

# 重庆城市职业学院 一体化课程项目工作任务单

课程名称： 网络数据库管理与应用

适用专业： 计算机应用技术

软件与信息服务

物联网应用技术

指导教师： 刘 洪 宾



## 一体化课程项目工作任务单

|  |   |    |         |
|--|---|----|---------|
| 名 称  | 安装 SQL Server 2008 数据管理系统   | 序号 | 1       |
| 目 标  | 1. 了解 SQL Server 2008 的主要版本<br>2. 熟悉 SQL Server 2008 安装所需的软、硬件条件<br>3. 熟悉 SQL Server 2008 安装的方法<br>4. 具备 SQL Server 2008 数据库服务器的安装的能力<br>5. 会启动 SQL Server 2008 的相关服务 |    |         |
| 工作要求   | 1. 安装之前应掌握 SQL Server 2008 数据库服务器的基础知识<br>2. 完成 SQL Server 2008 数据库服务器的安装并验证<br>3. 会启动 SQL Server 2008 的相关服务<br>4. 根据工作过程写出实验报告                                       |    |         |
| 企业标准   | 参照国家标准及企业相关要求   |    |         |
| 工作条件   | 1. 硬件条件：CPU4 核，内存 4G，硬盘空间 20G 以上<br>2. 操作系统： Windows 7<br>3. 虚拟软件：VMware Workstation 8.0 以上<br>4. Windows Enterprise Server 2003 虚拟服务器一台                              |    |         |
| 工 作 流 程  |   |    | 注 意 事 项 |
| <p><b>一、工作背景</b></p> <p>abc 学校以前由于规模小，学生人数少，学生管理采用的是传统的 Excel 模式，随着学校规模的增大，学生人数的增加，传统的 Excel 管理已经不能满足学校日益发展和管理的需要，为此，学校信息中心提出通过建立企业数据库系统以管理学生的数据。作为信息中心的网络管理员，需要根据学校的具体情况，在现有的 Windows Server 服务器中安装并配置一台数据库服务器，以便更好的管理学生数据。</p> <p><b>二、工作内容</b></p> <p>任务一：安装 SQL Server 2008 数据库服务器所需的软件环境<br/>                     任务二：安装 SQL Server 2008 数据库服务器并验证<br/>                     任务三：启动 SQL Server 2008 的相关服务</p> <p>1. 启动 SQL Server Management Studio<br/>                     2. 通过 Windows 服务启动或停止 SQL Server 服务<br/>                     3. 通过 SQL Server 配置管理器来启动或停止服务</p> |   |    |         |

### 三、工作流程

任务一：安装 SQL Server 2008 数据库服务器所需的软件环境

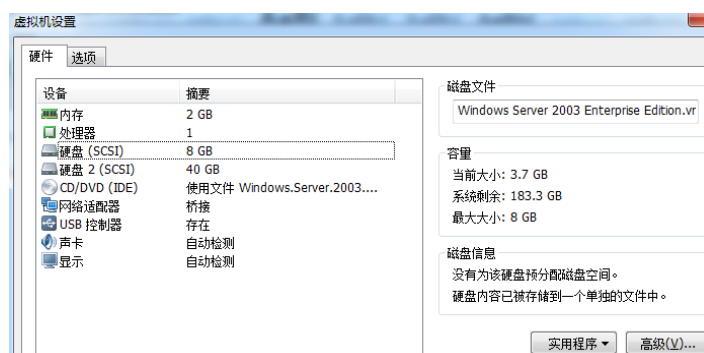
#### 1. 增加虚拟机的磁盘的空间

第一步：在 Windows Server 2003 Enterprise Edition 中，选择“编辑虚拟机设置。”

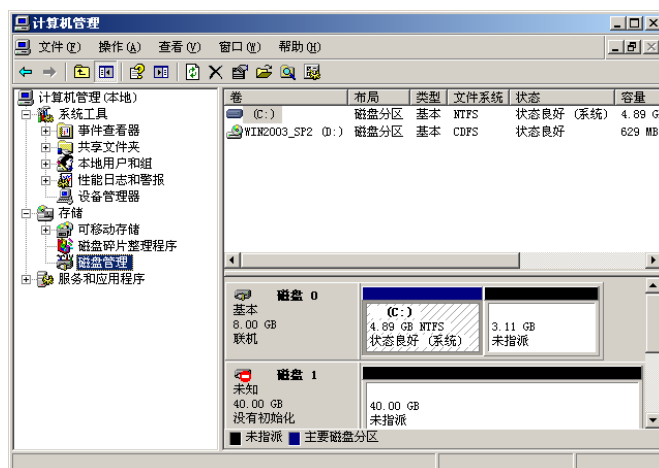


第二步：在“虚拟机设置”窗口中，选择“硬盘 SISC”，单击“添加”按钮，添加硬盘。

第三步：在“添加硬件”向导中，根据提示完成，硬盘的添加，添加完成后，如下图所示。



第四步：在虚拟机 Windows Server 2003 中，选择“硬盘管理”工具，添加的硬盘，并格式化。



第五步：完成磁盘的添加。

## 2. 安装 .net framework 3.5

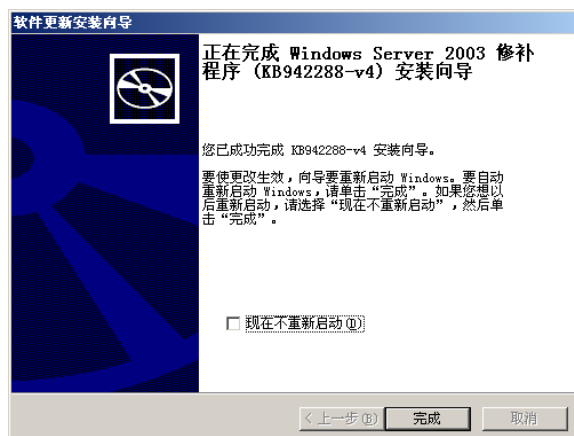
第一步：双击 dotnetfx35.exe，在 .net framework 3.5 安装窗口中，选择“我已经阅读并接受许可协议中的条款(A)”，单击“安装”按钮，安装 .net framework 3.5。



第二步：根据提示完成 .net framework 3.5 的安装。

## 3. 安装 Windows Installer

第一步：单击 WindowsServer2003-KB942288-v4-x86.exe，根据提示完成 Windows Installer



第二步：安装完成后，重启服务器。

## 任务二： 安装 SQL Server 2008 数据库服务器并验证

第一步：双击 SQL Server 2008 安装盘中的 setup.exe，打开如图所示的“SQL Server 安装中心”。

第二步：在“SQL Server 安装中心”窗口中，单击左侧窗口的“安装”选项。

第三步：在“SQL Server 安装中心”窗口中，单击右侧窗口的“全新 SQL Server 独立安装或向现有安装添加功能”，安装程序将

进行 SQL Server 安装程序及支持规则的检查，以避免安装过程中出现问题，单击“确定”按钮。

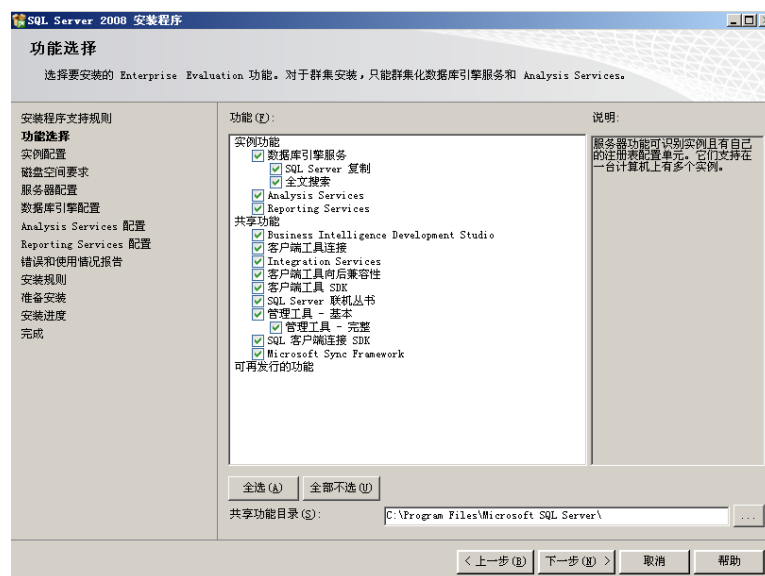
第四步：在“产品密钥”窗口中，输入已购买 SQL Server 产品的密钥，安装程序会根据你输入的产品密钥来确定将要安装的程序版本，如果没有产品密钥，可以单击“指定本用版本”，并从下拉列表框中选择“Enterprise Evaluation”或“Express”等版本，然后单击“下一步”按钮。

第五步：在“许可条款”窗口中阅读软件许可条款，并单击“我接受许可条款”复选框，单击“下一步”按钮。

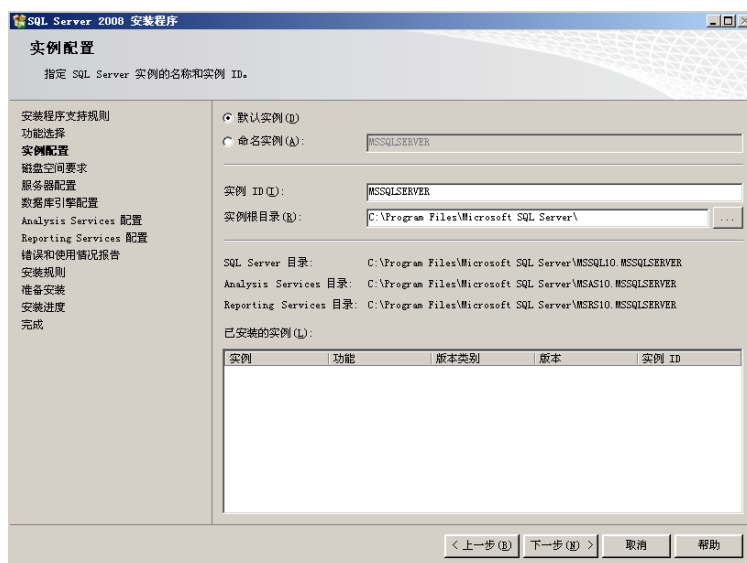
第六步：在“安装程序支持文件”窗口中，单击“安装”按钮，安装程序支持的文件。

第七步：安装程序支持的文件完成后，软件将再次进行 SQL Server 安装程序及支持规则的检查，可以单击“显示详细信息”来查看安装程序所支持文件的通过情况，单击“下一步”按钮。

第八步：在“功能选择”窗口中，选择要安装的 SQL Server 组件，单击“全选”按钮，全部安装。

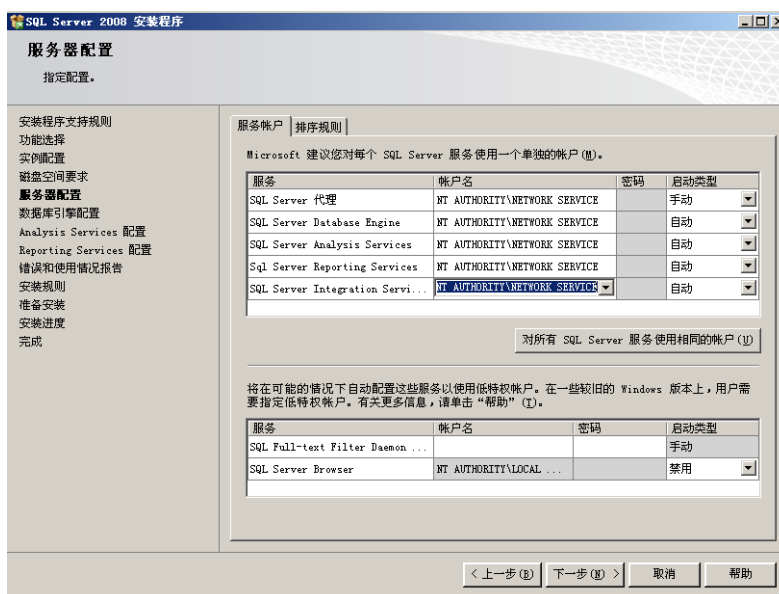


第九步：单击“下一步”按钮，在弹出的“实例配置”窗口中，选择“默认实例”，单击“下一步”按钮。

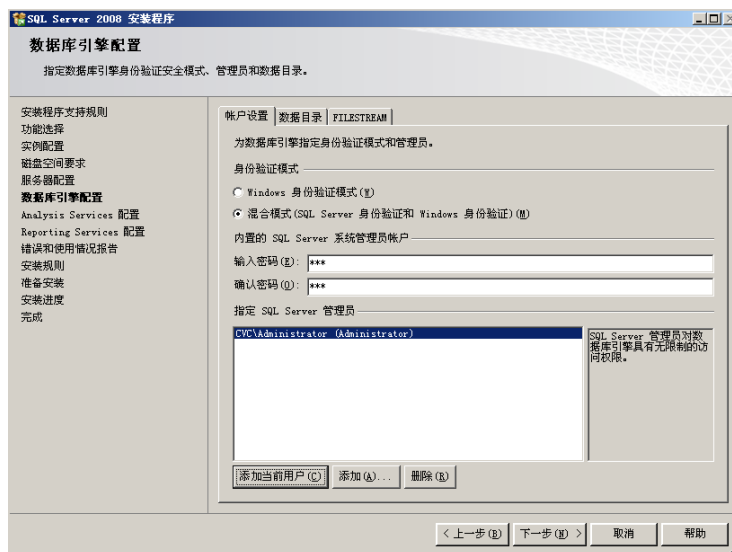


第十步：在“磁盘空间要求”窗口中，显示了安装 SQL Server 所需的磁盘空间信息，如果所需的磁盘空间不够，可以将共享安装目录和实例目录安装到其它磁盘中，单击“下一步”按钮。

