

# 第一篇 实验指导

## 实验 1 SQL Server 2008 的环境

### 一、实验目的

1. 熟悉 SQL Server 2008 的安装方法与步骤。
2. 熟悉 SQL Server 2008 的实例配置。
3. 熟悉 SQL Server 2008 的服务器配置。
4. 掌握 SQL Server 2008 安装中的“数据库引擎配置”方法。

### 二、实验准备

1. 了解 SQL Server 2008 常用的版本和适用的操作系统平台。
2. 了解 SQL Server 2008 安装的软、硬件要求。
3. 了解 SQL Server 2008 支持的身份验证模式。
4. 了解 SQL Server 2008 的安装规则。

### 三、实验内容及步骤

1. 安装 SQL Server 2008 的准备工作。

安装 SQL Server 2008 时首先要了解所需的必备条件，然后检查计算机的软、硬件配置是否满足 SQL Server 2008 开发安装要求。具体条件如表 1.1 所示。

表 1.1 安装 SQL Server 2008 所需的必备条件

软硬件	描述
软件	Microsoft Windows Installer 4.5 或更高版本以及 Microsoft 数据访问组件(MDAC) 2.8 SP1 或更高版本; Microsoft Windows .NET Framework 3.5; Microsoft SQL Server Native Client
处理器	Intel Pentium III 相容或更高性能的处理器; 1.4GHz 处理器, 建议使用 2.0GHz 或更高性能的处理器
内存 (RAM)	Enterprise、Developer、Workgroup 及 Standard Editions: 512 MB (1GB 或者更高) Express Edition: 192 MB (512MB 或者更高)
可用硬盘	至少 2.0GB 的可用磁盘空间; 其中, 数据库组件: 至少 280MB; Analysis Services (分析服务): 至少 90 MB; Reporting Services (报表服务): 至少 120MB; 客户端组件: 850MB
CD-ROM 或 DVD-ROM 驱动器	从磁盘进行安装时需要相应的 CD 或 DVD 驱动器
显示器	SQL Server 2008 图形工具需要使用 VGA 或更高分辨率; 分辨率至少为 1024*768 像素

## 2. 安装 SQL Server 2008 的操作。

在 Windows 7 下安装 SQL Server 2008 (Enterprise Edition, 企业版) 的操作步骤如下:

(1) 将 SQL Server 2008 安装盘放入光驱, 在弹出的“SQL Server 安装中心”对话框左侧, 单击“安装”选项, 再单击“全新 SQL Server 2008 独立安装或向现有安装添加功能”选项, 启动 SQL Server 2008 安装。

(2) 打开“安装程序支持规则”对话框, 系统进行安装程序支持规则检查, 以确定安装 SQL Server 2008 安装程序支持文件时可能发生的問題。必须更正所有的失败, 单击“确定”按钮安装程序才能继续运行。

(3) 在打开的“产品密钥”对话框中输入产品密钥。

(4) 单击“下一步”按钮, 进入“许可条款”对话框, 选中“我接受许可条款”单选按钮。

(5) 单击“下一步”按钮, 进入“安装程序支持文件”对话框, 单击“安装”按钮, 安装程序支持文件。

(6) 安装完程序支持文件后, 对话框上会出现“下一步”按钮, 单击“下一步”按钮, 进入“安装程序支持规则”对话框, 在此对话框中, 如果所有规则都通过, 则“下一步”按钮可用。

(7) 单击“下一步”按钮, 进入“功能选择”对话框, 这里可以选择要安装的功能, 如果全部安装, 则可以单击“全选”按钮进行安装。

(8) 单击“下一步”按钮, 进入“实例配置”对话框, 在该对话框选择实例的命名方式并命名实例, 然后选择实例根目录。

(9) 单击“下一步”按钮, 进入“磁盘空间要求”对话框, 对话框中显示安装 SQL Server 2008 所需的磁盘空间。

(10) 单击“下一步”按钮, 进入“服务器配置”对话框, 在该对话框中, 单击“对所有 SQL Server 服务使用相同的账户”按钮, 以便为所有的 SQL Server 服务设置统一账户。

(11) 单击“下一步”按钮, 进入“数据库引擎配置”对话框, 在该对话框中选择身份验证模式, 并输入密码。然后单击“添加当前用户”按钮。

(12) 单击“下一步”按钮, 进入“Analysis Services 配置”对话框, 在该对话框中单击“添加当前用户”按钮。

(13) 单击“下一步”按钮, 进入“Reporting Services 配置”对话框, 在该对话框中选择“本机模式默认配置”单选按钮。

(14) 单击“下一步”按钮, 进入“错误和使用情况报告”对话框, 在该对话框中设置是否将错误和使用报告发送到 Microsoft, 通常选择默认设置。

(15) 单击“下一步”按钮, 进入“安装规则”对话框, 在该对话框中如果所有规则都通过, 则“下一步”按钮可用。

(16) 单击“下一步”按钮, 进入“准备安装”对话框。该对话框中显示准备安装的 SQL Server 2008 功能。

(17) 单击“安装”按钮, 进入“安装进度”对话框, 在该对话框中显示 SQL Server 2008 的安装进度。

(18) 单击“下一步”按钮, 进入“完成”对话框, 单击“关闭”按钮, 即可完成 SQL Server 2008 的安装。

3. 连接 SQL Server 2008 服务器。

提示：参看教材的 1.5.3 节。

4. 查看本机登录的身份验证模式。

提示：在 SSMS 中选择登录的服务器并右击，在弹出的快捷菜单中选择“连接”命令，即可查看。

#### 四、实验思考

1. 在 Windows 7 操作系统上，可以安装 SQL Server 2008 的哪些版本？

2. 安装 SQL Server 2008 进入“实例配置”对话框时，在该对话框选择实例的命名方式并命名实例，然后选择实例根目录。请问什么是实例？默认情况下，实例根目录路径是什么？

3. 在 SQL Server 2008 中，身份验证的模式有几种？分别是什么？如何设置？

4. SQL Server 2008 的“服务器配置”是指什么？是否可以为所有的 SQL Server 服务分配相同的登录账户？

5. 什么是数据库引擎？数据库引擎组件是指什么？数据库引擎组件的作用是什么？

## 实验 2 SQL Server 2008 管理工具的使用

### 一、实验目的

1. 熟悉 SQL Server Management Studio 主窗口的组成及基本操作。
2. 熟悉对象资源管理器窗口界面的组成。
3. 熟悉模板资源管理器的基本操作。
4. 熟悉查询设计器窗口的使用方法。
5. 掌握 SQL Server 配置管理器基本操作。

### 二、实验准备

1. 了解 SQL Server 2008 的常用管理工具程序及其功能。

(1) 已注册服务器：启动、停止、暂停 SQL Server 服务。在对 SQL Server 中的数据库和表进行任何操作之前需要首先启动 SQL Server 服务。

(2) 对象资源管理器：有助于用户对 SQL Server 数据库进行管理和操作。

(3) 查询设计器：帮助用户调试 T-SQL 程序、测试查询及管理数据库。

(4) 联机丛书：是在使用 SQL Server 时可以随时参考的帮助说明。

2. 了解 SQL Server 配置管理器的基本功能及操作。

### 三、实验内容及步骤

1. 使用已注册的服务器

(1) 创建服务器组。通过使用 SSMS 中“已注册的服务器”页面创建服务器组，并将服务器放置在服务器组中来管理和组织服务器。可以随时在已注册的服务器中创建服务器。

创建服务器组的操作步骤如下:

1) 启动“SQL Server Management Studio”, 在菜单中选择“视图”→“已注册的服务器”命令, 弹出“已注册的服务器”对话框, 右击“数据库引擎”节点下的“Local Server Groups”, 在弹出的快捷菜单中选择“新建服务器组”命令, 如图 2.1 所示。

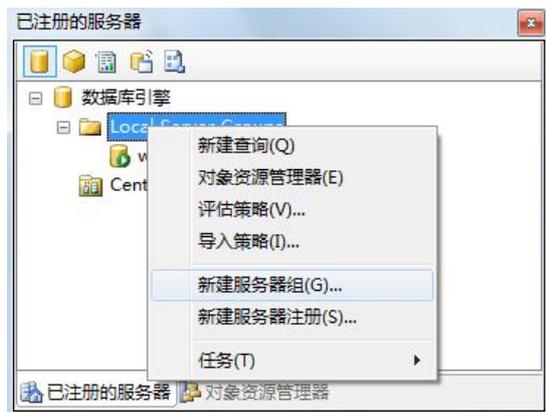


图 2.1 “新建服务器组”命令

2) 在弹出的“新建服务器组属性”对话框(图 2.2)中的“组名”文本框中输入要创建服务器组的名称(sql server groups), 在“组说明”文本框中输入关于这个服务器组的简要说明, 如图 2.2 所示。



图 2.2 “新建服务器组属性”对话框

3) 信息输入完毕, 单击“确定”按钮, 即可完成服务器组的创建。

(2) 注册新的服务器。用户管理服务器中的数据库, 需要注册方可使用。服务器既可以是网络服务器也可以是本地服务器, 若是本地服务器则在安装完成后, 自动完成注册。

注册新的服务器操作步骤如下:

1) 参照创建服务器组的步骤 1), 在图 2.1 所示的快捷菜单中选择“新建服务器注册”命令。

2) 在弹出的“新建服务器注册”对话框中, 单击“常规”选项卡。在“服务器名称”文本框中输入要注册的服务器名称, 如图 2.3 所示。在“连接属性”选项卡中可以指定要连接到的数据库名称和使用的网络协议等其他信息。

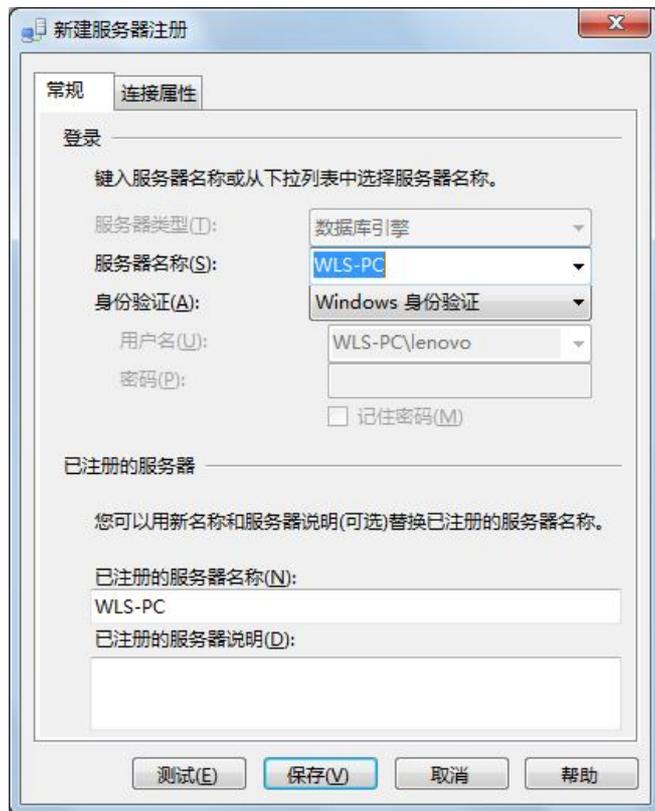


图 2.3 “新建服务器注册”对话框

3) 设置完成后单击“测试”按钮，测试连接，若成功则单击“保存”按钮，完成新建服务器注册的设置。此时，在“已注册的服务器”对话框中就可以看到刚才所注册的服务器的图标。

