立书目索引;
- (3) 完成日常图书检索、借还工作, 对读者档案、图书档案、借还系统的访问, 必需进行身份验证。

具体要求如下:

- (1) 读者档案数据包括: 借书证号、姓名、性别、出生日期、身份证号、单位、通讯地址、邮政编码、联系电话、办证日期、借阅范围(书库)、允许最多借书册数、借书期限、照片、职业等。
- (2) 图书档案数据包括: 书号、书名、作者、出版单位、出版日期、版次、单价、内容提要、分类号、索书号、藏书册数、每册图书馆藏注册号、所在书库、入库日期等。
- (3) 检索系统能根据书号、书名、作者、出版单位、内容提要关键字、分类号、索书号、每册图书馆藏注册号等进行查询。
- (4) 借书系统在输入借书条后, 能根据借书证号判断该读者可以借书的书库, 借书是否超出最大允许借书册数, 书库中是否还有该书可借。
- (5) 还书系统能对过期未还图书进行罚款, 对归还的图书能从借书登记表中取消, 对丢失的图书进行登记。

附录 1

序号	题目
1	医院管理信息系统
2	银行前台业务处理系统
3	企业进销存管理系统
4	仓库管理系统
5	在线课件管理系统
6	驾驶员模拟考试系统
7	网络求职平台
8	报刊订阅管理系统
9	志愿服务爱心交易商店
10	大学生心理健康监测与辅导网络平台
11	卡拉 OK 点歌系统
12	网上购物系统
13	航空票务管理系统
14	车站售票管理系统
15	在线作业系统
16	网络物流系统
17	题库系统