第十一步：在“服务器配置”窗口中，单击“服务账户”，设置 SQL Server 服务的登录账户和密码，可以为每个服务分别设置账户和密码，也可以单击“对所有 SQL Server 服务使用相同的账户”，设置账户名和密码。在这里对所有的服务均使用相同的 NETWORK SERVICE 账户，分别在帐户名的下拉列表中选择“NETWORK SERVICE 账户”，单击“下一步”按钮。



第十二步：在“数据库引擎配置”窗口中，对 SQL Server 的帐户的身份验证模式进行设置，SQL Server 支持两种身份验证模式：windows 身份验证模式和混合模式，在里选择混合模式，并设置内置 SQL Server 系统管理员的密码，单击“添加当前用户”将当前的 Windows 账户添加为 SQL Server 管理员，单击“下一步”按钮。



第十三步：在“Analysis Services 配置”窗口中，单击“添加当前用户”，指定当前的 Windows 账户对 Analysis Services 具有管理权限，单击“下一步”按钮。

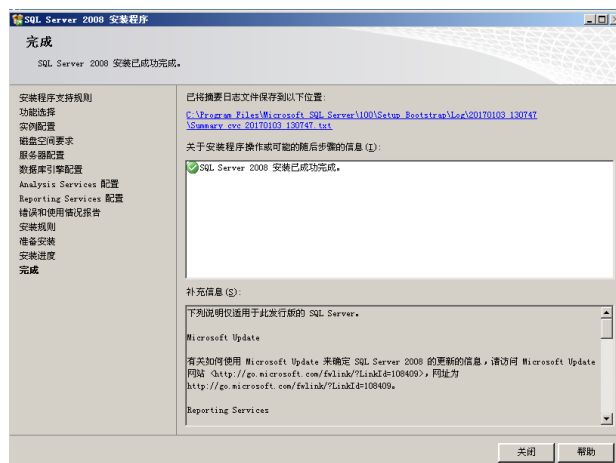
第十四步：在“Reporting Services 配置”窗口中，指定要创建的 Reporting Services 安装类型，单击选择“安装本机模式默认配置”单击按钮，单击“下一步”按钮。

第十五步：在“安装规则”窗口中，系统配置检查器再次运行一组规则来对用户指定的 SQL Server 功能验证计算机配置，可能单击“显示详细信息”进行查看，全部验证通过以后，单击“下一步”按钮。

第十六步：在“准备安装”窗口中，显示了用户在安装过程中指定的安装选项及功能的树型视图，单击“安装”按钮进行安装。

第十七步：在“安装进度”窗口中，显示了正在执行的安装状态。

第十八步：安装过程完成后，在“安装进度”窗口中显示安装组件的状态，单击“关闭”，按钮，完成 SQL Server 2008 的安装。





第十九步：验证 SQL Server 2008 的安装是否成功

依次单击“开始”→“所有程序”→“Microsoft SQL Server 2008”，如果可以正常启动相关程序，则安装成功。

任务三：启动 SQL Server 2008 的相关服务

1. 启动 SQL Server Management Studio

依次单击“开始”→“所有程序”→“Microsoft SQL Server 2008”→“SQL Server Management Studio”，在打开 SQL Server Management Studio 窗口，选择身份验证模式。

第一步：Windows 身份验证

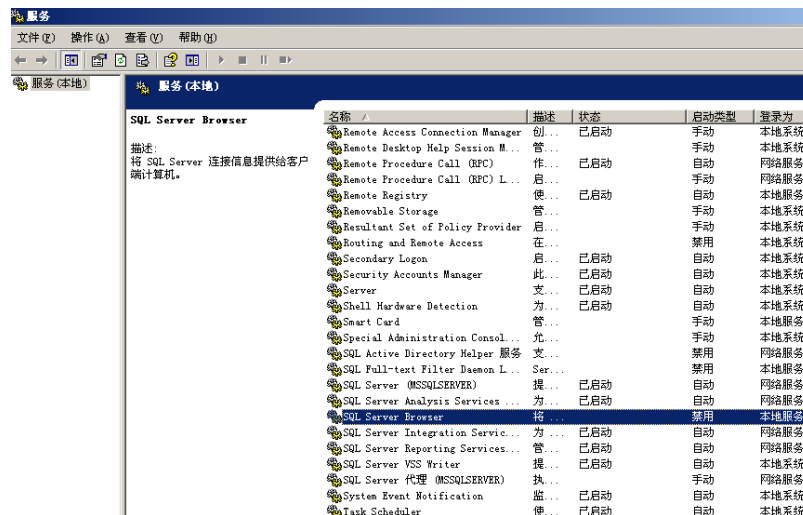
在“连接到服务器”窗口中，服务器类型选择“数据库引擎”，服务器名输入“.”，身份验证选择“Windows 身份验证”，单击“连接”按钮。

第二步：SQL Server 身份验证

在“连接到服务器”窗口中，服务器类型选择“数据库引擎”，服务器名输入“.”，身份验证选择“SQL Server 身份验证”，在登录名文本框中输入“sa”，在密码文本框中输入“123”，单击“连接”按钮。

2. 通过 Windows 服务启动或停止 SQL Server 服务

依次单击“开始”→“控制面板”→“管理工具”→“服务”，在“服务”窗口中启动或停止 SQL Server 服务。



3. 通过 SQL Server 配置管理器来启动或停止服务

依次单击“开始”→“所有程序”→“SQL Server2008”→“配置工具”→“SQL Server 配置管理器”，在 SQL Server 配置管理器口中启动或停止 SQL Server 服务。



### 考核标准（技能和素质考核）

#### 1、专业技能考核标准（共 80 分）

- (1) 正确安装 SQL Server 2008 数据库服务器所需的软件环境；（20 分）
- (2) 正确安装 SQL Server 2008 数据库服务器并验证成功；（40 分）
- (3) 会启动 SQL Server 2008 的相关服务；（20 分）

#### 2、学习态度考核标准（共 20 分）

- (1) 实训报告书质量（5 分）

认真对实训过程进行总结，发现和解决问题；认真按照要求填写；书面整洁，字迹清楚。

- (2) 工作职业道德（5 分）

安全文明工作，职业操守较好；爱护计算机等公共设施；基本按照布置的工作任务和要求去完成。

- (3) 团队合作精神（5 分）

具有良好的团队合作精神，热心帮助班级或小组的其它成员；能与团队成员有效沟通；能合理分配小组成员的工作任务；

- (4) 语言沟通能力（5 分）

能用专业的语言正确流利地展示项目成果；能准确回答老师提出的相关问题

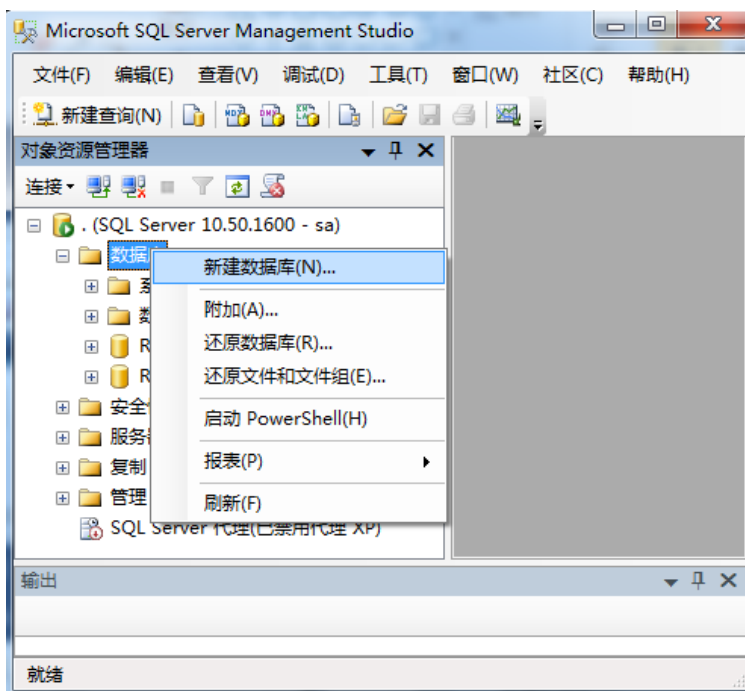
### 问题与思考

1. SQL Server 2008 有哪些版本。
2. 如何启动、暂停和停止 SQL Server 2008 的服务。

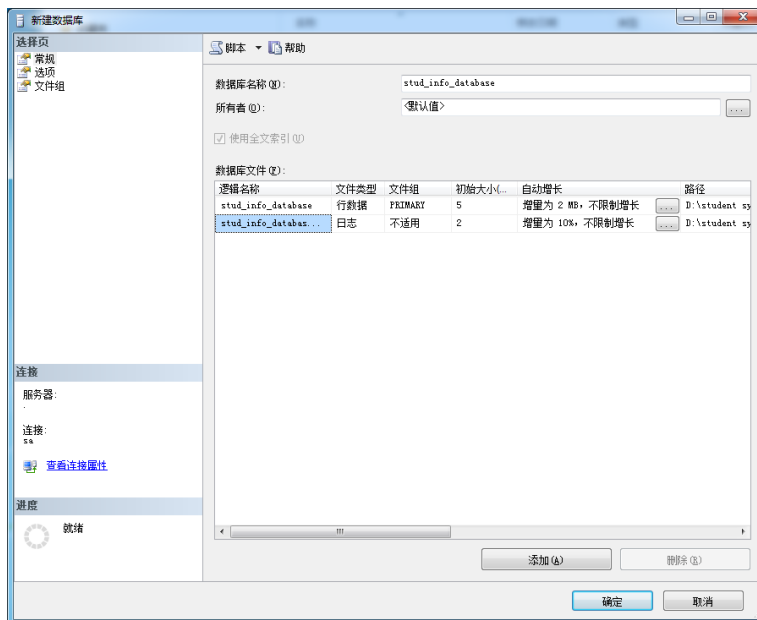
## 一体化课程项目工作任务单

|  |  |    |         |
|--|--|----|---------|
| 名 称  | 数据库的创建与管理  | 序号 | 2       |
| 目 标  | 1. 具备数据库创建的能力<br>2. 具备数据库修改的能力<br>3. 会删除数据库文件<br>4. 具备数据库分离与附加的能力              |    |         |
| 工作要求   | 1. 完成数据库的创建<br>2. 完成数据库的修改<br>3. 完成数据库的删除<br>4. 完成数据库的分离和附加<br>5. 根据工作过程写出实验报告 |    |         |
| 企业标准   | 1. 数据库对象的命名应该符合企业规范<br>2. SQL 代码必须符合基本编程规范                                     |    |         |
| 工作条件   | 1. 操作系统: Windows 7<br>2. 软件环境: SQL Server 2008 数据库                             |    |         |
| 工 作 流 程  |  |    | 注 意 事 项 |
| <p><b>一、工作背景</b></p> <p>abc 学校已经安装好了一台 SQL Server 服务器, 信息中心经过前期数据库的需求分析和概要设计, 选择在 SQL Server2008 上建立一个“学生信息数据库”。</p> <p><b>二、工作内容</b></p> <p>任务一: 数据库的创建</p> <p>1. 使用 SQL Server Managemet Studio 创建数据库 stud_info_database</p> <p>要求如下:</p> <p>(1) 数据库名: stud_info_database。</p> <p>(2) 数据库文件存储: d:\student system\</p> <p>(3) 数据库文件的初始大小为 5M, 自动增长, 增量为 2M</p> <p>(4) 数据库日志文件的初始大小为 2M, 自动增长, 增量为 10%</p> <p>(5) 设置数据库为自动收缩</p> <p>2. 使用 sql 语句创建数据库 stud_info_database2</p> <p>要求如下:</p> <p>(1) 数据库文件的初始大小为 5M, 自动增长, 增量为 2M</p> |  |    |         |