## 2. 使用对象资源管理器

(1) 了解系统数据库有几个，它们分别叫什么名字。

(2) 认识数据库对象。选择系统数据库“master”，观察 SQL Server 2008 对象资源管理器中数据库对象的组织方式，并在下列横线处填上适当的内容：

1) 表、视图在\_\_\_\_\_文件夹下。

2) “可编程性”文件夹下有\_\_\_\_\_对象。

(3) 体验不同数据库对象的操作方法。

1) 展开系统数据库“master”，展开“表”→“系统表”，右击“dbo.spt\_values”，查看系统弹出的快捷菜单有哪些内容。

2) 展开表“dbo.spt\_values”下的“列”文件夹，查看此表有哪些列，即认识表的结构。

## 3. 使用模板资源管理器

操作步骤如下：

(1) 在 SSMS 主窗口的菜单中选择“视图”→“模板资源管理器”命令，弹出“模板资源管理器”窗格。

(2) 在“模板资源管理器”窗格，按下 F1 键寻求帮助，学习和认识 SQL Server 2008 提供的模板。

(3) 在“模板资源管理器”窗格中找到“Database”，双击“Create Database”，查看 CREATE DATABASE 语句的结构，如图 2.4 所示。

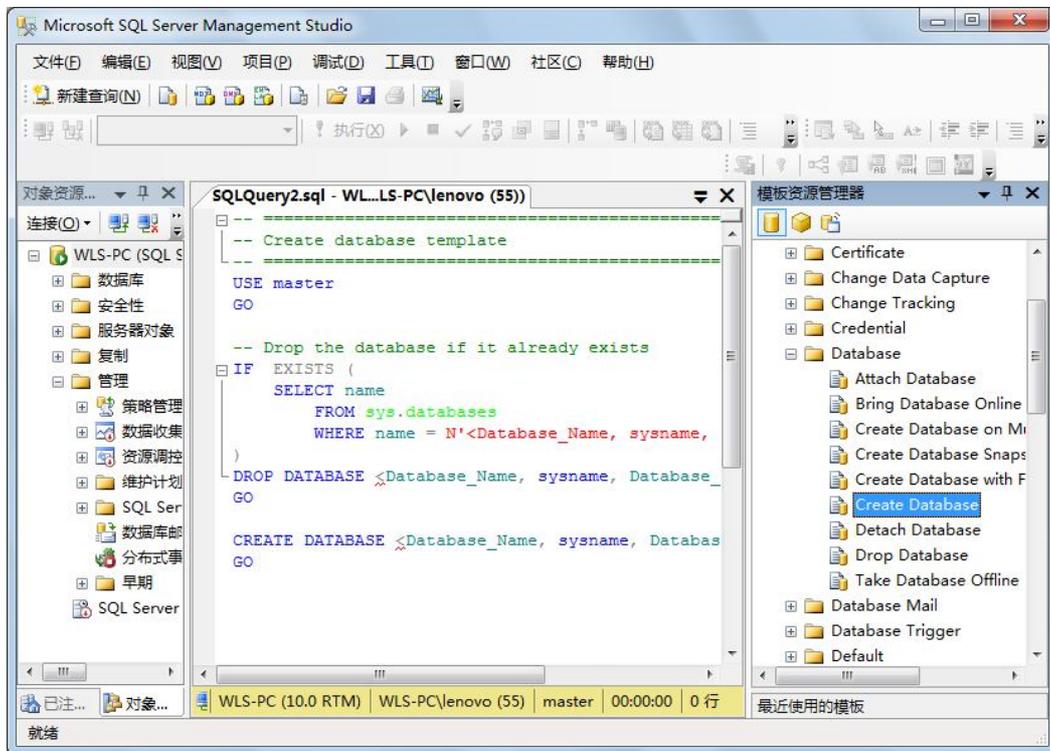


图 2.4 查看 CREATE DATABASE 语句的结构

#### 4. 使用查询设计器

(1) 在 SSMS 主窗口中单击“新建查询”按钮，观察主窗口有什么变化？

(2) 在“查询设计器”的文本窗口中输入查询语句：

```
select * from dbo.spt_values
```

(3) 执行“查询”→“执行”菜单命令或单击工具栏上的相应按钮执行查询，注意观察输出结果。然后关闭“SQL 查询设计器”窗口。

注意：执行查询语句之前，可以先执行“查询”→“分析”菜单命令，分析 SQL 代码的语句。

#### 四、实验思考

1. 什么是“对象资源管理器”？它的作用是什么？什么是查询设计器？查询设计器的作用是什么？

2. 使用什么方法可以改变查询设计器中的当前数据库？

3. 什么是 SQL Server 配置管理器？如何使用 SQL Server 配置管理器管理 SQL Server 2008 服务？

4. 在 SQL Server 2008 中，如何将 Windows 身份验证改为混合模式身份验证？

5. SQL Server Profiler 的中文意思是什么？它的作用是什么？如何使用？

## 实验 3 数据库的创建、修改和删除

### 一、实验目的

1. 了解 SQL Server 数据库的构成和数据库对象。
2. 了解数据库的物理存储。
3. 掌握在对象资源管理器中创建、修改和删除数据库的操作方法。
4. 学会使用 T-SQL 的 CREATE DATABASE、ALTER DATABASE 和 DROP DATABASE 命令创建、修改和删除数据库。

### 二、实验准备

1. 了解和明确创建、修改和删除数据库的权限范围。
2. 确定要创建的数据库的名称、所有者、数据库容量和存放数据文件的位置。
3. 了解要修改的数据库名称、所有者、存放数据文件的位置、数据库容量的基本要求和方法。
4. 了解创建、修改和删除数据库的常用方法。

### 三、实验内容及步骤

#### 1. 在对象资源管理器中创建、修改和删除数据库

(1) 不指定文件创建数据库。要求是：创建的数据库名为 MYTEST。创建完成后，在对象资源管理器中右击 MYTEST 数据库并选择“属性”命令，查看 MYTEST 数据库的属性，并在下列横线上按要求填入相关数据。

- 1) 主数据文件的逻辑文件名：\_\_\_\_\_。
- 2) 物理文件名：\_\_\_\_\_。
- 3) 存放的位置：\_\_\_\_\_。
- 4) 初始大小：\_\_\_\_\_。
- 5) 最大文件大小：\_\_\_\_\_。
- 6) 事务日志文件的逻辑文件名：\_\_\_\_\_。
- 7) 物理文件名：\_\_\_\_\_。
- 8) 存放的位置：\_\_\_\_\_。
- 9) 文件增长情况：\_\_\_\_\_。

**注意：**由于不指定文件，所以创建的数据库 MYTEST 的主数据文件大小为系统数据库 MODEL 的主数据文件大小。事务日志文件的大小为 MODEL 数据库事务日志文件的大小。而文件的最大大小可以增长到填满所有可用的磁盘空间为止。

(2) 创建指定数据文件和事务日志文件的数据库。要求是：创建学生信息 Student 数据库。主数据文件的物理文件名为 Student.mdf，存放的位置为 D:\Mydb，初始大小为 5MB，最大大小为 25MB，自动增长量为 5MB；事务日志文件的物理文件名为 Student\_log.ldf，存放的

位置为 D:\Mydb, 初始大小为 5MB, 最大大小为 20MB, 自动增长量为 5MB。

**注意:** 在创建 Student 数据库前必须先确定 D:盘上是否已建立 Mydb 文件夹, 若没有则创建该文件夹。

(3) 对 (2) 中创建好的 Student 数据库进行以下修改: 修改主数据文件的最大大小为无限大, 自动增长量为 10%; 修改事务日志文件的最大大小为 30MB。

(4) 删除数据库 MYTEST。

## 2. 在查询设计器中创建、修改和删除数据库

(1) 使用 CREATE DATABASE 命令创建简单的数据库。要求先阅读下述语句, 然后上机完成操作:

```
/*创建简单数据库的语句*/
CREATE DATABASE Products
ON
(NAME = prods_dat,
 FILENAME = 'D:\program files\microsoft SQL Server\mssql\data\prods.mdf',
 SIZE = 4,
 MAXSIZE = 10,
 FILEGROWTH = 1)
```

完成上机操作后, 在下列横线上填入相关数据。

- 1) 创建的数据库名: \_\_\_\_\_。
- 2) 主数据文件的逻辑文件名: \_\_\_\_\_。
- 3) 主数据文件的最大大小: \_\_\_\_\_。
- 4) 事务日志文件的逻辑文件名: \_\_\_\_\_。

**注意:** 在执行创建 Products 数据库的命令前, 必须先确定 D:盘上是否有路径:\program files\microsoft SQL Server\mssql\data, 若没有则改变或建立该路径。

(2) 使用 CREATE DATABASE 命令指定多个数据文件和事务日志文件创建数据库。要求先阅读下列语句, 在 “/\* \*/” 中的横线处填入相应的注释。

```
/*指定多个数据文件和事务日志文件创建数据库*/
CREATE DATABASE Archive
ON
/* 创建_____文件_____*/
PRIMARY
(NAME=Arch1,
 FILENAME='D:\Mydb\Archdat1.mdf',
 SIZE=5MB,
 MAXSIZE=30,
 FILEGROWTH=2),
/* 创建次数据文件 Arch2 */
(NAME=Arch2,
 FILENAME='D:\Mydb\Archdat2.ndf',
 SIZE=5MB,
 MAXSIZE=30,
 FILEGROWTH=2),
/*创建次数据文件_____*/
(NAME=Arch3,
```

```

        FILENAME='D:\Mydb\Archdat3.ndf',
        SIZE=5MB,
        MAXSIZE=30,
        FILEGROWTH=2)
LOG ON
/* _____ */
( NAME=Archlog1,
  FILENAME='D:\Mydb\Archlog1.ldf',
  SIZE=5MB,
  MAXSIZE=20,
  FILEGROWTH=2),
/* _____ */
( NAME=Archlog2,
  FILENAME='D:\Mydb\Archlog2.ldf',
  SIZE=5MB,
  MAXSIZE=20,
  FILEGROWTH=2)

```

完成上机操作后，在下列横线上填入相关数据。

语句中使用了\_\_\_\_\_个\_\_\_\_\_MB 数据文件，使用了\_\_\_\_\_个\_\_\_\_\_MB 事务日志文件。

(3) 使用 CREATE DATABASE 命令和文件组创建人事信息管理 Rsxxgl\_db 数据库。要求阅读下列语句，并在“/\* \*/”中的横线处填入相应的注释。

```

/*使用文件组创建人事信息管理 Rsxxgl_db 数据库的语句*/
CREATE DATABASE Rsxxgl_db
ON
/* _____ */
PRIMARY
( NAME=Rsx1_dat,
  FILENAME='D:\Mydb\Rsx1dat.mdf',
  SIZE=3,
  MAXSIZE=20,
  FILEGROWTH=5% ),
( NAME=Rsx2_dat,
  FILENAME='D:\Mydb\Rsx2dat.ndf',
  SIZE=3,
  MAXSIZE=20,
  FILEGROWTH=5% ),
/* _____ */
FILEGROUP RsxxglGroup1
( NAME=RGp1Fil_dat,
  FILENAME='D:\Mydb\RG1Fildt.ndf',
  SIZE=3,
  MAXSIZE=20,
  FILEGROWTH=5),
( NAME=RGp1Fi2_dat,
  FILENAME='D:\Mydb\RG1Fi2dt.ndf',

```

```

        SIZE=3,
        MAXSIZE=20,
        FILEGROWTH=5),
/* _____ */
FILEGROUP RsxxglGroup2
(NAME=RGrp2Fil_dat,
 FILENAME='D:\Mydb\RG2Fil.dtf',
 SIZE=3,
 MAXSIZE=20,
 FILEGROWTH=5),
(NAME=RGrp2Fi2_dat,
 FILENAME='D:\Mydb\RG2Fi2.dtf',
 SIZE=3,
 MAXSIZE=20,
 FILEGROWTH=5)
/* _____ */
LOG ON
(NAME='Rsxxgl_log',
 FILENAME='D:\Mydb\Rsxxgllog.ldf',
 SIZE=1MB,
 MAXSIZE=25MB,
 FILEGROWTH=3MB)

```

完成上机操作后，在下列横线上填入相关数据。

语句中包含 \_\_\_\_\_ 个文件组，其中主文件组包含文件 \_\_\_\_\_ 和 \_\_\_\_\_。这些文件的 FILEGROWTH 增量为 \_\_\_\_\_。

(4) 使用 ALTER DATABASE 命令修改数据库 Archive。

1) 下述语句的功能是将数据库 Archive 的主数据文件的逻辑文件名 Arch1 修改为 Arch1\_main。要求在横线处填入正确的内容，完成操作。

```

ALTER DATABASE Archive
MODIFY FILE _____

```

注意：修改数据文件的逻辑文件名的语法格式为：

```

MODIFY FILE (NAME = logical_file_name, NEWNAME = new_logical_name...)