|   |  |
|---|--|
| <p>(2) 数据库日志文件的初始大小为 2M, 自动增长, 增量为 10%</p> <p>(3) 数据库文件存储: d:\student system</p> <p>任务二: 数据库的修改</p> <p>1. 使用 SQL Server Managemet Studio 将数据库 stud_info_database2 的所有者修改为本机管理员, 并设置为单用户模式</p> <p>2. 使用 SQL 语名, 将数据库 stud_info_database2 重命名为 stud_data, 并增加一个数据库文件和一个日志文件至 stud_temp 数据库</p> <p>3. 使用 SQL 语名, 增加一个数据库文件和一个日志文件至 stud_temp 数据库中, (存储路径为: d:\student system, 初始大小为 2, 最大为 100M, 每次增量为 2M)</p> <p>任务三: 数据库的删除</p> <p>1. 使用 SQL Server Management Studio 删除数据库 stud_data</p> <p>2. 使用 SQL 语句删除数据库 stud_data</p> <p>任务四: 数据库的分离和附加</p> <p>1. 分离 stud_info_database 数据库</p> <p>2. 附加 stud_info_database 数据库</p> <p><b>三、工作流程</b></p> <p><b>任务一: 数据库的创建</b></p> <p>1. 使用 SQL Server Managemet Studio 创建数据库 stud_info_database</p> <p>要求如下:</p> <p>(1) 数据库名: stud_info_database。</p> <p>(2) 数据库文件存储: d:\student system\</p> <p>(3) 数据库文件的初始大小为 5M, 自动增长, 增量为 2M</p> <p>(4) 数据库日志文件的初始大小为 2M, 自动增长, 增量为 10%</p> <p>(5) 设置数据库为自动收缩</p> <p>第一步: 依次单击“开始”→“所有程序”→“ Microsoft SQL Server 2008” → “SQL Server Management Studio”, 在打开 SQL Server Management Studio 窗口, 选择“SQL Server 身份验证”, 在登录名文本框中输入“sa”, 在密码文本框中输入“123”, 单击“连接”按钮。</p> <p>第二步: 在 SQL Server Management Studio 的“对象资源管理器”中, 用鼠标右键单击“数据库”节点, 在弹出的快捷菜单中选择“新建数据库命令”。</p> |  |
|---|--|



第三步：在“新建数据库”窗口中，设置数据库名为：stud\_info\_database，数据库的初始大小为数据库文件的初始大小为 5M，并设置增量为 2M， unlimited growth，数据库日志文件的初始大小为 2M，增量为 10%， unlimited growth；并设置数据库保存的位置为：d:\student system。



第四步：单击“选项”卡，将数据库的自动收缩设置为 True。

第五步：单击“确定”按钮，完成数据库的创建。

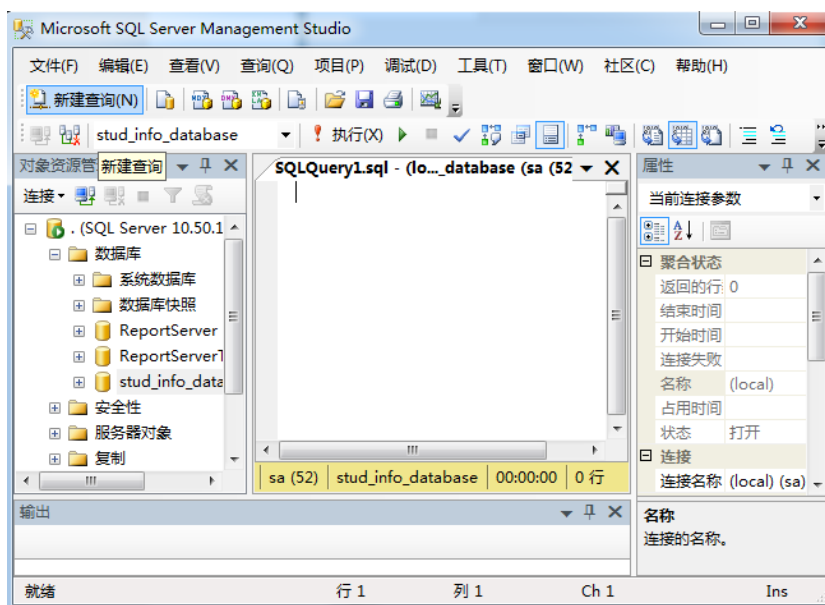
2. 使用 sql 语句创建数据库 stud\_info\_database2

要求如下：

- (1) 数据库文件的初始大小为 5M，自动增长，增量为 2M
- (2) 数据库日志文件的初始大小为 2M，自动增长，增量为 10%

(3) 数据库文件存储: d:\student system

第一步: 在打开的“SQL Server Management Studio”窗口中, 单击“新建查询”按钮。



第二步: 在打开的查询分析器中输入以下 T-SQL 语句。

```
create database stud_info_database2
on primary
(
name='stud_info_database2',
filename='D:\student system\stud_info_database2.mdf',
size=5MB,
maxsize=unlimited,
filegrowth=2MB
)
log on
(
name='stud_info_database2_log',
filename='D:\student system\ stud_info_database_log2.ldf',
size=2MB,
maxsize= unlimited,
filegrowth=10%
)
```

第三步: 查看数据库的信息。

在 SQL Server Management Studio 的“对象资源管理器”中, 用鼠标右键单击“stud\_info\_database2”节点, 在弹出的快捷菜单中选择“属性”。

在常规选项卡中可以看到数据库 stud\_info\_database2 的基本信息。

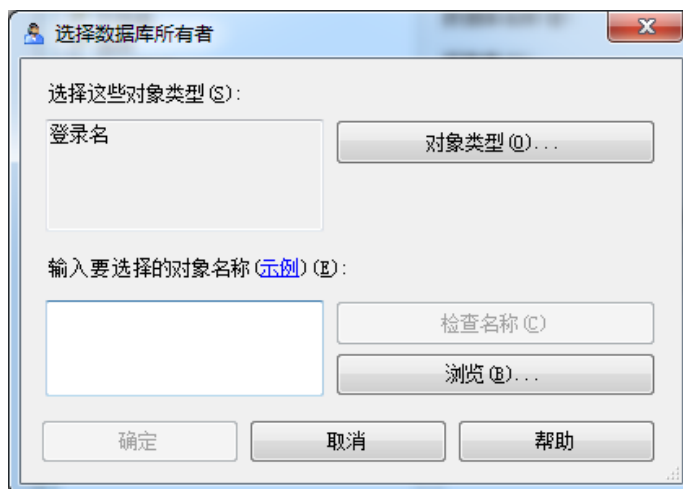
**任务二: 数据库的修改**

1. 使用 SQL Server Managemet Studio 将数据库 stud\_info\_database2 的所有者修改为本机管理员,并设置为单用户模式

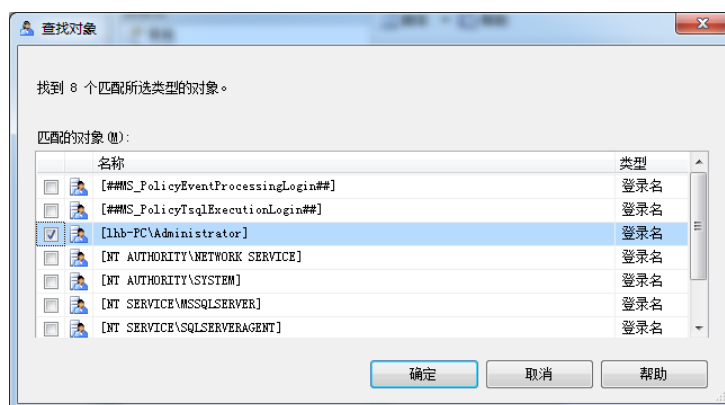
第一步: 在 SQL Server Management Studio 的“对象资源管理器”中,用鼠标右键单击“stud\_info\_database2”节点,在弹出的快捷菜单中选择“属性”命令。

第二步: 在“数据库属性—stud\_info\_database2”窗口中,单击“文件”选项,进入文件设置页面,单击“所有者”文本框后面的浏览按钮。

第三步: 在“选择数据库所有者”窗口中,单击“浏览”按钮。



第四步: 在“查找对象”窗口中,选择本机的管理员,本例为“lhb-PC\Administrator”,单击“确定”按钮,再单击“确定”按钮。



第五步: 单击“选项”,进入选项设置页面,将“限制访问”设置为“SINGLE\_USER”。



第六步: 单击“确定”按钮,完成数据库 stud\_info\_database2

的修改。

2. 使用 SQL 语句，将数据库 stud\_info\_database2 重命名为 stud\_data，并增加一个数据库文件和一个日志文件至 stud\_temp 数据库。

SQL语句：

```
alter database stud_info_database2  
modify name=stud_data
```

3. 使用 SQL 语句，增加一个数据库文件和一个日志文件至 stud\_temp 数据库中，（存储路径为：d:\student system，初始大小为 2，最大为 100M，每次增量为 2M）

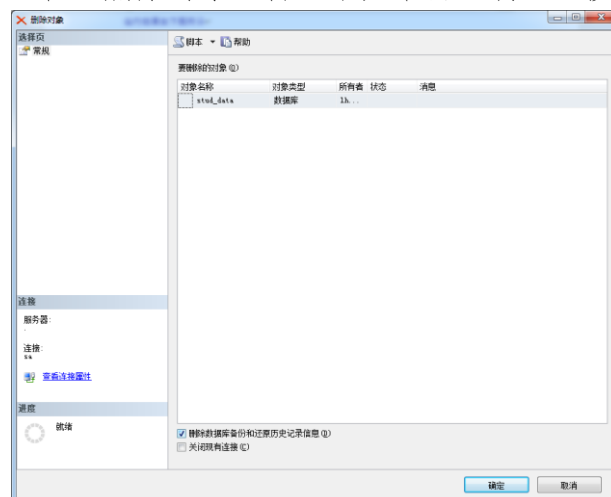
SQL 语句，略

#### 任务四：数据库的删除

1. 使用 SQL Server Management Studio 删除数据库 stud\_data

第一步：在 SQL Server Management Studio 的“对象资源管理器”中，用鼠标右键单击“stud\_data”节点，在弹出的快捷菜单中选择“删除”命令。

第二步：在“删除对象”窗口中，单击“确定”按钮。



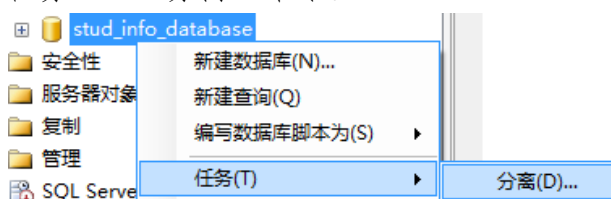
2. 使用 SQL 语句删除数据库 stud\_data

```
drop database stud_data
```

#### 任务四：数据库的分离和附加

1. 分离 stud\_info\_database 数据库

第一步：在 SQL Server Management Studio 的“对象资源管理器”中，用鼠标右键单击“stud\_data”节点，在弹出的快捷菜单中依次选择“任务”→“分离”命令。



第二步：在打开的“分离数据库”窗口中，单击“确定”按钮，

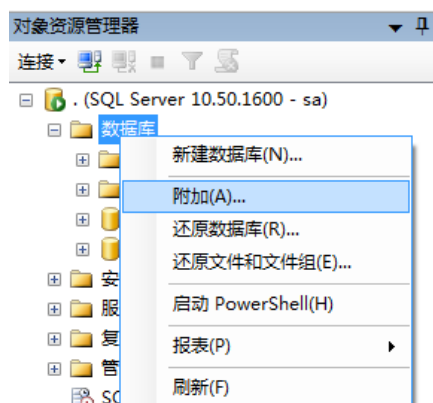


将数据库分离出去。

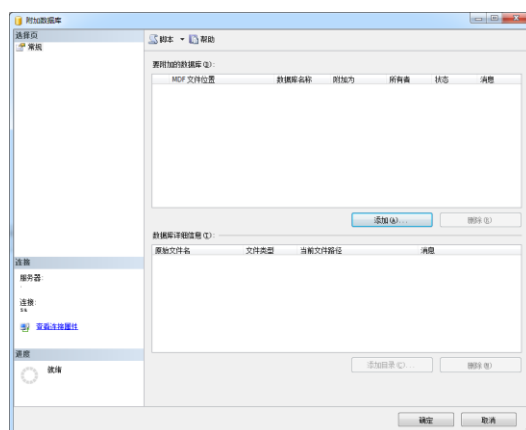
**温馨提示：**同学们每次做完项以后，都可以将数据库从数据库服务器中的分离出去，然后拷贝至自己的U盘中。

## 2. 附加 stud\_info\_database 数据库

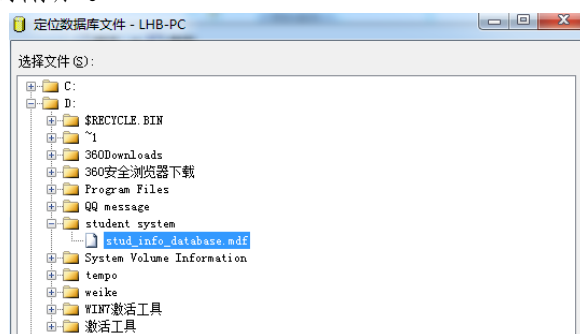
第一步：在 SQL Server Management Studio 的“对象资源管理器”中，用鼠标右键单击“数据库”节点，在弹出的快捷菜单中选择“附加”命令。



第二步：在“附加数据库”窗口中，单击“添加”按钮。



第三步：在“定位数据库文件”窗口中，打开数据库所在的路径，并选择该数据库的.mdf，（在本例中选择的路径为数据库为：D:\student system\stud\_info\_database.mdf）单击“确定”按钮，完成数据库的附加。



| 考核标准（技能和素质考核）   |
|---|
| <p><b>1、专业技能考核标准（共 80 分）</b></p> <p>（1）正确创建数据库；（25 分）</p> <p>（2）正确修改数据库；（25 分）</p> <p>（3）正确删除数据库；（10 分）</p> <p>（4）正确的分离和附加数据库（20）</p> <p><b>2、学习态度考核标准（共 20 分）</b></p> <p>（1）实训报告书质量（5 分）</p> <p>认真对实训过程进行总结，发现和解决问题；认真按照要求填写；书面整洁，字迹清楚。</p> <p>（2）工作职业道德（5 分）</p> <p>安全文明工作，职业操守较好；爱护计算机等公共设施；基本按照布置的工作任务和要求去完成。</p> <p>（3）团队合作精神（5 分）</p> <p>具有良好的团队合作精神，热心帮助班级或小组的其它成员；能与团队成员有效沟通；能合理分配小组成员的工作任务；</p> <p>（4）语言沟通能力（5 分）</p> <p>能用专业的语言正确流利地展示项目成果；能准确回答老师提出的相关问题</p> |
| 问题与思考   |
| <p>1. 创建、修改与删除数据库的 SQL 语句分别是什么？</p> <p>2. 根据下列要求写出创建数据库的 SQL 语句（代码附于工作报告中）</p> <p>（1）数据库名：stud。</p> <p>（2）数据库文件存储位置：d:\student\</p> <p>（3）数据库文件的初始大小为 3M，自动增长，增量为 1M</p> <p>（4）数据库日志文件的初始大小为 1M，自动增长，增量为 5%</p> <p>3. 使用 SQL 语名,增加一个数据库文件和一个日志文件至 stud_temp 数据库中，（存储路径为：d:\student system，初始大小为 2，最大为 100M，每次增量为 2M）（代码附于工作报告中）。</p>  |

## 一体化课程项目工作任务单

| 名 称  | 数据表的创建与管理   | 序号       | 3                |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
|--|---|----------|------------------|--------------|--|--|-----|------|----|-----|------------|---------|-----|------------|-------|-----|---------------|---------|-----|------------|-------|-----|-----------|-------|-----|-------------|----------|-----|------------|---------|-----|-------------|----------|-----|-----------|-------|-----------------------|
| 目 标  | 1. 具备数据表创建的能力<br>2. 具备数据表修改的能力<br>3. 会管理表中的数据                       |          |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 工作要求   | 1. 完成数据表的创建<br>2. 完成数据表的修改<br>3. 完成表中数据的插入、修改和删除<br>4. 根据工作过程写出实验报告 |          |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 企业标准   | 1. 数据表对象命名应该符合企业规范<br>2. 数据表中避免可空列<br>3. SQL 代码必须符合基本编程规范           |          |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 工作条件   | 1.操作系统： Windows 7<br>2.软件环境：SQL Server 2008 数据库                     |          |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 工作流程   |   |          | 注<br>意<br>事<br>项 |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 一、工作背景<br><br>abc 学校的信息中心在完成数据库的创建以后，接下来需要根据数据库表概念结构、逻辑结构的设计完成数据表的创建和管理，并管理表里面的数据。<br><br>二、工作内容<br><br>任务一：数据表的创建<br>使用 SQL 语句或 SQL Server Managemet Studio 创建表以下表<br><table><tr><th colspan="3">表 1.1 数据表及说明</th></tr><tr><th>实体名</th><th>数据库名</th><th>说明</th></tr><tr><td>系部表</td><td>Depart_tab</td><td>系部基本信息表</td></tr><tr><td>民族表</td><td>Nation_tab</td><td>民族信息表</td></tr><tr><td>专业表</td><td>Specialty_tab</td><td>专业基本信息表</td></tr><tr><td>性别表</td><td>gender_tab</td><td>学生的性别</td></tr><tr><td>职称表</td><td>title_tab</td><td>教师职称表</td></tr><tr><td>教师表</td><td>Teacher_tab</td><td>教师的基本信息表</td></tr><tr><td>课程表</td><td>Course_tab</td><td>课程基本信息表</td></tr><tr><td>学生表</td><td>Student_tab</td><td>学生的基本信息表</td></tr><tr><td>成绩表</td><td>Score_tab</td><td>学生成绩表</td></tr></table><br><table><tr><th colspan="2">表 1.2 系部表（Depart_tab）</th></tr></table> |   |          |                  | 表 1.1 数据表及说明 |  |  | 实体名 | 数据库名 | 说明 | 系部表 | Depart_tab | 系部基本信息表 | 民族表 | Nation_tab | 民族信息表 | 专业表 | Specialty_tab | 专业基本信息表 | 性别表 | gender_tab | 学生的性别 | 职称表 | title_tab | 教师职称表 | 教师表 | Teacher_tab | 教师的基本信息表 | 课程表 | Course_tab | 课程基本信息表 | 学生表 | Student_tab | 学生的基本信息表 | 成绩表 | Score_tab | 学生成绩表 | 表 1.2 系部表（Depart_tab） |
| 表 1.1 数据表及说明   |   |          |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 实体名  | 数据库名  | 说明       |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 系部表  | Depart_tab  | 系部基本信息表  |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 民族表  | Nation_tab  | 民族信息表    |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 专业表  | Specialty_tab   | 专业基本信息表  |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 性别表  | gender_tab  | 学生的性别    |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 职称表  | title_tab   | 教师职称表    |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 教师表  | Teacher_tab   | 教师的基本信息表 |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 课程表  | Course_tab  | 课程基本信息表  |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 学生表  | Student_tab   | 学生的基本信息表 |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 成绩表  | Score_tab   | 学生成绩表    |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |
| 表 1.2 系部表（Depart_tab）  |   |          |                  |              |  |  |     |      |    |     |            |         |     |            |       |     |               |         |     |            |       |     |           |       |     |             |          |     |            |         |     |             |          |     |           |       |                       |

| 列名          | 数据类型        | 可否为空     | 说明       |
|-------------|-------------|----------|----------|
| id          | Int         | Not null | 编号（递增 1） |
| depart_id   | Varchar（2）  | Not null | 系部号（主键）  |
| Depart_name | Varchar（20） | Not null | 系部名      |

表 1.3 民族表（Nation\_tab）

| 列名          | 数据类型        | 可否为空     | 说明       |
|-------------|-------------|----------|----------|
| id          | Int         | Not null | 编号（递增 1） |
| Nation_id   | Varchar（2）  | Not null | 民族编号     |
| Nation_name | Varchar（12） | Not null | 民族名称     |

表 1.4 专业表（Specialty\_tab）

| 列名             | 数据类型        | 可否为空     | 说明       |
|----------------|-------------|----------|----------|
| id             | Int         | Not null | 编号（递增 1） |
| Specialty_id   | Varchar（3）  | Not null | 专业编号     |
| Specialty_name | Varchar（30） | Not null | 专业名称     |

表 1.5 性别表（gender\_tab）

| 列名          | 数据类型       | 可否为空     | 说明       |
|-------------|------------|----------|----------|
| id          | Int        | Not null | 编号（递增 1） |
| gender_id   | Varchar（1） | Not null | 性别编号     |
| gender_name | Varchar（2） | Not null | 性别       |

表 1.6 职称表（title\_tab）

| 列名         | 数据类型        | 可否为空     | 说明       |
|------------|-------------|----------|----------|
| id         | Int         | Not null | 编号（递增 1） |
| title_id   | Varchar（2）  | Not null | 职称编号     |
| title_name | Varchar（20） | Not null | 职称       |

表 1.7 教师表（teacher\_tab）

| 列名           | 数据类型        | 可否为空     | 说明             |
|--------------|-------------|----------|----------------|
| id           | Int         | Not null | 编号（递增 1）       |
| teacher_id   | Varchar（20） | Not null | 教师号（主键）        |
| Teacher_name | Varchar（10） | Not null | 教师名            |
| gender_id    | Varchar（1）  | Not null | 性别（外键，参照性别表）   |
| Nation_id    | Varchar（2）  | Not null | 民族（外键，参照民族表）   |
| Title_id     | Varchar（2）  | Not null | 职称编号（外键，参照职称表） |
| Depart_id    | Varchar（2）  | Not null | 所在系（外键，参照系部表）  |

表 1.8 课程表（Course\_tab）

| 列名           | 数据类型        | 可否为空     | 说明             |
|--------------|-------------|----------|----------------|
| id           | Int         | Not null | 编号（递增 1）       |
| course_id    | Varchar（10） | Not null | 课程号（主键）        |
| Course_name  | Varchar（20） | Not null | 课程名            |
| Course_xizi  | Varchar（12） |          | 课程性质           |
| teacher_id   | Varchar（20） | Not null | 主讲教师（外键，参照教师表） |
| term         | Varchar（10） |          | 开课学期           |
| First_course | Varchar（10） |          | 先修课（空表示无先修课）   |
| Depart_id    | Varchar（2）  |          | 开课系部（外键，参照系部表） |
| Class_hour   | Varchar（3）  | Not null | 学时             |

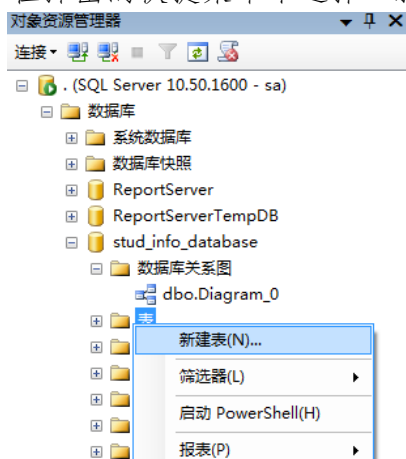
|   |              |          |                 |
|---|--------------|----------|-----------------|
| credits   | Varchar (3)  | Not null | 学分              |
| note  | text         |          | 备注              |
| 表 1.9 学生信息表 (Student_tab)                         |              |          |                 |
| 列名  | 数据类型         | 可否为空     | 说明              |
| id  | Int          | Not null | 编号 (递增 1)       |
| Stud_id   | Varchar (15) | Not null | 学号 (主键)         |
| Stud_name   | Varchar (10) | Not null | 姓名              |
| gender_id   | Varchar (1)  | Not null | 性别 (外键, 参照性别表)  |
| Nation_id   | Varchar (2)  | Not null | 民族 (外键, 参照民族表)  |
| brithday  | date         |          | 出生日期            |
| Specialty_id                                      | Varchar (3)  | Not null | 专业 (外键, 参照专业表)  |
| Depart_id   | Varchar (2)  | Not null | 系部 (外键, 参照系部表)  |
| tel   | Varchar (11) |          | 联系电话 (11 位的数字)  |
| qq  | Varchar (20) |          | QQ              |
| E_mail  | Varchar (30) |          | E_mail          |
| credits   | Varchar (3)  |          | 总学分 (初始值为 0)    |
| note  | text         |          | 备注              |
| 表 1.10 成绩表 (score_tab)                            |              |          |                 |
| 列名  | 数据类型         | 可否为空     | 说明              |
| id  | Int          | Not null | 编号 (递增 1)       |
| Stud_id   | Varchar (15) | Not null | 学号 (外键, 参照学生表)  |
| course_id   | Varchar (10) | Not null | 课程号 (外键, 参照课程表) |
| score   | Varhcar (3)  | Not null | 成绩              |
| 任务二: 数据表的修改                                       |              |          |                 |
| 1. 使用 SQL Server Managemet Studio 修改表             |              |          |                 |
| 2. 使用 SQL 语句修改表                                   |              |          |                 |
| (1) 在学生表中添加一个 stud_sfz(身份证号)列, 字符型, 长度为 10        |              |          |                 |
| (2) 修改学生表的结构, 增加 age 字段, 数据类型为: int               |              |          |                 |
| (3) 修改学生表的结构, 把身份证字段的长度改为 18 位, 数据类型为: varchar    |              |          |                 |
| (4) 将成绩表的 score (成绩) 字段, 设置为 decimal(5, 2)        |              |          |                 |
| (5) 在成绩表的 score (成绩) 字段中设置一个约束条件, 要求成绩只能为 0~100 分 |              |          |                 |
| (6) 在学生信息表的 credits (总学分) 字段中将默认值设置为 “0”          |              |          |                 |
| 任务三: 管理表中的数据                                      |              |          |                 |
| 1. 使用 SQL Server Managemet Studio 向表中的添加数据        |              |          |                 |
| 2. 使用 SQL 语句管理表中的数据                               |              |          |                 |
| (1) 使用 SQL 语句向表中插入数据                              |              |          |                 |
| (2) 使用 SQL 语句向更新表中数据                              |              |          |                 |
| (3) 使用 SQL 语句向删除表中数据                              |              |          |                 |
| 三、工作流程  |              |          |                 |