```

2) 下列语句的功能是将数据库 Archive 的主数据文件中的最大大小修改为 35，文件自动增长量修改为 5。要求在下列横线处填入正确的内容，完成操作。

```

ALTER DATABASE Archive
MODIFY FILE
(NAME= _____ ,
 MAXSIZE =35,
 _____ )

```

注意：当修改数据库的容量时，容量的大小必须比文件当前容量的大小要大。若要修改数据库文件的属性，每次只能更改这些属性中的一种。

3) 下列语句的功能是将数据库 Archive 中物理文件名为 Arch3 的文件删除。要求在横线处填入正确的内容，完成操作。

```

ALTER DATABASE Archive _____

```

(5) 使用 DROP DATABASE 命令一次删除 Archive 和 Rsxxgl\_db 两个数据库。要求在横

线处填入正确的语句。

**注意：**删除数据库与删除数据库文件的区别。

(6) 请在横线处填入合适的内容，使得下列语句可以将 Sales 数据库名改为 NewSales。

```
ALTER _____ Sales _____ NAME=NewSales
```

**提示：**重命名数据库的语法格式如下：

```
ALTER DATABASE database MODIFY NAME = new_dbname
```

(7) 将数据库 Products 中的数据文件 Prods\_dat 的文件大小增加到 10MB。

```
ALTER DATABASE Products
      _____ FILE
      ( NAME = Prods_dat,
        SIZE =10MB )
```

#### 四、实验思考

1. 在对象资源管理器中对数据库更名有什么要求？是否能对创建好的数据文件或事务日志文件更名？
2. SQL Server 2008 服务器正在运行，当用户将已经创建好的数据库 Test1 在对象资源管理器中删除时，系统提示不能进行删除操作，这是什么原因？正确的操作是什么？
3. 在实验内容 2（4）的 3）中，若数据库 Archive 的物理文件 Arch3 有信息，是否能删除？
4. 分析在对象资源管理器和查询设计器中创建数据库有什么异同？

## 实验 4 数据库的分离、附加、收缩

### 一、实验目的

1. 掌握数据库分离和附加的基本概念。
2. 掌握数据库分离和附加的基本操作方法。
3. 掌握设置 SQL Server 数据库收缩的操作方法。

### 二、实验准备

1. 了解数据库分离和附加的基本概念，明确分离数据库的目的。
2. 了解数据库附加的基本操作方法。
3. 了解 SQL Server 数据库收缩。

### 三、实验内容及步骤

1. 复制在实验 3 中创建的学生信息数据库 Student 文件。

**提示：**停止运行 SQL Server 服务器，找到数据库 Student 存放的位置 D:\Mydb，并选定 Student.mdf 和 Student\_log.ldf 两个文件进行复制，然后粘贴至目的位置。

2. 将在实验 3 中创建的学生信息数据库 Student 移动至 E:\Mytest 目录下。

根据题意分析, 可选择数据库分离和附加的方法实现。

提示: 将 Student.mdf 和 Student\_log.ldf 两个文件复制并粘贴至 E:\Mytest 下 (参考实验内容 1 的操作提示), 然后启动 SQL Server 服务器。右击“数据库”文件夹并在弹出的快捷菜单中选择“所有任务”→“附加数据库”命令, 在弹出的“附加数据库”窗口中单击“添加”按钮, 指定要附加数据库的 Student.mdf 文件, 如图 4.1 所示。单击“确定”按钮执行附加操作。

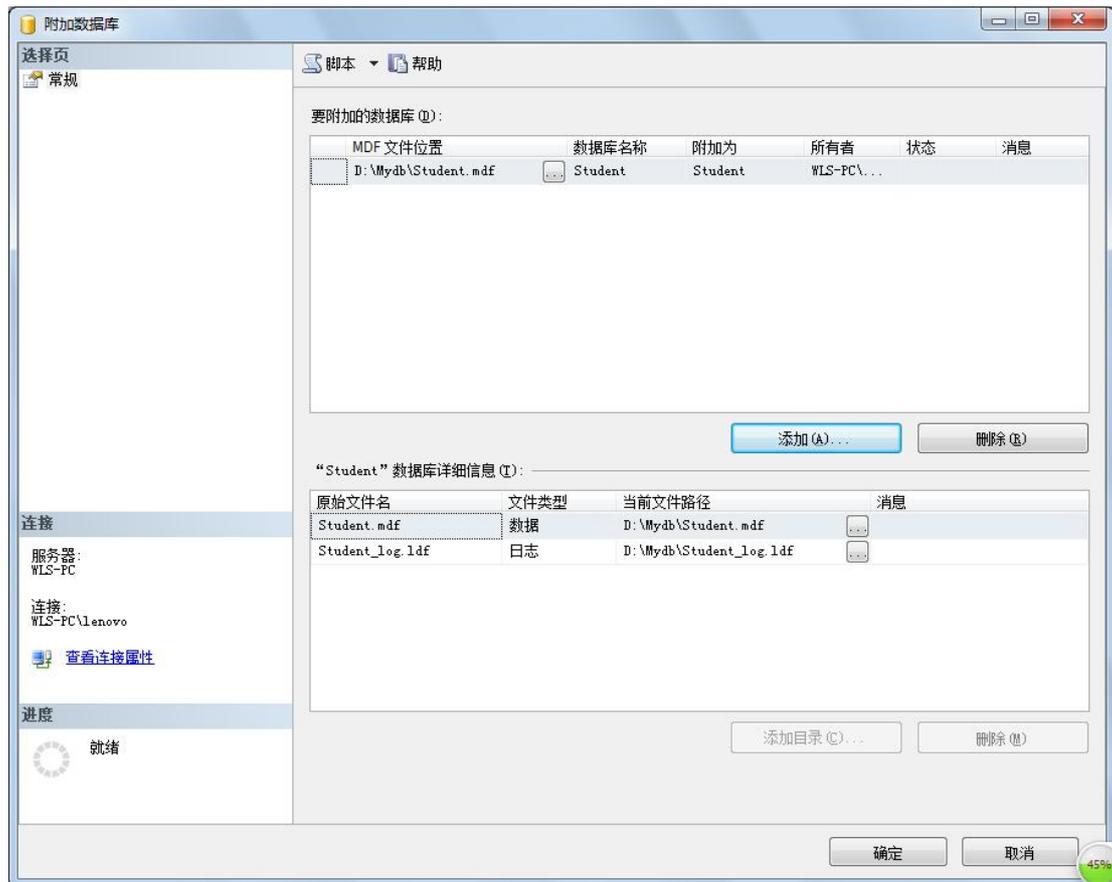


图 4.1 “附加数据库”窗口

3. 删除实验 3 中创建的数据库 Rsxxgl\_db, 再检查与数据库相对应的操作系统文件是否还存在。

4. 将 Student\_db 数据库设置为自动收缩方式。

提示: 为了观察数据库收缩前后的变化, 收缩前打开数据库的属性, 查看一下文件大小情况, 收缩后再查看一下文件大小情况。

5. 将 Student\_db 数据库的日志文件 stu\_info.ldf 设置为手动收缩。

#### 四、实验思考

1. 当用户对已经创建好的数据库进行分离后, 会发现在“对象资源管理器”中该数据库的名称已经不存在了, 此时是否可以认为该数据库不存在了呢? 为什么?

2. 当 SQL Server 服务器正在运行时，用户需要将自己创建好的数据库复制带走，应该如何操作？
3. 在什么情况下要进行数据库分离？为什么？什么情况下要进行收缩数据库？其目的是什么？

## 实验 5 数据表的创建和管理

### 一、实验目的

1. 了解 SQL Server 数据表的构成。
2. 掌握在对象资源管理器中创建、修改和删除数据表的操作方法。
3. 学会使用 T-SQL 创建、修改和删除数据表。
4. 掌握在对象资源管理器中创建、修改和删除记录的操作方法。
5. 学会使用 T-SQL 添加、修改和删除记录。

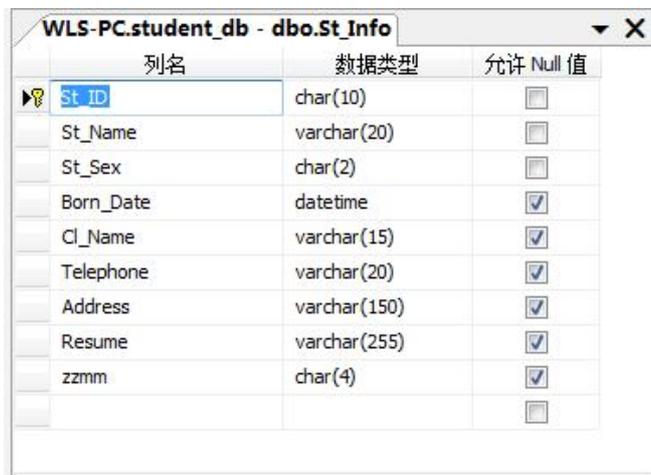
### 二、实验准备

1. 了解创建、修改和删除数据表的常用方法。
2. 根据实际需要定义好要创建的数据表中列的数据类型。了解哪些列允许空值，是否要使用以及何时使用约束、默认设置或规则。
3. 了解添加、修改和删除记录的常用方法。

### 三、实验内容及步骤

1. 在对象资源管理器中创建数据表。

(1) 向学生数据库 Student\_db 中添加学生信息 St\_Info 表，表结构和表记录分别如图 5.1 和图 5.2 所示。



列名	数据类型	允许 Null 值
St_ID	char(10)	<input type="checkbox"/>
St_Name	varchar(20)	<input type="checkbox"/>
St_Sex	char(2)	<input type="checkbox"/>
Born_Date	datetime	<input checked="" type="checkbox"/>
Cl_Name	varchar(15)	<input checked="" type="checkbox"/>
Telephone	varchar(20)	<input checked="" type="checkbox"/>
Address	varchar(150)	<input checked="" type="checkbox"/>
Resume	varchar(255)	<input checked="" type="checkbox"/>
zzm	char(4)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