## 任务一：数据表的创建

### 1. 使用 Server Managemet Studio 创建数据表

以系部表为例：

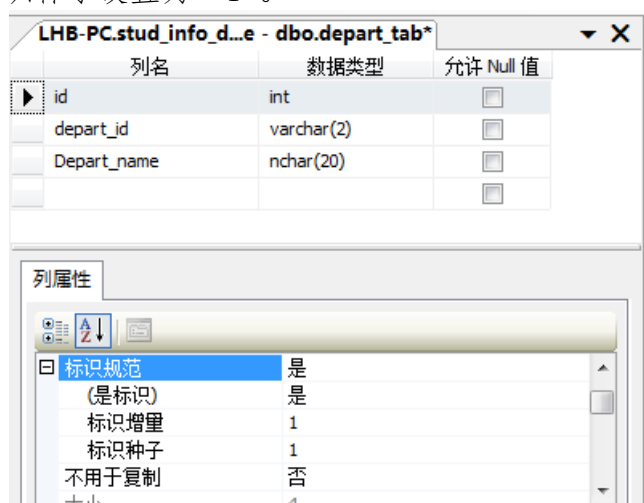
第一步：在 SQL Server Management Studio 的“对象资源管理器”中，依次单击展开节点“数据库”→“stud\_info\_database”→“表”，用鼠标右键单击“表”节点，在弹出的快捷菜单中选择“新建表”命令。



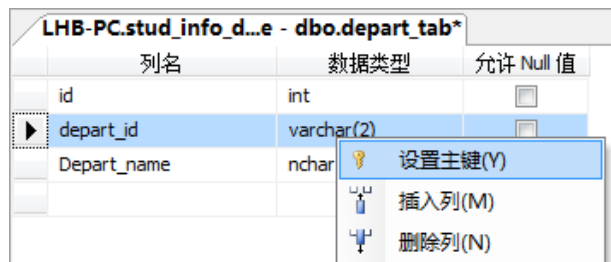
第二步：根据系部表的结构，定义列的属性，设置列名、数据类型、长度、是否为空等。

| LHB-PC.stud_info_d...e - dbo.depart_tab* |             |            |                          |
|--|-------------|------------|--------------------------|
|  | 列名          | 数据类型       | 允许 Null 值                |
|  | id          | int        | <input type="checkbox"/> |
| ▶  | depart_id   | varchar(2) | <input type="checkbox"/> |
|  | Depart_name | nchar(20)  | <input type="checkbox"/> |
|  |             |            | <input type="checkbox"/> |

第三步：设置列名“id”的标识规范，在“是标识”中设置为“是”，标识增量和标识种子设置为“1”。



第四步：设置列名“depart\_id”为主键。用鼠标右击列名“depart\_id”，在弹出的快捷菜单中选择“设置主键”命令。



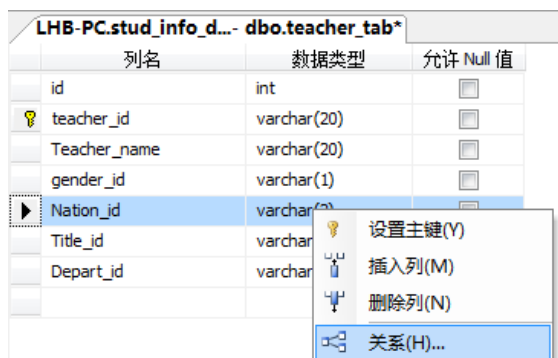
第五步：将表保存为“depart\_tab”。

## (2) 建立表与表之间的联系

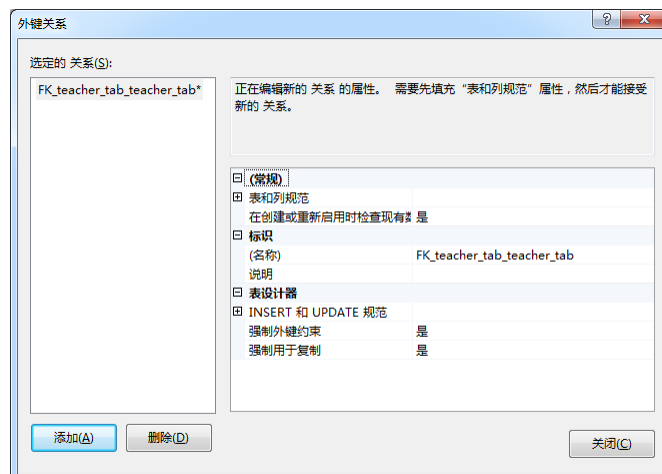
以教师表中 depart\_id 与系部表中的 depart\_id 为例

第一步：打开教师表的设计窗口。

第二步：在教师表的设计窗口中，用鼠标右击，在弹出的快捷菜单中选择“关系”。



第三步：在“外键关系”窗口中，单击“添加”按钮。



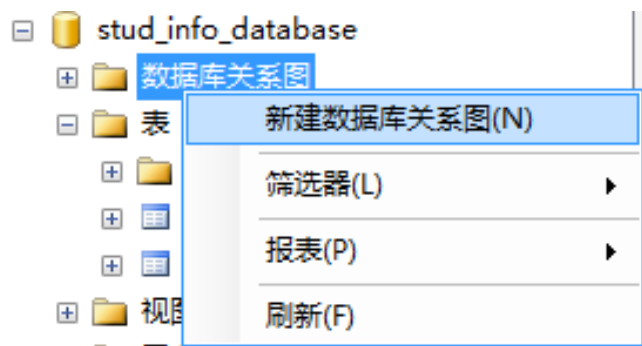
第四步：展开“表和列规范”节点，单击最右侧的，弹出“表和列”窗口。

第五步：在“表和列”窗口中，在关系名文本框中输入“FK\_depart\_tg”，在“主键表”的下拉列表框中选择“depart\_tab”，并选择“depart\_id”，在“外键表”teacher\_tab 中选择“depart\_id”，单击“确定”按钮。

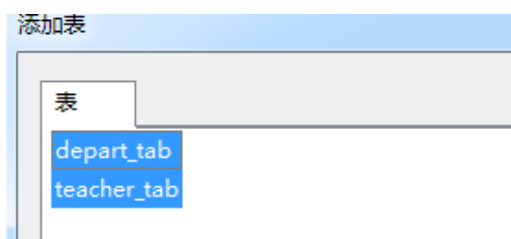
温馨提示：也可以在数据库关系图中创建关系中，通过用鼠标拖动的方式建立关系，具体方法是：选中要创建的外键字段，将它拖动至主表中的主键字段中。

(3) 查看数据库库关系图。

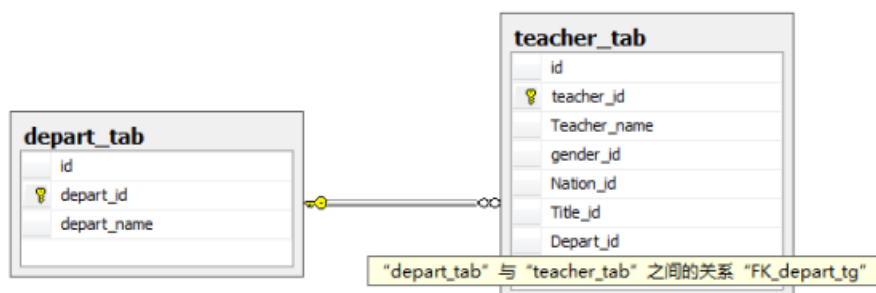
第一步：在 SQL Server Management Studio 的“对象资源管理器”中，展开“stud\_info\_database”节，用鼠标右击“数据库关系图”，在弹出的快捷菜单中选择“新建数据库关系图”。



第二步：在“添加表”窗口中，选择您要查看的表，单击“添加”按钮。



系部表与教师表之间建立的关系如下图所示。



如果所有的外键创建好以后，数据库的关系图如下。



(4) 使用 SQL 语句创建表



以民族表和课程表为例

SQL 代码略

任务二： 数据表的修改

1. 使用 SQL Server Managemet Studio 修改表

略

2. 使用 SQL 语句管理表中的数据

(1) 在学生表中添加一个 stud\_sfz(身份证号)列，字符型，长度为 10  
use stud\_info\_database

```
alter table student_tab add stud_sfz char(10)
```

(2) 修改学生表的结构，增加 age 字段，数据类型为：int

```
alter table student_tab add age int
```

(3) 修改学生表的结构，把身份证字段的长度改为 18 位，数据类型为：  
varchar

```
alter table student_tab alter column stud_sfz varchar(18)
```

(4) 将成绩表的 score(成绩)字段，设置为 decimal(5, 2)

```
alter table score_tab alter column score decimal(5, 2)
```

(5) 在成绩表的 score(成绩)字段中设置一个约束条件，要求成绩  
只能为 0~100 分。

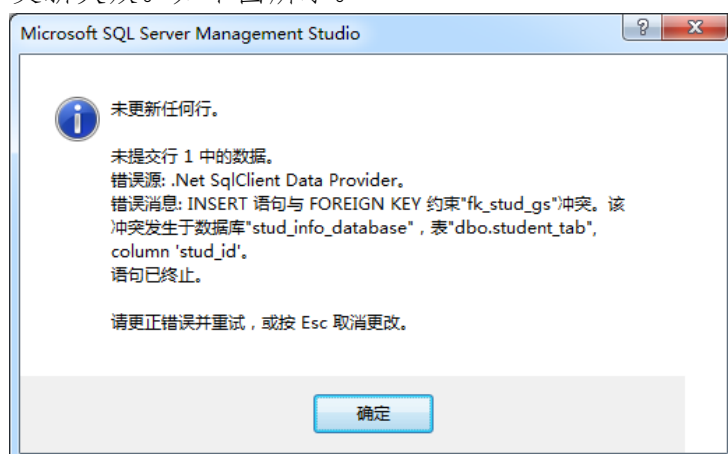
```
alter table score_tab add constraint ck_score check(score between 0 and 100)
```

(6) 在学生信息表的 credits(总学分)字段中将默认值设置为“0”。

代码附于工作报告中

任务三：管理表中的数据

注意：在表中插入、更新、删除数据，如果该表有关联表，在插入数据时，如果关联表中没有相应的记录，则插入失败；在删除数据时，如果被删除的记录在关系表中有记录，删除失败，更新数据时，在关联表中没有相应的数据，更新失败。如下图所示。



1. 使用 SQL Server Managemet Studio 向表中的添加数据

使用 SQL Server Managemet Studio 向表中的添加如下数据

系部表

| depart_id | Depart_name |
|-----------|-------------|
| 01        | 信息系         |
| 02        | 工商系         |
| 03        | 建筑系         |
| 04        | 数学系         |

民族表

|    |      |
|----|------|
| 01 | 汉族   |
| 02 | 阿昌族  |
| 03 | 白族   |
| 04 | 保安族  |
| 05 | 布朗族  |
| 06 | 布依族  |
| 07 | 达斡尔族 |
| 08 | 朝鲜族  |
| 09 | 傣族   |
| 10 | 德昂族  |

性别表

| gender_id | gender_name |
|-----------|-------------|
| 0         | 男           |
| 1         | 女           |

专业表

| Specialty_id | Specialty_name |
|--------------|----------------|
| 01           | 计算机应用技术        |
| 02           | 网络管理           |
| 03           | 报关与国际货运        |
| 04           | 电子商务           |
| 05           | 市场营销           |
| 06           | 建设工程管理         |
| 07           | 基础数学           |

职称表

| title_id | title_name |
|----------|------------|
| 01       | 无          |
| 02       | 助教         |
| 03       | 讲师         |
| 04       | 双师         |
| 05       | 副教授        |
| 06       | 教授         |
| 07       | 研究员        |

教师表

| teacher_id | Teacher_name | gender_id | Nation_id | Title_id | Depart_id |
|------------|--------------|-----------|-----------|----------|-----------|
| 301        | 王梅           | 1         | 01        | 01       | 01        |
| 302        | 王海           | 0         | 01        | 05       | 01        |
| 303        | 张东           | 0         | 02        | 04       | 02        |
| 304        | 刘军           | 0         | 03        | 03       | 02        |
| 305        | 罗晓           | 1         | 04        | 06       | 03        |

学生表

| Stud_id | Stud_name | gender_id | Nation_id | brithday | Specialty_id | Depart_id |
|---------|-----------|-----------|-----------|----------|--------------|-----------|
| 101     | 张海        | 0         | 01        |          | 01           | 01        |
| 102     | 王艳        | 1         | 01        |          | 01           | 01        |
| 103     | 叶强        | 0         | 02        |          | 03           | 02        |
| 104     | 刘兰        | 1         | 04        |          | 04           | 02        |
| 105     | 罗小红       | 1         | 01        |          | 06           | 04        |
| 106     | 林小华       | 0         | 03        |          | 06           | 04        |

## 2. 使用 SQL 语句管理表中的数据

### (1) 使用 SQL 语句向表中插入数据

以课程表和学生成绩表为例

课程表

| course_id | Course_name | Course_xizi | teacher_id | term | First_course | Depart_id | Class_hour | credits |
|-----------|-------------|-------------|------------|------|--------------|-----------|------------|---------|
| 100101    | 计算机基础       | 通识课         | 301        | 第一学期 | 无            | 01        | 48         | 3       |
| 100102    | 数据库设计       | 专业核心课       | 302        | 第二学期 | 100101       | 01        | 64         | 4       |
| 100103    | 服务器配置       | 专业核心课       | 302        | 第三学期 | 100101       | 01        | 64         | 4       |
| 100104    | 网络故障排除      | 专业核心课       | 301        | 第五学期 | 100103       | 01        | 64         | 4       |
| 100105    | 网络营销        | 专业基础课       | 303        | 第一学期 | 无            | 02        | 48         | 3       |
| 100106    | 电子商务运营      | 专业基础课       | 303        | 第一学期 | 100104       | 02        | 48         | 3       |
| 100107    | 工程识图        | 专业基础课       | 303        | 第一学期 | 无            | 03        | 48         | 3       |
| 100108    | 建筑基础        | 专业基础课       | 304        | 第二学期 | 无            | 03        | 64         | 4       |
| 100109    | 建筑施工技术      | 专业核心课       | 304        | 第四学期 | 100108       | 03        | 48         | 3       |
| 100110    | 高等数学        | 通识          | 305        | 第一学期 | 无            | 04        | 48         | 3       |

|         |           |       |  |  |
|---------|-----------|-------|--|--|
| 成绩表     |           |       |  |  |
| Stud_id | course_id | score |  |  |
| 101     | 100101    | 68    |  |  |
| 101     | 100102    | 72    |  |  |
| 101     | 100103    | 55    |  |  |
| 102     | 100101    | 88    |  |  |
| 102     | 100102    | 92    |  |  |
| 102     | 100103    | 76    |  |  |
| 103     | 100105    | 85    |  |  |
| 103     | 100106    | 69    |  |  |
| 104     | 100105    | 75    |  |  |
| 104     | 100106    | 85    |  |  |
| 105     | 100110    | 52    |  |  |

--向课程表中插入数据

insert into course\_tab

(course\_id,Course\_name,Course\_xizi,teacher\_id,term,First\_course,Depart\_id,Class\_hour,credits) values('100101','计算机基础','通识课','301','第一学期','无','01','48','3')

--向成绩表中插入数据

insert into score\_tab values('101','100101',68)

(2) 使用 SQL 语句向更新表中数据

要求：使用 SQL 语句向学生表中更新如下数据

代码附于工作报告中

|         |             |            |                    |     |
|---------|-------------|------------|--------------------|-----|
| Stud_id | tel         | qq         | stud_sfz           | age |
| 101     | 13547856792 | 298451124  | 500236199701037035 | 20  |
| 102     | 13102567756 | 156384759  | 510503199505061243 | 22  |
| 103     | 18588485612 | 5684235792 | 500230199606144990 | 21  |
| 104     | 15025146983 | 12574684   | 340823199808051966 | 19  |
| 105     | 15388456631 | 65214521   | 412723199611126424 | 21  |
| 106     | 17788456631 | 568749562  | 500383199512127017 | 22  |

(3) 使用 SQL 语句向删除表中数据

删除民族号为 10 的记录。

删除学生成绩表中的所有记录。

考核标准（技能和素质考核）

1、专业技能考核标准（共 80 分）

(1) 正确创建数据表；（40 分）

(2) 正确修改数据表；（15 分）

(3) 完成表中数据的插入、修改和删除；（25 分）

2、学习态度考核标准（共 20 分）

(1) 实训报告书质量（5 分）

认真对实训过程进行总结，发现和解决问题；认真按照要求填写；书面整

洁，字迹清楚。

(2) 工作职业道德 (5 分)

安全文明工作，职业操守较好；爱护计算机等公共设施；基本按照布置的工作任务和要求去完成。

(3) 团队合作精神 (5 分)

具有良好的团队合作精神，热心帮助班级或小组的其它成员；能与团队成员有效沟通；能合理分配小组成员的工作任务；

(4) 语言沟通能力 (5 分)

能用专业的语言正确流利地展示项目成果；能准确回答老师提出的相关问题

### 问题与思考

1. 请写出创建教师表的 SQL 语句（代码附于工作报告中）。
2. 在学生信息表的 credits（总学分）字段中将默认值设置为“0”（代码附于工作报告中）。
3. 使用 SQL 语句向更新学生表中数据（代码附于工作报告中）。

## 一体化课程项目工作任务单

| 名 称  | 数据查询  | 序号 | 4        |
|--|---|----|----------|
| 目 标  | 1. 掌握 SQL 查询语句的基本结构<br>2. 具备简单查询的能力<br>3. 具备条件查询的能力<br>4. 具备连接查询的能力<br>5. 具备子查询的能力                      |    |          |
| 工作要求   | 1. 理解 SQL 语句的基本结构<br>2. 完成数据表的简单查询<br>3. 完成数据表的条件查询<br>4. 完成数据表的连接查询<br>5. 完成数据表的子查询<br>6. 根据工作过程写出实验报告 |    |          |
| 企业标准   | SQL 代码必须符合基本编程规范  |    |          |
| 工作条件   | 1. 操作系统: Windows 7<br>2. 软件环境: SQL Server 2008 数据库  |    |          |
| 工作流程   |   |    | 注意<br>事项 |
| <p><b>一、工作背景</b></p> <p>abc 学校的信息中心在完成数据库表的创建和数据的录入以后, 需要进一步测试表中的数据, 开发小组提出通过 SQL 查询来测试数据表中的数据。</p> <p><b>二、工作内容</b></p> <p>工作内容:</p> <p>任务一: 数据表的简单查询</p> <ol style="list-style-type: none"> <li>1. 查询全体教师的教工号与姓名</li> <li>2. 查询学生的学号, 姓名, 电话号码和身份证号, 并使用列别名改变查询结果的列标题</li> <li>3. 查询所有课程的详细信息</li> <li>4. 查全体学生的姓名、学号、电话及身份证号</li> <li>5. 查询有教学任务的教师的教工号</li> <li>6. 查询选修了课程的学生人数</li> <li>7. 查询教师总人数</li> <li>8. 求各个课程号及相应的选课人数</li> <li>8. 查询全体学生成绩情况, 查询结果成绩降序排列, 成绩相同情况</li> </ol> |   |    |          |

下按学号升序排列

任务二：数据表的条件查询

1. 查询成绩不及格的学生学号、课程号和成绩  
2. 查询 100103 课程的平均分、最高分及最低分，查询结果按分数降序排列

3. 查询选修了两门以上课程的学生学号
4. 查询有 2 门以上的课程是 85 分以上的学生学号
5. 查询成绩在 70~89 分之间的学生学号及课程号及成绩
6. 查询成绩不在 70~79 分之间的学生学号、课程号及成绩
7. 查询汉族和阿昌族的学生详细信息
8. 查询姓王，并且名为一个字的教师详细信息
9. 查询电话以 15 开头的学生详细信息

任务三：数据表的连接查询

1. 使用内连接查询学生的学号，姓名，选修课程及成绩
2. 使用左外连接查询学生的学号，姓名，选修课程及成绩
3. 使用右外连接查询学生的学号，姓名，选修课程及成绩
4. 使用全外连接查询学生的学号，姓名，选修课程及成绩
5. 查询成绩大于 80 分的学生学号，姓名，课程名及该课程的成绩，查询结果先按学号由低到高排序，同一学生的成绩按由高到低排序
6. 统计各门课程的选课情况（课程名，选课人数，该门课程的平均分），查询结果按成绩的由高到低排序

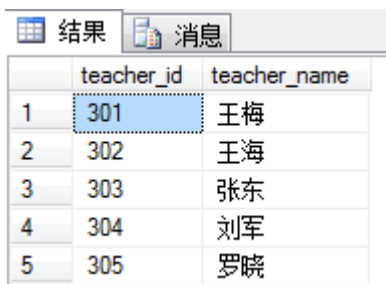
任务四：数据表的子查询

1. 查询与张海是同一民族的学生的学号和姓名
2. 查询选修了网络营销的学生的学号和姓名
3. 查询没有被学生选的课程号及课程名
4. 查询其他系中比数学系任意一个学生年龄小的学生的姓名和年龄

### 三、工作流程

任务一：数据表的简单查询

1. 查询全体教师的教工号与姓名



|   | teacher_id | teacher_name |
|---|------------|--------------|
| 1 | 301        | 王梅           |
| 2 | 302        | 王海           |
| 3 | 303        | 张东           |
| 4 | 304        | 刘军           |
| 5 | 305        | 罗晓           |

2. 查询学生的学号，姓名，电话号码和身份证号，并使用列别名改变查询结果的列标题

| 学号  | 姓名  | 电话号码        | 身份证号码              |
|-----|-----|-------------|--------------------|
| 101 | 张海  | 13547856792 | 500236199501037035 |
| 102 | 王艳  | 13102567756 | 510503199405061243 |
| 103 | 叶强  | 18588485612 | 500230199406144990 |
| 104 | 刘兰  | 15025146983 | 340823199808051966 |
| 105 | 罗小红 | 15388456631 | 412723199611126424 |

### 3. 查询所有课程的详细信息

| course_id | course_name | course_xizi | teacher_id | term | first_course | depart_id | class_hour | credits | note |
|-----------|-------------|-------------|------------|------|--------------|-----------|------------|---------|------|
| 100101    | 计算机基础       | 通识课         | 301        | 第一学期 | 无            | 01        | 48         | 3       | NULL |
| 100102    | 数据库设计       | 专业核心课       | 302        | 第二学期 | 100101       | 01        | 64         | 4       | NULL |
| 100103    | 服务器配置       | 专业核心课       | 302        | 第三学期 | 100101       | 01        | 64         | 4       | NULL |
| 100104    | 网络故障排除      | 专业核心课       | 301        | 第五学期 | 100103       | 01        | 64         | 4       | NULL |
| 100105    | 网络营销        | 专业基础课       | 303        | 第一学期 | 无            | 02        | 48         | 3       | NULL |
| 100106    | 电子商务运营      | 专业基础课       | 303        | 第一学期 | 100104       | 02        | 48         | 3       | NULL |
| 100107    | 工程识图        | 专业基础课       | 303        | 第一学期 | 无            | 03        | 48         | 3       | NULL |
| 100108    | 建筑基础        | 专业基础课       | 304        | 第二学期 | 无            | 03        | 64         | 4       | NULL |
| 100109    | 建筑施工技术      | 专业核心课       | 304        | 第四学期 | 100108       | 03        | 48         | 3       | NULL |
| 100110    | 高等数学        | 通识课         | 305        | 第一学期 | 无            | 04        | 48         | 3       | NULL |

### 4. 查全体学生的姓名、学号、电话及身份证号

| stud_id | stud_name | stud_sfz           |
|---------|-----------|--------------------|
| 101     | 张海        | 500236199501037035 |
| 102     | 王艳        | 510503199405061243 |
| 103     | 叶强        | 500230199406144990 |
| 104     | 刘兰        | 340823199808051966 |
| 105     | 罗小红       | 412723199611126424 |

### 5. 查询有教学任务的教师的教工号

|   | teacher_id |
|---|------------|
| 1 | 301        |
| 2 | 302        |
| 3 | 303        |
| 4 | 304        |
| 5 | 305        |

### 6. 查询选修了课程的学生人数

图略

### 7. 查询教师总人数

图略

### 8. 求各个课程号及相应的选课人数

|   | course_id | (无列名) |
|---|-----------|-------|
| 1 | 100101    | 2     |
| 2 | 100102    | 2     |
| 3 | 100103    | 2     |
| 4 | 100105    | 2     |
| 5 | 100106    | 2     |
| 6 | 100110    | 1     |

### 9. 查询全体学生成绩情况，查询结果成绩降序排列，成绩相同情况下按学号升序排列

图略



## 任务二：数据表的条件查询

### 1. 查询成绩不及格的学生学号、课程号和成绩

|   | stud_id | course_id | score |
|---|---------|-----------|-------|
| 1 | 101     | 100103    | 55    |
| 2 | 105     | 100110    | 52    |