图 5.1 学生信息 St\_Info 表的表结构

WLS-PC.student_db - dbo.St_Info									
St_ID	St_Name	St_Sex	Born_Date	CJ_Name	Telephone	Address	Resume	zzmm	
0603060108	徐文文	男	1987-12-10 00:...	材料科学0601	0731_20223388	湖南省长沙市...	NULL	团员	
0603060109	黄正刚	男	1987-12-26 00:...	材料科学0601	NULL	贵州省平坝县...	NULL	团员	
0603060110	张红飞	男	1988-03-29 00:...	材料科学0601	NULL	河南省焦作市...	NULL	团员	
0603060111	曾莉娟	女	1987-05-13 00:...	材料科学0601	NULL	湖北省天门市...	NULL	团员	
0603060201	张红飞	男	1988-03-29 00:...	材料科学0602	NULL	河南省焦作市...	NULL	团员	
2001050105	邓红艳	女	1986-07-03 00:...	法学0501	NULL	广西桂林市兴...	NULL	团员	
2001050106	金萍	女	1984-11-06 00:...	法学0502	NULL	广西桂平市社...	NULL	团员	
2001050107	吴中华	男	1985-04-10 00:...	法学0503	NULL	河北省邯郸市...	NULL	团员	
2001050108	王铭	男	1987-09-09 00:...	法学0504	NULL	河南省上蔡县...	2003年获县级...	团员	
2001060103	郑远月	男	1986-06-18 00:...	法学0601	0731_88837342...	湖南省邵阳市...	2003年获市级...	团员	
2001060104	张力明	男	1987-08-29 00:...	法学0602	8834123	安徽省太湖县...	NULL	团员	
2001060105	张好然	女	1988-04-19 00:...	法学0603	010_86634234 ...	北京市西城区...	NULL	团员	
2001060106	李娜	女	1988-10-21 00:...	法学0604	13518473581 ...	重庆市黔江中学	NULL	团员	
2602060105	杨平娟	女	1988-05-20 00:...	口腔(七)0601	NULL	北京市西城区...	NULL	团员	
2602060106	王小维	男	1987-12-11 00:...	口腔(七)0601	NULL	泉州泉秀花园...	NULL	团员	
2602060107	刘小玲	女	1988-05-20 00:...	口腔(七)0601	NULL	厦门市前埔二...	NULL	团员	
2602060108	何邵阳	男	1987-06-01 00:...	口腔(七)0601	NULL	广东省韶关市...	NULL	团员	
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

图 5.2 学生信息 St\_Info 表的表记录

(2) 向学生数据库 Student\_db 中添加课程信息 C\_Info 表, 表结构和表记录分别如图 5.3 和图 5.4 所示。

WLS-PC.student_db - dbo.C_Info			
列名	数据类型	允许 Null 值	
C_No	char(10)	<input type="checkbox"/>	
C_Name	varchar(30)	<input type="checkbox"/>	
C_Type	char(4)	<input checked="" type="checkbox"/>	
C_Credit	smallint	<input type="checkbox"/>	
C_Des	varchar(255)	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

图 5.3 课程信息 C\_Info 表的表结构

WLS-PC.student_db - dbo.C_Info					
	C_No	C_Name	C_Type	C_Credit	C_Des
▶	19010122	艺术设计史	选修	6	NULL
	20010051	民法学	必修	7	NULL
	29000011	体育	必修	6	NULL
	9710011	大学计算机基础	必修	2	NULL
	9710021	VB程序设计基础	必修	4	NULL
	9710031	数据库应用基础	必修	4	NULL
	9710041	C++程序设计...	必修	4	NULL
	9720013	大学计算机基...	实践	2	NULL
	9720033	数据库应用基...	实践	3	本实践是在学...
	9720043	C++程序课程...	实践	4	NULL
	9720044	网络技术与应用	选修	4	NULL
	97200445	网络技术与应用1	选修	4	NULL
	9720045	Web开发技术	NULL	3	NULL
*	NULL	NULL	NULL	NULL	NULL

图 5.4 课程信息 C\_Info 表的表记录

(3) 向学生数据库 Student\_db 中添加学院信息 D\_Info 表, 表结构和表记录分别如图 5.5 和图 5.6 所示。

WLS-PC.student_db - dbo.D_Info			
列名	数据类型	允许 Null 值	
D_ID	char(2)	<input type="checkbox"/>	
D_Name	varchar(30)	<input type="checkbox"/>	
		<input type="checkbox"/>	

图 5.5 学院信息 D\_Info 表的表结构

WLS-PC.student_db - dbo.D_Info	
D_ID	D_Name
01	地学与环境工...
02	资源与安全工...
03	资源加工与生...
04	信息物理工程...
05	冶金科学与工...
06	材料科学与工...
07	粉末冶金研究院
08	机电工程学院
09	信息科学与工...
10	能源科学与工...
11	交通运输工程...
12	土木建筑学院
13	数学科学与计...
14	物理科学与核...
15	化学化工学院
16	商学院
17	文学院
18	外国语学院
19	艺术学院
20	法学院
21	政治学与行政...
22	湘雅医学院
23	药学院
24	护理学院
25	公共卫生学院
26	口腔医学院

图 5.6 学院信息 D\_Info 表的表记录

完成以上操作后，根据创建的数据表在下列横线处填入相关数据。

- ① 学生信息 St\_Info 表的主键名是：\_\_\_\_\_。
- ② 学生信息 St\_Info 表中不允许为空的字段有：\_\_\_\_\_。
- ③ 学生信息 St\_Info 表中的 Born\_Date 字段是\_\_\_\_\_类型，其宽度由\_\_\_\_\_设定。
- ④ 课程信息 C\_Info 表中的 C\_Credit 字段是 smallint 类型，其宽度为\_\_\_\_\_字节。

## 2. 在查询设计器中使用 CREATE TABLE 命令创建数据表。

(1) 阅读下列语句，语句中的 S\_NO 表示学生学号，NAME 表示学生名称，AGE 表示学生年龄，然后在查询窗口中输入：

```
CREATE TABLE Student
(S_NO CHAR(7) PRIMARY KEY(S_NO),
NAME CHAR(10),
AGE SMALLINT CHECK(AGE BETWEEN 15 AND 20))
```

执行上述语句后，在下列横线处填入相关数据。

- ① 创建的数据表的表名是：\_\_\_\_\_。
- ② 数据表的主键名是：\_\_\_\_\_。
- ③ AGE 列的数据类型是：\_\_\_\_\_。
- ④ 语句 CHECK(AGE BETWEEN 15 AND 20)表示\_\_\_\_\_。

(2) 阅读下列语句，语句中的 St\_ID 表示学生号，C\_NO 表示课程号，Score 表示所修课程的成绩，然后在查询窗口中输入：

```
CREATE TABLE S_C_Info
(St_ID CHAR(10) NOT NULL,
C_NO CHAR(10) NOT NULL,
Score INT NULL,
PRIMARY KEY(St_ID, C_NO),
FOREIGN KEY(St_ID) REFERENCES St_Info(St_ID),
FOREIGN KEY(C_NO) REFERENCES C_Info(C_NO))
```

完成上机操作后，在下列横线处填入相关数据。

- ① 数据表 S\_C\_Info 的主键名是: \_\_\_\_\_。
- ② Score 列的数据类型是: \_\_\_\_\_。
- ③ 语句 FOREIGN KEY(St\_ID) REFERENCES St\_Info(St\_ID)表示\_\_\_\_\_。

3. 在对象资源管理器中将实验内容 2 (2) 中创建的 S\_C\_Info 数据表的记录输入, 记录内容如图 5.7 所示。

WLS-PC.student_db - dbo.S_C_Info		
St_ID	C_No	Score
0603060108	9710021	56
0603060108	9710041	67
0603060109	9710041	78
0603060110	9710041	52
0603060111	9710041	99
2001050105	9710011	88
2001050105	9720013	90
2001050106	9710011	89
2001050106	9720013	93
2001050107	9710011	76
2001050107	9720013	77
2001050108	9710011	66
2001050108	9720013	88
2602060105	29000011	77
2602060106	29000011	97
2602060107	29000011	92
2602060108	29000011	83
* NULL	NULL	NULL

图 5.7 选课信息 S\_C\_Info 表的表记录

4. 在对象资源管理器中修改和删除记录。
- (1) 将学生信息 St\_Info 表中学号为“2001050108”的记录删除。
- (2) 将课程信息 C\_Info 表中“民法学”的学分修改为 6 学分。
5. 将下列语句在查询设计器中完成上机操作后, 在横线处填入相关数据。
- (1)

```
INSERT INTO St_Info
VALUES ('2001050109', '杨柳', '女', '1988-12-12', '法学 0503', null, null, null, null)
```

这个语句的功能是: \_\_\_\_\_。

(2)

```
UPDATE St_Info
SET CL_Name='计算机科学 0601'
WHERE St_ID='0603060109'
```

这个语句的功能是: \_\_\_\_\_。

6. 在对象资源管理器和查询设计器中分别实现以下操作:
- (1) 在 S\_C\_Info 表中添加一个新列: 选课类别, 列名为 xklb, 类型为 char(4)。
- (2) 将 C\_Info 表中的 C\_Credit 列的类型修改为 tinyint。
- (3) 将 S\_C\_Info 表中新添加的列 xklb 的类型改为 char(6)。

#### 四、实验思考

1. 在对象资源管理器中能否对数据表更名?
2. 在对象资源管理器中能否对数据表进行复制? 复制的内容和方法是什么?

3. 在数据库中，取 NULL 值与取零值的含义相同吗？如果不同，它们的区别是什么？
4. 主键的设置如何操作？在对象资源管理器中使多列作主键时应如何操作？

## 实验 6 数据库完整性设置

### 一、实验目的

1. 掌握用对象资源管理器创建关系图的方法。
2. 掌握用对象资源管理器和 CREATE TABLE 语句创建主键约束、唯一性约束和 NOT NULL 约束的方法。
3. 掌握用对象资源管理器和 CREATE TABLE 语句创建 CHECK 约束和 DEFAULT 约束的方法。
4. 掌握用对象资源管理器删除约束的方法。

### 二、实验准备

1. 了解关系图的概念。
2. 了解数据库完整性的概念。
3. 了解约束的类型和创建约束、删除约束的语法。

### 三、实验内容及步骤

1. 使用对象资源管理器在数据库 Student\_db 中创建 StCSCD 关系图。关系图由实验 5 中创建的学生信息 St\_Info 表、课程信息 C\_Info 表、选课信息 S\_C\_Info 表组成，如图 6.1 所示。

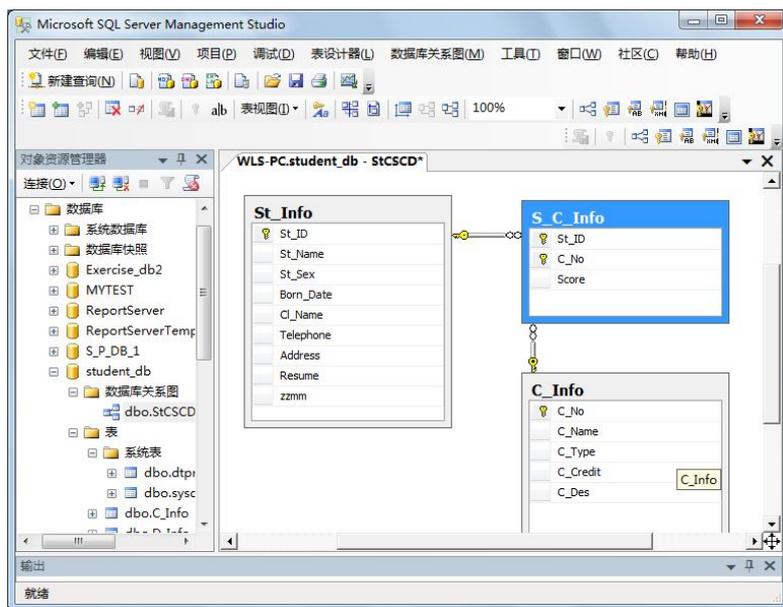


图 6.1 StCSCD 关系图

提示: 创建 StCSCD 关系图之前必须先创建好 St\_Info 表和 C\_Info 表的主键。

2. 打开 St\_Info 表并修改 Telephone 字段, 将长度 varchar(20)修改为 varchar(15)。

注意: 要修改此选项, 需在“工具”菜单中单击“选项”命令, 打开“选项”对话框, 如图 6.2 所示, 单击“表设计器和数据库设计器”, 清除“阻止保存要求重新创建表的更改”复选框。