### 2. 查询 100103 课程的平均分、最高分及最低分，查询结果按分数降序排列

图略

### 3. 查询选修了两门以上课程的学生学号

|   | stud_id |
|---|---------|
| 1 | 101     |
| 2 | 102     |
| 3 | 103     |
| 4 | 104     |

### 4. 查询有 2 门以上的课程是 85 分以上的学生学号

```
select stud_id, COUNT(course_id)
```

```
from score_tab
```

```
where score >= 85
```

```
group by stud_id
```

```
having COUNT(course_id) >= 2
```

### 5. 查询成绩在 70~89 分之间的学生学号及课程号及成绩

|   | stud_id | course_id | score |
|---|---------|-----------|-------|
| 1 | 101     | 100102    | 72.00 |
| 2 | 102     | 100101    | 88.00 |
| 3 | 102     | 100103    | 76.00 |
| 4 | 103     | 100105    | 85.00 |
| 5 | 104     | 100105    | 75.00 |
| 6 | 104     | 100106    | 85.00 |

### 6. 查询成绩不在 70~79 分之间的学生学号、课程号及成绩

|   | stud_id | course_id | score |
|---|---------|-----------|-------|
| 1 | 101     | 100101    | 68.00 |
| 2 | 101     | 100103    | 55.00 |
| 3 | 102     | 100101    | 88.00 |
| 4 | 102     | 100102    | 92.00 |
| 5 | 103     | 100105    | 85.00 |
| 6 | 103     | 100106    | 69.00 |
| 7 | 104     | 100106    | 85.00 |
| 8 | 105     | 100110    | 52.00 |

### 7. 查询汉族和阿昌族的学生详细信息

```
select * from student_tab
where nation_id in('01','02')
```

8. 查询姓王，并且名为一个字的教师详细信息

| id | teacher_id | teacher_name | gender_id | nation_id | title_id | depart_id |
|----|------------|--------------|-----------|-----------|----------|-----------|
| 1  | 301        | 王梅           | 1         | 01        | 01       | 01        |
| 2  | 302        | 王海           | 0         | 01        | 05       | 01        |

9. 查询电话以 15 开头的学生详细信息

| id | stud_id | stud_name | gender_id | nation_id | brithday | Specialty_id | depart_id | tel         | qq       | e_mail | credits | note | stud_sfz           |
|----|---------|-----------|-----------|-----------|----------|--------------|-----------|-------------|----------|--------|---------|------|--------------------|
| 4  | 104     | 刘兰        | 1         | 04        | NULL     | 04           | 02        | 15025146983 | 12574684 | NULL   | 0       | NULL | 340823199808051966 |
| 5  | 105     | 罗小红       | 1         | 01        | NULL     | 06           | 04        | 15388456631 | 65214521 | NULL   | 0       | NULL | 412723199611126424 |

任务三：数据表的连接查询

1. 使用内连接查询学生的学号，姓名，选修课程及成绩

|    | 学号  | 姓名  | 课程名    | 成绩    |
|----|-----|-----|--------|-------|
| 1  | 101 | 张海  | 计算机基础  | 68.00 |
| 2  | 101 | 张海  | 数据库设计  | 72.00 |
| 3  | 101 | 张海  | 服务器配置  | 55.00 |
| 4  | 102 | 王艳  | 计算机基础  | 88.00 |
| 5  | 102 | 王艳  | 数据库设计  | 92.00 |
| 6  | 102 | 王艳  | 服务器配置  | 76.00 |
| 7  | 103 | 叶强  | 网络营销   | 85.00 |
| 8  | 103 | 叶强  | 电子商务运营 | 69.00 |
| 9  | 104 | 刘兰  | 网络营销   | 75.00 |
| 10 | 104 | 刘兰  | 电子商务运营 | 85.00 |
| 11 | 105 | 罗小红 | 高等数学   | 52.00 |

2. 使用左外连接查询学生的学号，姓名，选修课程及成绩

|    | 学号  | 姓名  | 课程名    | 成绩    |
|----|-----|-----|--------|-------|
| 1  | 101 | 张海  | 计算机基础  | 68.00 |
| 2  | 101 | 张海  | 数据库设计  | 72.00 |
| 3  | 101 | 张海  | 服务器配置  | 55.00 |
| 4  | 102 | 王艳  | 计算机基础  | 88.00 |
| 5  | 102 | 王艳  | 数据库设计  | 92.00 |
| 6  | 102 | 王艳  | 服务器配置  | 76.00 |
| 7  | 103 | 叶强  | 网络营销   | 85.00 |
| 8  | 103 | 叶强  | 电子商务运营 | 69.00 |
| 9  | 104 | 刘兰  | 网络营销   | 75.00 |
| 10 | 104 | 刘兰  | 电子商务运营 | 85.00 |
| 11 | 105 | 罗小红 | 高等数学   | 52.00 |
| 12 | 106 | 林小华 | NULL   | NULL  |

3. 使用右外连接查询学生的学号，姓名，选修课程及成绩

|    | 学号   | 姓名   | 课程名    | 成绩    |
|----|------|------|--------|-------|
| 1  | 101  | 张海   | 计算机基础  | 68.00 |
| 2  | 102  | 王艳   | 计算机基础  | 88.00 |
| 3  | 101  | 张海   | 数据库设计  | 72.00 |
| 4  | 102  | 王艳   | 数据库设计  | 92.00 |
| 5  | 101  | 张海   | 服务器配置  | 55.00 |
| 6  | 102  | 王艳   | 服务器配置  | 76.00 |
| 7  | NULL | NULL | 网络故障排除 | NULL  |
| 8  | 103  | 叶强   | 网络营销   | 85.00 |
| 9  | 104  | 刘兰   | 网络营销   | 75.00 |
| 10 | 103  | 叶强   | 电子商务运营 | 69.00 |
| 11 | 104  | 刘兰   | 电子商务运营 | 85.00 |
| 12 | NULL | NULL | 工程识图   | NULL  |
| 13 | NULL | NULL | 建筑基础   | NULL  |
| 14 | NULL | NULL | 建筑施工技术 | NULL  |
| 15 | 105  | 罗小红  | 高等数学   | 52.00 |

4. 使用全外连接查询学生的学号，姓名，选修课程及成绩

|    | 学号   | 姓名   | 课程名    | 成绩    |
|----|------|------|--------|-------|
| 1  | 101  | 张海   | 计算机基础  | 68.00 |
| 2  | 101  | 张海   | 数据库设计  | 72.00 |
| 3  | 101  | 张海   | 服务器配置  | 55.00 |
| 4  | 102  | 王艳   | 计算机基础  | 88.00 |
| 5  | 102  | 王艳   | 数据库设计  | 92.00 |
| 6  | 102  | 王艳   | 服务器配置  | 76.00 |
| 7  | 103  | 叶强   | 网络营销   | 85.00 |
| 8  | 103  | 叶强   | 电子商务运营 | 69.00 |
| 9  | 104  | 刘兰   | 网络营销   | 75.00 |
| 10 | 104  | 刘兰   | 电子商务运营 | 85.00 |
| 11 | 105  | 罗小红  | 高等数学   | 52.00 |
| 12 | 106  | 林小华  | NULL   | NULL  |
| 13 | NULL | NULL | 网络故障排除 | NULL  |
| 14 | NULL | NULL | 工程识图   | NULL  |
| 15 | NULL | NULL | 建筑施工技术 | NULL  |
| 16 | NULL | NULL | 建筑基础   | NULL  |

5. 查询成绩大于 80 分的学生学号，姓名，课程名及该课程的成绩，  
查询结果先按学号由低到高排序，同一学生的成绩按由高到低排序

|   | 学号  | 姓名 | 课程名    | 成绩    |
|---|-----|----|--------|-------|
| 1 | 102 | 王艳 | 数据库设计  | 92.00 |
| 2 | 102 | 王艳 | 计算机基础  | 88.00 |
| 3 | 103 | 叶强 | 网络营销   | 85.00 |
| 4 | 104 | 刘兰 | 电子商务运营 | 85.00 |

6. 统计各门课程的选课情况（课程名，选课人数，该门课程的平均分），  
查询结果按成绩的由高到低排序

| 课程名    | 选课人数 | 平均分       |
|--------|------|-----------|
| 数据库设计  | 2    | 82.000000 |
| 网络营销   | 2    | 80.000000 |
| 计算机基础  | 2    | 78.000000 |
| 电子商务运营 | 2    | 77.000000 |
| 服务器配置  | 2    | 65.500000 |
| 高等数学   | 1    | 52.000000 |

任务四：数据表的子查询

1. 查询与张海是同一民族的学生的学号和姓名

第一步：确定张海的民族

|   | nation_id |
|---|-----------|
| 1 | 01        |

第二步：查询所有民族号为 01 的学生的学号和姓名

| stud_id | stud_name |
|---------|-----------|
| 101     | 张海        |
| 102     | 王艳        |
| 105     | 罗小红       |

第三步：将第一步查询嵌入到第二步查询的条件中

2. 查询选修了网络营销的学生的学号和姓名

第一步：首先要知道课程网络营销的课程代号

第二步：根据课程号查询相应的学生学号

第三步：根据学生学号查询学生姓名

第四步：将第一至三步综合起来

|   | stud_name |
|---|-----------|
| 1 | 叶强        |
| 2 | 刘兰        |

3. 查询没有被学生选的课程号及课程名

|   | course_id | course_name |
|---|-----------|-------------|
| 1 | 100104    | 网络故障排除      |
| 2 | 100107    | 工程识图        |
| 3 | 100108    | 建筑基础        |
| 4 | 100109    | 建筑施工技术      |

4. 查询其他系中比数学系任意一个学生年龄小的学生的姓名和年龄

第一步：查询数学系的学生的年龄情况

第二步：根据系名查询相应的系部号

第三步：找出最小年龄的学生学号和年龄，并且不在数学系

第四步：将第一至三步综合起来

| 结果 |           | 消息  |
|----|-----------|-----|
|    | stud_name | age |
| 1  | 张海        | 20  |
| 2  | 刘兰        | 19  |

### 考核标准（技能和素质考核）

#### 1、专业技能考核标准（共 80 分）

- （1）正确完成数据表的简单查询；（20 分）
- （2）正确完成数据表的条件查询；（20 分）
- （3）正确完成数据表的连接查询；（20 分）
- （4）正确完成数据表的子查询（20 分）

#### 2、学习态度考核标准（共 20 分）

- （1）实训报告书质量（5 分）

认真对实训过程进行总结，发现和解决问题；认真按照要求填写；书面整洁，字迹清楚。

- （2）工作职业道德（5 分）

安全文明工作，职业操守较好；爱护计算机等公共设施；基本按照布置的工作任务和要求去完成。

|   |
|---|
| <p>(3) 团队合作精神 (5 分)</p> <p>具有良好的团队合作精神，热心帮助班级或小组的其它成员；能与团队成员有效沟通；能合理分配小组成员的工作任务；</p> <p>(4) 语言沟通能力 (5 分)</p> <p>能用专业的语言正确流利地展示项目成果；能准确回答老师提出的相关问题</p>   |
| <p style="text-align: center;"><b>问题与思考</b></p>   |
| <p>1. 编写 SQL 语句，查询学生的学号，姓名，选修课程及成绩（用两种方式实现，代码附于工作报告中）。</p> <p>2. 统计各门课程的选课情况（课程名，选课人数，该门课程的平均分），查询结果按成绩的由高到低排序（用两种方式实现，代码附于工作报告中）</p> <p>3. 写出“查询其他系中比数学系任意一个学生年龄小的学生的姓名和年龄”的查询思路和 SQL 代码（代码附于工作报告中）。</p> |

## 一体化课程项目工作任务单

|   |  |    |         |
|---|--|----|---------|
| 名 称   | 视图、存储过程  | 序号 | 5       |
| 目 标   | 1. 具备视图创建的能力<br>2. 具备视图修改的能力<br>3. 具备创建存储过程的能力<br>4. 具备调用存储过程的能力 |    |         |
| 工作要求  | 1. 完成视图的创建与修改<br>2. 完成存储过程的创建与调用<br>3. 根据工作过程写出实验报告              |    |         |
| 企业标准  | T-SQL 代码必须符合基本编程规范   |    |         |
| 工作条件  | 1. 操作系统： Windows 7<br>2. 软件环境： SQL Server 2008 数据库               |    |         |
| 工 作 流 程   |  |    | 注 意 事 项 |
| <p><b>一、工作背景</b></p> <p>abc 学校的信息中心在完成系统开发以后，进一步对系统进行完善，开发小组提出需要创建视图和存储过程。</p> <p><b>二、工作内容</b></p> <p>任务一：视图的创建与修改</p> <p>1. 使用 SQL Server Managemet Studio 创建视图 “teacher_view” 要求显示教师的姓名、性别、职称和所在的系部</p> <p>2. 使用 SQL 语句创建视图 “student_view” 要求显示学生的学号，姓名，所选课程和成绩</p> <p>3. 使用 SQL Server Managemet Studio 修改视图 “teacher_view” 要求显示教师的工号、姓名、性别、民族、职称和所在的系部，并按教师号降序排序</p> <p>4. 使用 SQL 语句修改视图 “student_view” 要求显示学生的学号，姓名，所选课程和成绩，并按成绩降序排序，同成绩按学号升序排序</p> <p>任务二：存储过程的创建与调用</p> <p>1. 创建一个简单的存储过程 pro_stu1，查询所有学生的信息</p> <p>2. 创建一个带有输入参数的存储过程 pro_stu2，要求根据输入学生的学号来查询相应的学生信息。</p> <p>3. 创建一个带有两输入参数的存储过程 pro_score1 要求输入两个 0-100 的整数，查询大于成绩在这两个数之间的学生姓名、所选择课程和成绩</p> |  |    |         |