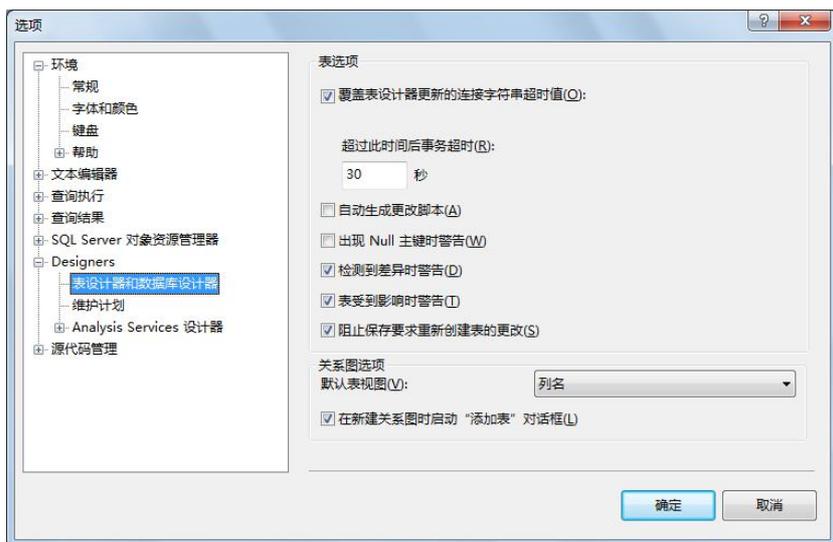


图 6.2 “选项”对话框

3. 在表 St\_Info 中, 为学号创建一个 CHECK 约束 CK\_St\_id, 限制所输入的学号为 10 位 0~9 的数字符号组成。

提示:

(1) 创建 CHECK 约束 CK\_St\_id 的表达式为:

`st_id like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]`

(2) 在 St\_Info 表中输入记录, 学号为 2602060A08, 检验 CHECK 约束的有效性。

4. 在 St\_Info 表的“性别”列中创建一个 CHECK 约束 CK\_St\_info\_sex, 以保证输入的性别值只能是“男”或“女”。

提示: 约束 CK\_St\_id 的表达式为“`st_sex like '[男女]'`”。

5. 删除 CK\_St\_info\_sex 约束。

6. 在 Student\_db 数据库中建立日期、货币和字符等数据类型的 DEFAULT 约束。

(1) 在 Student\_db 数据库中创建 stu\_fee 数据表, 表结构如图 6.3 所示。

WLS-PC.student_db - dbo.stu_fee			
列名	数据类型	允许 Null 值	
学号	char(10)	<input type="checkbox"/>	
姓名	varchar(10)	<input checked="" type="checkbox"/>	
学费	int	<input checked="" type="checkbox"/>	
交费日期	datetime	<input checked="" type="checkbox"/>	
电话号码	char(11)	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

图 6.3 stu\_fee 数据表的表结构

(2) 在 `stu_fee` 的表结构中选择“交费日期”列，在“列”选项卡的“默认值或绑定”文本框中输入 ('20130913')，即设置交费日期的默认值为 2013 年 9 月 13 日。

(3) 参照以上操作，在 `stu_fee` 的表结构中为“学费”列和“电话号码”列创建 DEFAULT 约束，其值分别为 \$100 和 unknown。

(4) 在对象资源管理器中对 `stu_fee` 数据表输入 3 条记录，输入时观察 `stu_fee` 表的数据变化情况，操作完成后删除 DEFAULT 约束。

7. 用对象资源管理器在 `Student_db` 数据库中创建表 `St_c`，数据结构如图 6.4 所示。

列名	数据类型	允许 Null 值
St_ID	char(4)	<input type="checkbox"/>
St_Name	varchar(8)	<input type="checkbox"/>
St_Sex	char(2)	<input type="checkbox"/>
Born_Date	datetime	<input checked="" type="checkbox"/>

图 6.4 St\_c 表的数据结构

要求：

- (1) 将 `St_ID` 设置为主键，主键名为：\_\_\_\_\_。
- (2) 为 `St_Name` 创建唯一性约束 (UNIQUE)，约束名为 `uk_stname`。
- (3) 设置 `Born_Date` 允许空。
- (4) 为表 `St_c` 插入以下记录：
 

0011	王芳	1997-02-10
0012	王芳	1996-12-12

观察出现的情况并确定产生的原因。

**提示：**创建唯一约束的操作是先在设计器中选中列“`St_Name`”并右击，在弹出的快捷菜单中单击“索引/键”命令，在弹出的“索引/键”对话框中进行设置，如图 6.5 所示。

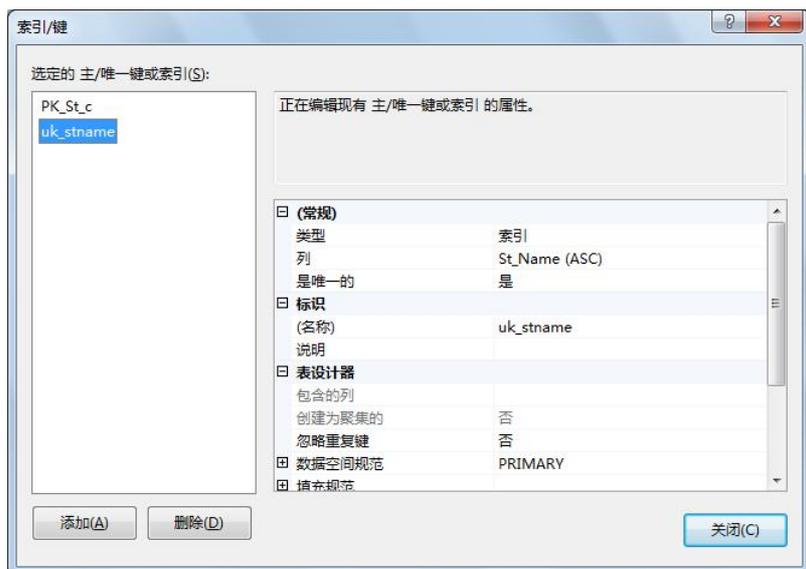


图 6.5 为 `St_Name` 创建唯一性约束

(5) 使用 ALTER TABLE 语句的 DROP CONSTRAINT 参数项在查询设计器中删除为 St\_c 表所建的约束。

#### 四、实验思考

1. 实体完整性和域完整性分别是对数据库的哪些方面进行保护? 如何设置?
2. 在数据库中, 主键是唯一的吗? 如果新输入的记录主键和原来的主键重复会出现什么后果?
3. 实现参照完整性有什么意义? 假设存在学生信息表 Student(s\_no,s\_name,age,d\_no)和院系信息表 dept(d\_no,d\_name,address,te1), 为什么要确保 Student 表中的 d\_no 列的取值参照 dept 表的 d\_no 列?
4. SQL Server 提供了哪两个独立于表的数据对象用以维护数据库的完整性? 它们各自的作用和功能是什么? 如何设置?

## 实验 7 基本查询

### 一、实验目的

1. 掌握数据查询的概念和查询语句的执行方法。
2. 掌握简单查询和条件查询的查询方法。
3. 掌握查询结果的处理方法。
4. 掌握嵌套查询的查询方法。
5. 掌握等值内连接的查询方法。

### 二、实验准备

1. 复习数据查询的概念和查询语句的一般格式。
2. 复习查询条件的表示方法。
3. 了解查询结果的各种实现方法。

### 三、实验内容及步骤

1. 根据题目要求, 在下列横线处填入适当内容, 实现相关操作或回答相关问题。
  - (1) 在数据库 Student\_db 的数据表 C\_Info 中查询所有课程信息, 查询结果集如图 7.1 所示。  
SELECT \* \_\_\_\_\_ C\_Info
  - (2) 查询全体学生的姓名、出生年份和所在班级的信息, 查询结果集如图 7.2 所示。  
SELECT St\_name AS 姓名, \_\_\_\_\_(Born\_Date) AS 出生年份, Cl\_Name AS 班级  
FROM St\_info
  - (3) 查询 1988 年出生的学生的姓名及其年龄, 查询结果集如图 7.3 所示。  
SELECT St\_name AS 姓名, Year(getdate())-Year(Born\_date) AS 年龄  
FROM St\_info  
WHERE \_\_\_\_\_

	C_No	C_Name	C_Type	C_Credit	C_Des
1	19010122	艺术设计史	选修	6	NULL
2	20010051	民法学	必修	7	NULL
3	29000011	体育	必修	6	NULL
4	9710011	大学计算机基础	必修	2	NULL
5	9710021	VB程序设计基础	必修	4	NULL
6	9710031	数据库应用基础	必修	4	NULL
7	9710041	C++程序设计基础	必修	4	NULL
8	9720013	大学计算机基...	实践	2	NULL
9	9720033	数据库应用基...	实践	3	本实践是在学完《数据库技术与...
10	9720043	C++程序课程设计	实践	4	NULL
11	9720044	网络技术与应用	选修	4	NULL
12	97200445	网络技术与应用1	选修	4	NULL
13	9720045	Web开发技术	NULL	3	NULL

图 7.1 实验内容 1 (1) 的查询结果集

	姓名	出生年份	班级
1	徐文文	1987	材料科学0601
2	黄正刚	1987	计算机科学0601
3	张红飞	1988	材料科学0601
4	曾莉娟	1987	材料科学0601
5	张红飞	1988	材料科学0602
6	邓红艳	1986	法学0501
7	金萍	1984	法学0502
8	吴中华	1985	法学0503
9	王铭	1987	法学0504
10	杨柳	1988	法学0503
11	郑远月	1986	法学0601
12	张力明	1987	法学0602
13	张好然	1988	法学0603
14	李娜	1988	法学0604
15	杨平娟	1988	口腔(七)0601
16	王小维	1987	口腔(七)0601
17	刘小玲	1988	口腔(七)0601
18	何邵阳	1987	口腔(七)0601

	姓名	年龄
1	张红飞	25
2	张红飞	25
3	杨柳	25
4	张好然	25
5	李娜	25
6	杨平娟	25
7	刘小玲	25

图 7.2 实验内容 1 (2) 的查询结果集

图 7.3 实验内容 1 (3) 的查询结果集

(4) 查询考试成绩在 85 分以上的学生的学号。

```
SELECT DISTINCT St_ID FROM S_C_info WHERE Score>85
```

语句中的 DISTINCT 表示: \_\_\_\_\_。

```
SELECT St_ID FROM S_C_info WHERE Score>85
```

以上两条语句的区别是: \_\_\_\_\_。

(5) 对 S\_C\_info 表列出成绩在 70~80 分之间的学生名单。

```
SELECT * FROM S_C_info WHERE Score BETWEEN 70 AND 80
```

这条语句的等价语句是: \_\_\_\_\_。

(6) 查询所有姓“王”的学生的姓名、学号和性别。

```
SELECT St_name, St_ID, St_sex
```

```
FROM St_Info
```

```
WHERE St_name LIKE _____
```

(7) 查询选修了课程号为 9710011 的课程的学生的学号和成绩, 并按分数降序排列。

```
SELECT St_ID, Score
```

```
FROM S_C_info
```

WHERE C\_NO=

(8) 使用合并查询列出 C\_Info 表中“艺术设计史”或“民法学”的课程代码、课程类型和学分。

```
SELECT C_NO, C_Type, C_credit FROM C_Info WHERE C_name='艺术设计史'
```

```
SELECT C_no, C_Type, C_credit FROM C_Info WHERE C_name='民法学'
```

(9) 查询全体学生的情况, 查询结果按学号升序排列, 结果存入新表 new 中并浏览该表。

```
SELECT * INTO new
```

```
FROM St_Info
```

```
SELECT * FROM new
```

(10) 对 St\_Info 表, 分别统计各班级的学生人数, 其查询结果集如图 7.4 所示。

```
SELECT
```

```
FROM St_Info
```

```
GROUP BY Cl_Name
```

(11) 对 S\_C\_Info 表中选修了课程编号为 29000011 的体育课的学生的平均成绩生成汇总行和明细行, 其查询结果集如图 7.5 所示。

```
SELECT St_ID, Score
```

```
FROM S_C_Info
```

```
WHERE C_No = '29000011'
```

```
COMPUTE
```

	班级名	人数
1	材料科学0601	3
2	材料科学0602	1
3	法学0501	1
4	法学0502	1
5	法学0503	2
6	法学0504	1
7	法学0601	1
8	法学0602	1
9	法学0603	1
10	法学0604	1
11	计算机科学0601	1
12	口腔(七)0601	4

图 7.4 实验内容 1 (10) 的查询结果集

	St_ID	Score
1	2602060105	77
2	2602060106	97
3	2602060107	92
4	2602060108	83
avg		
1	87	

图 7.5 实验内容 1 (11) 的查询结果集

2. 根据下列内容写出 SELECT 语句, 上机操作完成基本查询。

- (1) 在数据库 Student\_db 的数据表 St\_Info 中查询全体学生的所有信息。
- (2) 在数据库 Student\_db 的数据表 St\_Info 中查询每个学生的学号、姓名、出生日期信息。
- (3) 查询学号为 2001050108 的学生的姓名和家庭住址。
- (4) 找出所有男同学的学号和姓名。
- (5) 查询 St\_Info 表中班级名为“材料科学 0601”的学生的学号、姓名、班级名。
- (6) 查询 C\_Info 表中学分数为 3 和 4 的课程信息。
- (7) 统计 S\_C\_Info 表中每门课程的平均成绩, 要求显示课程编号和平均成绩。
- (8) 查询所有姓“张”的学生的学号、姓名、性别和出生日期。

3. 上机完成下列语句的操作, 观察输出结果。

- (1) SELECT COUNT(\*) FROM S\_C\_Info
- (2) SELECT SUBSTRING(St\_Name,1,1) FROM St\_Info
- (3) SELECT Year(Getdate()), Month(Getdate()), Day(Getdate())

#### 四、实验思考

1. 使用下列语句查询年龄最大学生的姓名和年龄时系统报错, 为什么?

```
SELECT St_Name, MAX(year(getdate())-Year(Born_Date))
FROM St_Info
```

2. 使用下列语句查询修课成绩最高学生的学号时系统报错, 为什么?

```
SELECT St_ID
FROM S_C_Info
WHERE Score=MAX(Score)
```

## 实验 8 嵌套查询

### 一、实验目的

1. 理解数据库嵌套查询的概念与作用。
2. 掌握数据查询中 IN、ANY、SOME 和 ALL 等操作符的使用方法。

### 二、实验准备

1. 复习嵌套查询的概念与作用。
2. 了解 IN、ANY、SOME 和 ALL 等操作符的功能与使用方法。

### 三、实验内容及步骤

1. 阅读语句, 上机完成查询操作。

(1) 在表 St\_info 中查找与杨平娟在同一个班级学习的学生信息。

```
SELECT * FROM St_info
WHERE Cl_name=
  (SELECT Cl_name FROM St_info
   WHERE St_name='杨平娟')
```

(2) 使用 IN 子查询查找选修了课程名为“体育”的学生的学号和成绩。

```
SELECT St_ID, Score
FROM S_C_info
WHERE C_No IN
  (SELECT C_No FROM C_info
   WHERE C_name='体育')
```

(3) 查询选修了课程编号为 9710011 和 9710041 的课程的学生的学号和姓名。

```
SELECT St_ID, St_name
FROM St_info
```

```
WHERE St_ID IN
      (SELECT St_ID FROM S_C_info
       WHERE C_NO IN ('9710011', '9710041'))
```

2. 在下列横线处填入适当内容, 完善语句并实现相应功能。

(1) 查询成绩高于 90 分的学生的学号和姓名, 查询结果集如图 8.1 所示。

```
SELECT St_ID, St_Name
FROM St_Info
WHERE St_ID _____
      (SELECT St_ID FROM _____
       WHERE Score >= 90)
```

(2) 查询选修了学分数为 4 的课程的学生的学号、课程编号、成绩信息, 查询结果集如图 8.2 所示。

```
SELECT St_ID, C_No, Score
FROM S_C_Info
WHERE _____ IN
      (SELECT C_No FROM C_Info
       WHERE _____)
```

	St_ID	St_Name
1	0603060111	曾莉娟
2	2001050105	邓红艳
3	2001050106	金萍
4	2602060106	王小维
5	2602060107	刘小玲

图 8.1 实验内容 2 (1) 的查询结果集

	St_ID	C_No	Score
1	0603060108	9710021	56
2	0603060108	9710041	67
3	0603060109	9710041	78
4	0603060110	9710041	52
5	0603060111	9710041	99

图 8.2 实验内容 2 (2) 的查询结果集

(3) 查询其他班级中比“法学 0501”的学生年龄都大的学生姓名和年龄, 查询结果集如图 8.3 所示。

```
SELECT St_name, Year(Getdate())-Year(Born_date) AS '年龄'
FROM St_info
WHERE _____
      (SELECT Year(Getdate())-Year(Born_date)
       FROM St_info
       WHERE Cl_name='法学 0501')
```

	St_name	年龄
1	金萍	29
2	吴中华	28

图 8.3 实验内容 2 (3) 的查询结果集

(4) 列出选修 9710041 (即“C 语言程序设计基础”) 的学生的成绩比选修 29000011 (即“体育”) 的学生的最低成绩高的学生的学号和成绩, 查询结果集如图 8.4 所示。

```
SELECT St_id, score
FROM S_C_info
WHERE _____
      (SELECT Score
       FROM S_C_info
       WHERE C_NO='29000011')
```

	St_id	score
1	0603060111	99

图 8.4 实验内容 2 (4) 的查询结果集

(5) 查询没有选修 9710041 号课程的学生姓名和所在班级, 查询结果集如图 8.5 所示。

```
SELECT St_Name, Cl_Name
FROM St_Info
WHERE _____
```

```
(SELECT St_ID FROM S_C_info
WHERE C_NO='9710041')
```

(6) 查询所有姓“王”的学生所修课程的成绩，查询结果集如图 8.6 所示。

```
SELECT C_No, Score
FROM S_C_Info
WHERE _____ IN
( SELECT St_ID FROM St_Info
WHERE St_Name LIKE _____ )
```

	St_Name	Ci_Name
1	张红飞	材料科学0602
2	邓红艳	法学0501
3	金萍	法学0502
4	吴中华	法学0503
5	王铭	法学0504
6	杨柳	法学0503
7	郑远月	法学0601
8	张力明	法学0602
9	张好然	法学0603
10	李娜	法学0604
11	杨平娟	口腔(七)0601
12	王小维	口腔(七)0601
13	刘小玲	口腔(七)0601
14	何邵阳	口腔(七)0601

图 8.5 实验内容 2 (5) 的查询结果集

	C_No	Score
1	9710011	66
2	9720013	88
3	29000011	97

图 8.6 实验内容 2 (6) 的查询结果集

#### 四、实验思考

1. 查询与“口腔（七）0601”班所有学生的年龄均不同的学生学号、姓名和年龄。
2. 查询选修了学号为 2001050105 的学生所选修的全部课程的学生学号和姓名。
3. 查询每个学生的课程成绩最高的成绩信息。
4. 列出学号为 2001050108 的学生的分数比学号为 2001050105 的学生的最低分数高的课程编号和分数。

## 实验 9 多表连接查询和综合查询

### 一、实验目的

1. 理解数据库多表查询的概念与作用。
2. 掌握多表连接查询的方法。
3. 掌握各种形式的查询方法。

### 二、实验准备

1. 复习多表连接查询的概念与作用。
2. 了解多表连接查询的种类、区别与实现方法。

### 三、实验内容及步骤

1. 写出语句, 上机完成查询操作。

(1) 使用多表连接查询分数在 80~90 范围内的学生的学号、姓名和分数, 查询结果集如图 9.1 所示。

	St_ID	St_Name	Score
1	2001050105	邓红艳	88
2	2001050105	邓红艳	90
3	2001050106	金萍	89
4	2001050108	王铭	88
5	2602060108	何邵阳	83

图 9.1 实验内容 1 (1) 的查询结果集

(2) 使用多表连接查询选修“C++程序设计基础”课程的学生的学号、姓名和分数, 查询结果集如图 9.2 所示。

	St_ID	St_Name	Score
1	0603060108	徐文文	67
2	0603060109	黄正刚	78
3	0603060110	张红飞	52
4	0603060111	曾莉娟	99

图 9.2 实验内容 1 (2) 的查询结果集

(3) 查询所有课程的不及格成绩单, 要求给出学生的学号、姓名、课程名称和成绩, 查询结果集如图 9.3 所示。

	St_ID	St_Name	c_name	Score
1	0603060108	徐文文	VB程序设计基础	56
2	0603060110	张红飞	C++程序设计基础	52

图 9.3 实验内容 1 (3) 的查询结果集

2. 在下列横线处填入适当内容, 完善语句并实现相应功能。

(1) 查询法专业的学生学号、姓名、课程名称和成绩, 并按学号升序排序, 查询结果集如图 9.4 所示。

```
SELECT st.St_ID, St_Name, C_Name, Score
FROM St_Info st JOIN S_C_Info sc ON _____
      JOIN C_Info c ON sc.C_NO=c.C_No
WHERE Cl_Name LIKE _____
ORDER BY st.St_ID
```

	St_ID	St_Name	C_Name	Score
1	2001050105	邓红艳	大学计算机基础	88
2	2001050105	邓红艳	大学计算机基础实践	90
3	2001050106	金萍	大学计算机基础	89
4	2001050106	金萍	大学计算机基础实践	93
5	2001050107	吴中华	大学计算机基础	76
6	2001050107	吴中华	大学计算机基础实践	77
7	2001050108	王铭	大学计算机基础	66
8	2001050108	王铭	大学计算机基础实践	88

图 9.4 实验内容 2 (1) 的查询结果集

(2) 查询每门课程的课程名称和最高分, 查询结果集如图 9.5 所示。

```
SELECT C_Name AS 课程名称, _____ AS 最高分
FROM S_C_Info sc JOIN C_Info c ON _____
```

GROUP BY C\_Name

(3) 将 C\_Info 表左外连接 S\_C\_Info 表。

```
SELECT a.C_NO,a.C_Name, b.St_ID,b.Score
FROM C_Info _____
      S_C_Info b ON a.C_NO=b.C_NO
```

(4) 将 C\_Info 表右外连接 S\_C\_Info 表。

```
SELECT a.C_NO, a.C_Name,b.St_ID, b.Score
FROM C_Info _____
      S_C_Info b ON a.C_NO = b.C_NO
```

(5) 将 C\_Info 表全外连接 S\_C\_Info 表。

```
SELECT C_Info.C_NO, C_Name,S_C_Info.C_NO, S_C_Info.Score
FROM C_Info _____
      S_C_Info ON C_Info.C_NO = S_C_Info.C_NO
```

3. 查询所有必修课程的课程号、课程名称、学分及选修学生的姓名和分数。

4. 查询每个学生所选课程的最高成绩，要求列出学号、姓名、课程编号和分数。

5. 查询所有学生的总成绩，要求列出学号、姓名、总成绩，没有选修课程的学生的总成绩为空。

6. 查询所有男同学的选课情况，要求列出学号、姓名、课程名称和分数。

#### 四、实验思考

1. 如何查询“大学计算机基础”课程考试成绩前3名的学生的姓名和成绩？

2. SQL Server 数据库中的多表连接查询和子查询的区别是什么？

3. 连接可以在 SELECT 语句的 FROM 子句或 WHERE 子句中建立，为什么？在 FROM 子句和在 WHERE 子句连接的区别是什么？哪种连接方式好一些？为什么？

	课程名称	最高分
1	C++程序设计基础	99
2	VB程序设计基础	56
3	大学计算机基础	89
4	大学计算机基础实践	93
5	体育	97

图 9.5 实验内容 2 (2) 的查询结果集

## 实验 10 创建数据库的索引和视图

### 一、实验目的

1. 学会使用对象资源管理器和 T-SQL 语句创建索引。
2. 学会使用对象资源管理器查看索引和删除索引。
3. 掌握使用对象资源管理器和 T-SQL 语句 CREATE VIEW 创建视图。
4. 掌握使用对象资源管理器和 T-SQL 语句 ALTER VIEW 修改视图。
5. 掌握使用对象资源管理器和 T-SQL 语句 DROP VIEW 删除视图。

### 二、实验准备

1. 了解索引的作用与分类；了解聚集索引和非聚集索引的概念。
2. 了解视图的概念。
3. 了解使用对象资源管理器创建索引的步骤。
4. 了解创建视图的 T-SQL 语句 CREATE VIEW 的语法格式及用法。

5. 了解修改视图的 T-SQL 语句 ALTER VIEW 的语法格式。
6. 了解删除视图的 T-SQL 语句 DROP VIEW 的用法。

### 三、实验内容及步骤

1. 在对象资源管理器中创建索引。

(1) 为 Student\_db 数据库的 S\_C\_Info 表的成绩 “Score” 字段创建一个非聚集索引，命名为 Score\_index。

提示：具体操作参见图 10.1。

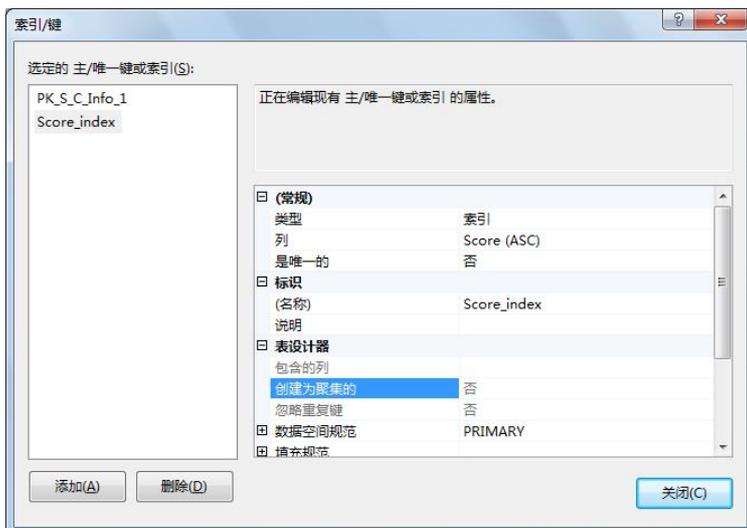


图 10.1 为 “Score” 字段创建非聚集索引 Score\_index

(2) 为 Student\_db 数据库的 S\_C\_Info 表的学号 “St\_ID” 和课程编号 “C\_No” 字段创建一个复合唯一非聚集索引，命名为 SC\_id\_c\_ind。

提示：参照图 10.2 所示操作创建一个复合唯一非聚集索引。

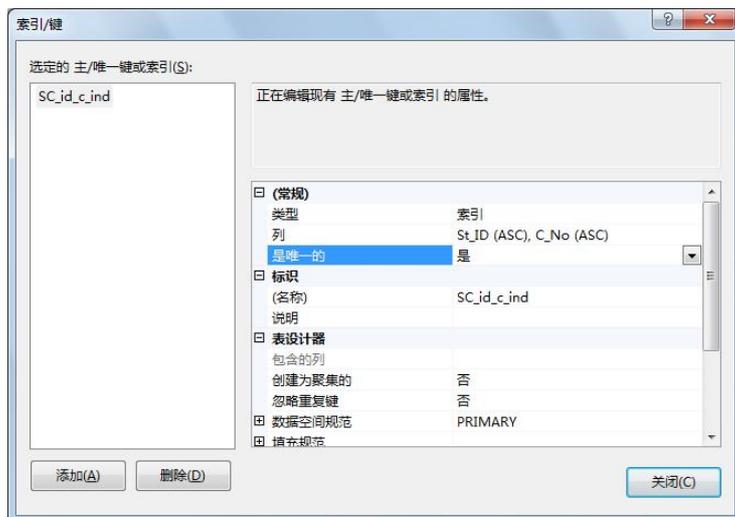


图 10.2 为学号 “St\_ID” 和课程编号 “C\_No” 创建的复合唯一非聚集索引

2. 在对象资源管理器中查看所建的索引 Score\_index 和 SC\_id\_c\_ind, 如图 10.3 所示。

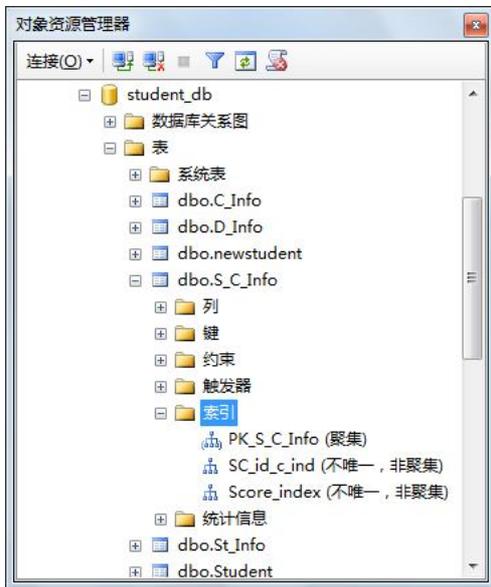


图 10.3 在对象资源管理器中查看索引

3. 使用对象资源管理器为 S\_C\_Info 表的学号 “St\_ID” 和课程编号 “C\_No” 字段创建一个唯一聚集索引, 命名为 S\_C\_index。

提示: 若 S\_C\_Info 表为学号 “St\_ID” 和课程编号 “C\_No” 字段创建了主键, 需先取消 S\_C\_Info 表的主键设置, 才能再创建一个复合唯一聚集索引。这是为什么?

4. 使用对象资源管理器删除索引 SC\_id\_c\_ind。

5. 使用对象资源管理器创建下列视图。

(1) 在 Student\_db 数据库中以 St\_Info 表为基础, 建立名为 v\_stu\_i 的视图, 使视图显示学生姓名、性别、家庭住址, 如图 10.4 所示。

姓名	性别	家庭住址
徐文文	男	湖南省长沙市韶山北路
黄正刚	男	贵州省平坝县夏云中学
张红飞	男	河南省焦作市西环路26号
曾莉娟	女	湖北省天门市多宝镇公益村六组
张红飞	男	河南省焦作市西环路26号
邓红艳	女	广西桂林市兴安县溶江镇司门街
金萍	女	广西桂平市社坡福和11队
吴中华	男	河北省邯郸市东街37号
王铭	男	河南省上蔡县大路李乡涧沟王村
郑远月	男	湖南省邵阳市一中
张力明	男	安徽省太湖县北中镇桐山村
张好然	女	北京市西城区新街口外大街34号
李娜	女	重庆市黔江中学
杨平娟	女	北京市西城区复兴门内大街97号
王小维	男	泉州泉秀花园西区十二幢
刘小玲	女	厦门市前埔二里42号0306室
何邵阳	男	广东省韶关市广东北江中学
* NULL	NULL	NULL

图 10.4 v\_stu\_i 视图的结果集

(2) 基于表 St\_Info、表 S\_C\_Info 和表 C\_Info, 建立一个名为 v\_stu\_c 的视图, 显示学生的学号、姓名、所选课程的课程名称, 如图 10.5 所示。

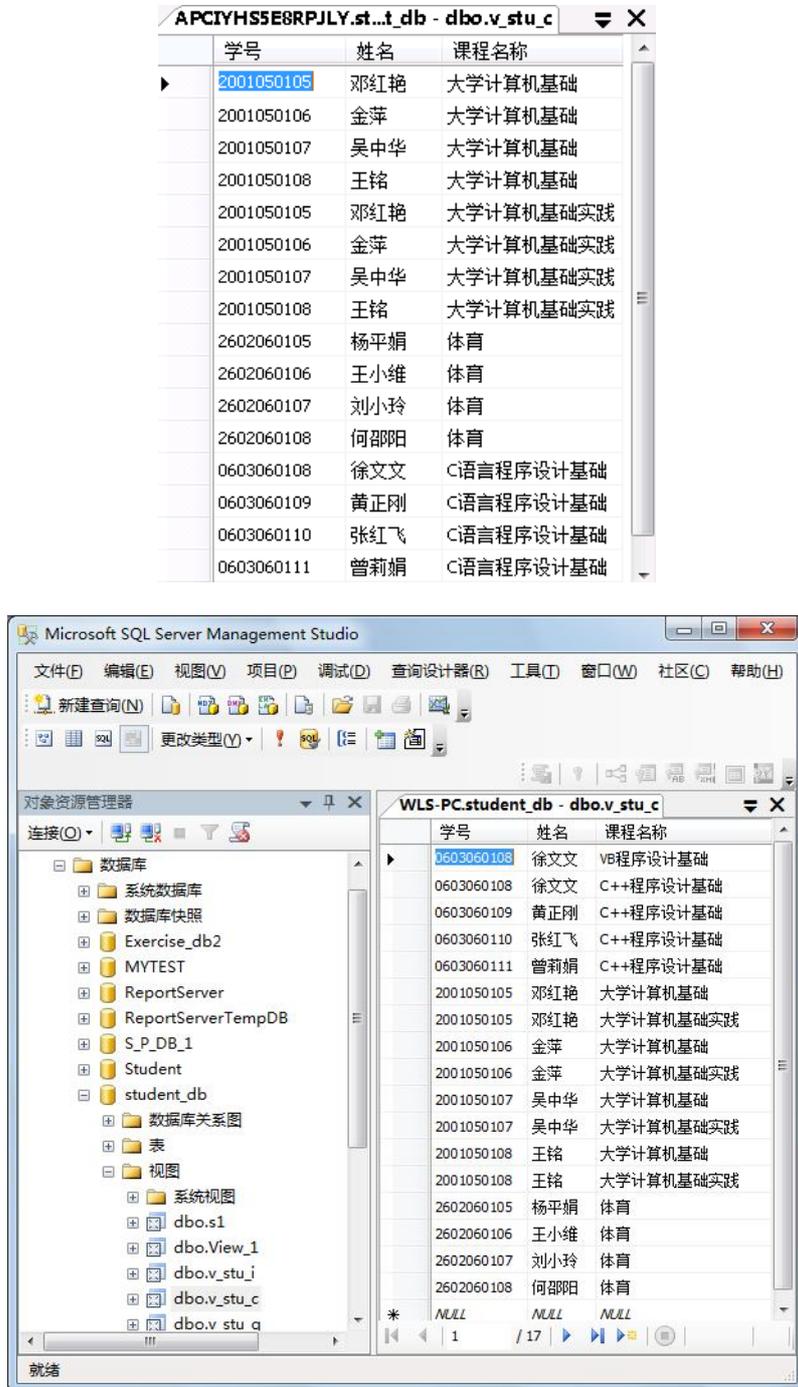


图 10.5 v\_stu\_c 视图结果集

(3) 基于表 St\_Info、表 C\_Info 和表 S\_C\_Info, 建立一个名为 v\_stu\_g 的视图, 显示所有学生的学号、姓名、课程名称、成绩, 如图 10.6 所示。

学号	姓名	课程名称	成绩
2001050105	邓红艳	大学计算机基础	88
2001050106	金萍	大学计算机基础	89
2001050107	吴中华	大学计算机基础	76
2001050108	王铭	大学计算机基础	66
2001050105	邓红艳	大学计算机基础实践	90
2001050106	金萍	大学计算机基础实践	93
2001050107	吴中华	大学计算机基础实践	77
2001050108	王铭	大学计算机基础实践	88
2602060105	杨平娟	体育	77
2602060106	王小维	体育	97
2602060107	刘小玲	体育	92
2602060108	何邵阳	体育	83
0603060108	徐文文	C语言程序设计基础	67
0603060109	黄正刚	C语言程序设计基础	78
0603060110	张红飞	C语言程序设计基础	52
0603060111	曾莉娟	C语言程序设计基础	99

St_ID	St_Na...	C_Name	Score
0603060108	徐文文	VB程序设计基础	56
0603060108	徐文文	C++程序设计基础	67
0603060109	黄正刚	C++程序设计基础	78
0603060110	张红飞	C++程序设计基础	52
0603060111	曾莉娟	C++程序设计基础	99
2001050105	邓红艳	大学计算机基础	88
2001050105	邓红艳	大学计算机基础实践	90
2001050106	金萍	大学计算机基础	89
2001050106	金萍	大学计算机基础实践	93
2001050107	吴中华	大学计算机基础	76
2001050107	吴中华	大学计算机基础实践	77
2001050108	王铭	大学计算机基础	66
2001050108	王铭	大学计算机基础实践	88
2602060105	杨平娟	体育	77
2602060106	王小维	体育	97
2602060107	刘小玲	体育	92
2602060108	何邵阳	体育	83
NULL	NULL	NULL	NULL

图 10.6 v\_stu\_g 视图的结果集

6. 在横线处写出使用视图 v\_stu\_g 查询学号为“2001050108”的学生的所有课程与成绩的查询语句，查询结果集如图 10.7 所示。

学号	姓名	课程名称	成绩
2001050108	王铭	大学计算机基础	66
2001050108	王铭	大学计算机基础实践	88

图 10.7 学号为“2001050108”的学生的视图信息

7. 使用 T-SQL 语句为学生信息表 St\_Info 创建视图 v\_Count, 统计“材料科学 0601”班的男生人数和女生人数, 如图 10.8 所示, 请补全以下语句:

```
CREATE VIEW v_Count
AS
SELECT _____, COUNT(*) AS 人数
FROM St_Info
WHERE Cl_Name = '材料科学 0601'
_____ St_Sex
```

	St_sex	人数
	男	3
	女	1

图 10.8 视图 v\_Count 信息

8. 请补全下列语句, 实现对表 S\_C\_Info 的文件视图 v\_stu\_a, 以查询学号为“2001050107”的学生的所有课程和成绩, 如图 10.9 所示。

```
CREATE VIEW _____
AS
SELECT * FROM S_C_Info
WHERE St_ID = _____
```

	St_ID	C_No	Score
	2001050107	9710011	76
	2001050107	9720013	77

图 10.9 视图 v\_stu\_a 信息

9. 使用 T-SQL 语句 DROP VIEW 删除视图 v\_stu\_c 和 v\_stu\_g。

10. 以下 T-SQL 语句 ALTER VIEW 修改视图 v\_stu\_i, 使其具有列名学号、姓名、性别, 请补全该语句。

```
ALTER VIEW _____
AS
SELECT St_ID, St_Name, St_Sex FROM St_Info
```

#### 四、实验思考

1. 索引和视图主要起什么作用? 它们的区别是什么?
2. 是否可以通过视图修改表 S\_C\_Info 中的数据?
3. 通过对视图的操作, 比较通过视图和基本表操作数据的异同。

## 实验 11 存储过程的创建和使用

### 一、实验目的

1. 了解存储过程的作用。
2. 掌握使用对象资源管理器和 T-SQL 语句创建存储过程的方法和步骤。

3. 掌握使用对象资源管理器和 T-SQL 语句执行存储过程的方法。
4. 掌握使用系统存储过程对用户自定义存储过程进行管理的方法。

## 二、实验准备

1. 了解存储过程的基本概念和分类。
2. 掌握通过对象资源管理器、向导和 T-SQL 语句创建存储过程的基本方法。
3. 掌握执行、查看、修改（内容和名称）、删除存储过程的使用方法。
4. 掌握在存储过程中使用参数的基本方法。

## 三、实验内容及步骤

存储过程是一系列预先编辑好的、能实现特定数据操作功能的 T-SQL 代码集，它与特定的数据库相关联，存储在 SQL Server 服务器上。用户可以像使用自定义函数那样重复调用这些存储过程，实现它所定义的操作。以下实验内容均以教材中所说明的学生数据库 Student\_db 为例进行操作。

1. 在对象资源管理器下创建一个名为 selectScore 的存储过程，输入以下代码，查询所有考试课程成绩优秀（≥90 分）的学生学号、姓名、课程名称和成绩，并按成绩降序排列。

```
CREATE PROCEDURE selectScore AS
SELECT a.St_ID, a.St_Name, b.C_Name, c.Score
FROM St_Info a, C_Info b, S_C_Info c
WHERE a.St_ID = c.St_ID AND b.C_No = c.C_No AND c.Score >= 90
ORDER BY c.Score DESC
```

执行以上语句，展开对象资源管理器下的“可编程性”→“存储过程”节点，查看 student\_db 数据库中是否已创建了存储过程 selectScore。

单击工具栏中的“新建查询”按钮，右侧“查询”窗格中使用 EXECUTE 命令执行存储过程 selectScore，并观察存储过程的输出结果是否如图 11.1 所示。



	St_ID	St_Name	C_Name	Score
1	0603060111	曾莉娟	C++程序设计基础	99
2	2602060106	王小维	体育	97
3	2001050106	金萍	大学计算机基础实践	93
4	2602060107	刘小玲	体育	92
5	2001050105	邓红艳	大学计算机基础实践	90

图 11.1 存储过程 selectScore 执行结果

2. 以下代码使用 T-SQL 语句创建一个名为 studentScore 的带输入参数的存储过程，实现查询指定学号的学生所选修课程的成绩。

```
CREATE PROCEDURE studentScore @stuID varchar(20) AS
SELECT a.St_Name, b.C_No, b.C_Name, c.Score
FROM St_Info a, C_Info b, S_C_Info c
WHERE a.St_ID = c.St_ID AND b.C_No = c.C_No AND a.St_ID = @stuID
```

单击工具栏中的“执行”按钮完成存储过程的创建。如果创建成功，在新建“查询”窗格中输入以下包含系统存储过程的命令序列：

```

sp_help studentScore
GO
sp_helptext studentScore

```

观察系统存储过程 `sp_help` 和 `sp_helptext` 输出的不同结果, 理解系统存储过程的作用。

再打开新的“查询”窗格, 使用 `EXECUTE` 命令执行存储过程 `studentScore`, 并分别以参数值“2001050105”、“0603060109”、“2602060100”作为输入参数, 观察存储过程的输出结果。

如果存储过程 `studentScore` 执行时没有提供参数, 则按默认值查询 (假设默认值为空字符串, 表示查询所有学号的学生), 如何修改该存储过程的定义?

3. 利用 T-SQL 语句创建对表 `C_Info` 进行插入、修改和删除操作的 3 个存储过程: `insertCInfo`、`updateCInfo`、`deleteCInfo`, 实现以下功能:

`insertCInfo`: 将所有字段作为存储过程的输入参数, 插入一条新记录。

`updateCInfo`: 将所有字段作为存储过程的输入参数, 按课程编号 `C_No` 修改课程内容。

`deleteCInfo`: 将课程编号 `C_No` 作为存储过程的输入参数, 删除该课程记录。

4. 使用 `Student_db` 数据库中的 `St_Info` 表、`C_Info` 表、`D_Info` 表, 完成以下要求:

(1) 创建一个存储过程 `getPractise`, 查询指定院系 (名称) 中参与“实践”课程学习的所有学生学号、姓名、所学课程编号和课程名称。

**提示:** `D_Info` 表中存储了院系代码和名称, `St_Info` 表中的学号字段 `St_ID` 的前两位与之对应; 提取 `St_ID` 的前两位字符可以使用 T-SQL 中的 `LEFT` 函数 (具体用法通过查询设计器进行查阅或参看有关触发器的实验内容)。

(2) 分别执行存储过程 `getPractise`, 查询“法学院”和“材料科学与工程学院”的学生参与“实践”课程学习的所有学生学号、姓名、所学课程编号和课程名称。

(3) 利用系统存储过程 `sp_rename` 将 `getPractise` 更名为 `getPctStu`。

(4) 修改存储过程 `getPctStu`, 返回指定院系中参与“实践”课程学习的学生人次数, 并利用不同的输入参数验证存储过程执行的结果。

更进一步, 如果希望让存储过程 `getPctStu` 返回学生的人数, 那应该如何修改存储过程呢? 需要注意的是, “人数”与“人次数”是两个概念, 对于某一个学生而言, 如果参与了多门实践课程, 则“人次数”是其课程门数, 而“人数”仍然等于 1。

(5) 修改存储过程 `getPctStu`, 实现如果输入的院系并不存在, 提示相应的信息并返回过程的状态码值等于 -1, 否则返回过程的状态码值等于 0。

(6) 使用系统存储过程 `sp_helptext` 查看存储过程的定义文本。

(7) 复制 `getPctStu` 的定义, 产生一个加密的新的存储过程——`getPctStuTemp`, 并使用 `sp_helptext` 查看该存储过程的定义, 观察显示结果。

(8) 使用 T-SQL 语句 `DROP PROCEDURE` 删除存储过程 `getPctStuTemp`。

#### 四、实验思考

1. 什么是存储过程? 使用存储过程有什么好处?
2. 怎么给存储过程赋予使用授权?
3. 如果一个存储过程的定义需要修改但又不希望先删除它, 应使用何种方式操作?