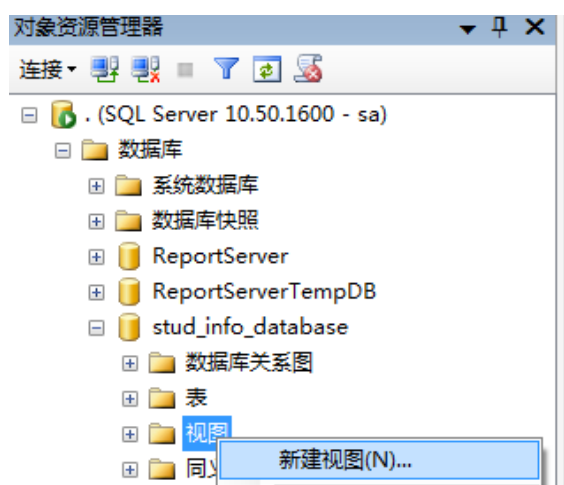
4. 创建一个带有输入参数和输出参数的存储过程 pro\_teacher，其中输入参数用于接收教师工号，输出参数用于显示该教师的姓名、性别、职称和所在的系部

### 三、工作流程

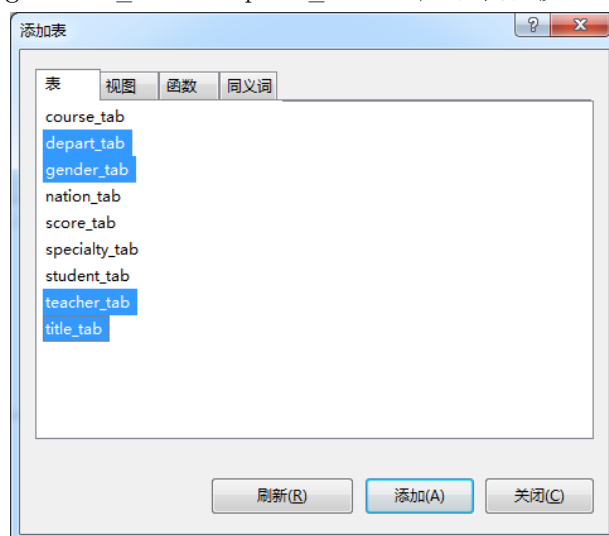
任务一：视图的创建与修改

1. 使用 SQL Server Managemet Studio 创建视图 “teacher\_view” 要求显示教师的姓名、性别、职称和所在的系部

第一步：在 SQL Server Management Studio 的“对象资源管理器”中，依次单击展开节点“数据库”→“stud\_info\_database”→“视图”，用鼠标右键单击“视图”节点，在弹出的快捷菜单中选择“新建视图”命令。

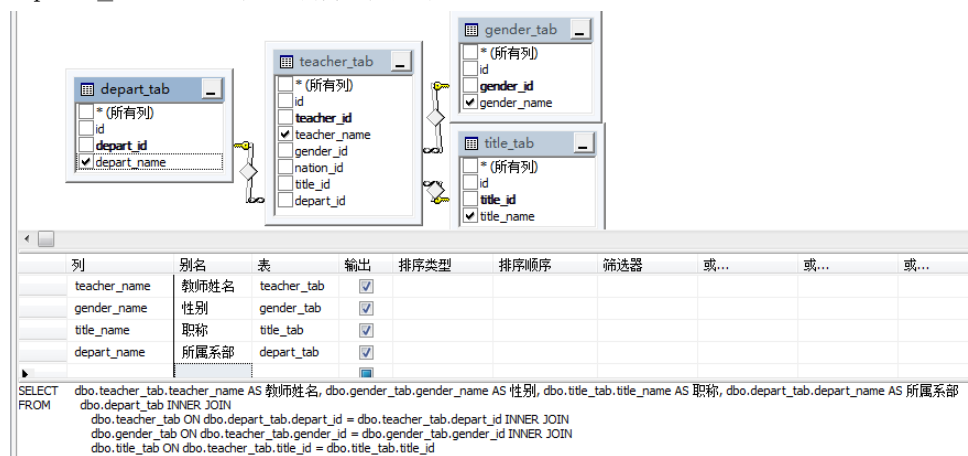


第二步：在弹出的“添加表”对话框中选择“Teacher\_tab、title\_tab、gender \_tab、Depart\_tab”单击添加按钮。



第三步：表添加完成后，选择四个表中需要提取的列，分别在 Teacher\_tab 表中选择 Teacher\_name，在 gender \_tab 表中选择 gender\_name，在 title\_tab 表中选择 title\_name，在 Depart\_tab 表

中选择 depart\_name，并设置 teacher\_name 的别名为教师姓名，设置 gender\_name 的别名为性别，设置 title\_name 的别名为职称，设置 depart\_name 的别名为所属系部。



第四步：设置内部联接。如果在创建表的时候已经创建了外键，此步骤可以略。

第五步：单击工具栏中的“查询设计器”→“执行 SQL 命令”执行查询，查询结果如下图所示。

| 教师姓名 | 性别 | 职称  | 所属系部 |
|------|----|-----|------|
| 王梅   | 女  | 无   | 信息系  |
| 王海   | 男  | 副教授 | 信息系  |
| 张东   | 男  | 双师  | 工商系  |
| 刘军   | 男  | 讲师  | 工商系  |
| 罗晓   | 女  | 教授  | 建筑系  |

第六步：将该视图保存为 teacher\_view。

2. 使用 SQL 语句创建视图“student\_view”要求显示学生的学号，姓名，所选课程和成绩

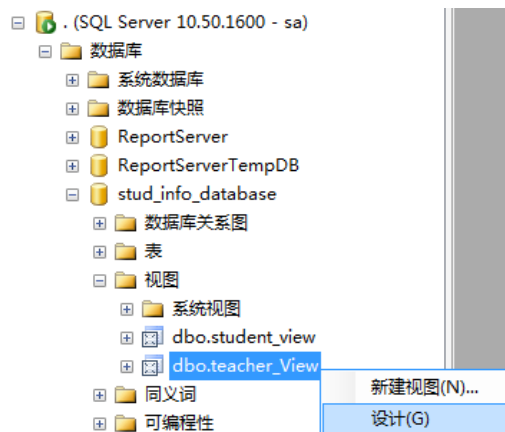
SQL 代码略，

SQL 代码请同学完成测试以后附于工作报告中

3. 使用 SQL Server Managemet Studio 修改视图“teacher\_view”要求显示教师的工号、姓名、性别、民族、职称和所在的系部，并按教师号降序排序

第一步：在 SQL Server Management Studio 的“对象资源管理器”中，依次单击展开节点“数据库”→“stud\_info\_database”→“视图”→“dbo.teacher\_View”用鼠标右键单击“dbo.teacher\_View”节点，在弹出的快捷菜单中选择“设计”命令。





第二步：在 teacher\_tab 表中选择 teacher\_id，在排序类型中选择降序，排序顺序设置为 1，并将 teacher\_id 拖动至最顶端。

第三：单击工具栏中的“查询设计器”→“执行 SQL 命令”执行查询，查询结果如下图所示。

| 教师工号 | 教师姓名 | 性别 | 职称  | 所属系部 |
|------|------|----|-----|------|
| 305  | 罗晓   | 女  | 教授  | 建筑系  |
| 304  | 刘军   | 男  | 讲师  | 工商系  |
| 303  | 张东   | 男  | 双师  | 工商系  |
| 302  | 王海   | 男  | 副教授 | 信息系  |
| 301  | 王梅   | 女  | 无   | 信息系  |

第九步：在“用户向导步骤 2”界面中，“密码”文本框保留为空，单击“下一步”按钮。

第十步：在“用户向导步骤 3”界面中，为域名用户指定根目录，单击“浏览”按钮选择用户根目录，或直接输入“C:\Inetpub\ftproot\software”，选中“锁定用户到根目录”复选框，单击“下一步”按钮。

第四步：保存。

4. 使用 SQL 语句修改视图“student\_view”要求显示学生的学号，姓名，所选课程和成绩，并按成绩降序排序，同成绩按学号升序排序  
SQL 代码略，

SQL 代码请同学完成测试以后附于工作报告中

任务二：存储过程的创建与调用

1. 创建一个简单的存储过程 pro\_stu1，查询所有学生的信息  
SQL 代码如下：

```
create procedure Pro_stu
as
select * from student_tab
调用存储过程

execute Pro_stu
```

| id | stud_id | stud_name | gender_id | nation_id | brthday | Specialty_id | depart_id | tel         | qq         | e_mail | credits | note | stud_sfz           | age |
|----|---------|-----------|-----------|-----------|---------|--------------|-----------|-------------|------------|--------|---------|------|--------------------|-----|
| 1  | 101     | 张海        | 0         | 01        | NULL    | 01           | 01        | 13547856792 | 298451124  | NULL   | 0       | NULL | 500236199701037035 | 20  |
| 2  | 102     | 王艳        | 1         | 01        | NULL    | 01           | 01        | 13102567756 | 156384759  | NULL   | 0       | NULL | 510503199505061243 | 22  |
| 3  | 103     | 叶强        | 0         | 02        | NULL    | 03           | 02        | 18588485612 | 5684235792 | NULL   | 0       | NULL | 500230199606144990 | 21  |
| 4  | 104     | 刘兰        | 1         | 04        | NULL    | 04           | 02        | 15025146983 | 12574684   | NULL   | 0       | NULL | 340823199808051966 | 19  |
| 5  | 105     | 罗小红       | 1         | 01        | NULL    | 06           | 04        | 15388456631 | 65214521   | NULL   | 0       | NULL | 412723199611126424 | 21  |
| 6  | 106     | 林小华       | 0         | 03        | NULL    | 06           | 04        | 17788456631 | 568749562  | NULL   | 0       | NULL | 500383199512127017 | 22  |

2. 创建一个带有输入参数的存储过程 pro\_stu2, 要求根据输入学生的学号来查询相应的学生信息。

SQL 代码如下:

```
create procedure Pro_stu1
@studid varchar(15)
as
select * from student_tab
where stud_id=@studid
调用存储过程
execute Pro_stu1
```

@studid='102'

3. 创建一个带有两输入参数的存储过程 pro\_score1 要求输入两个 0-100 的整数, 查询大于成绩在这两个数之间的学生姓名、所选择课程和成绩

--创建一个带有两输入参数的存储过程pro\_score1要求输入两个-100的整数, 查询大于成绩在这两个数之间的学生姓名、所选择课程和成绩

```
create procedure Pro_score1
@low int,
@top int
as
SELECT student_tab.stud_name AS 姓名,
course_tab.course_name AS 所选课程, score_tab.score AS 成绩
FROM course_tab INNER JOIN score_tab ON course_tab.course_id
= score_tab.course_id INNER JOIN
student_tab ON score_tab.stud_id =student_tab.stud_id
where score_tab.score between @low and @top
```

调用存储过程

```
execute Pro_score1
@low=70,
@top=80
```

| 姓名 | 所选课程  | 成绩    |
|----|-------|-------|
| 张海 | 数据库设计 | 72.00 |
| 王艳 | 服务器配置 | 76.00 |
| 刘兰 | 网络营销  | 75.00 |

4. 创建一个带有输入参数和输出参数的存储过程 pro\_teacher, 其中输入参数用于接收教师工号, 输出参数用于显示该教师的姓名、性别、职称和所在的系部

SQL代码如下:

--创建一个带有输入参数和输出参数的存储过程pro\_teacher, 其中输入参数用于接收教师工号, 输出参数用于显示该教师的姓名、性别、职称和所在的系部

```
create procedure pro_teacher
```

```

@studid varchar(15),
@name varchar(20) output,
@course varchar(20) output,
@grade decimal(5, 2) output
as
    SELECT  @name=student_tab.stud_name,
@course=course_tab.course_name , @grade= score_tab.score
    FROM   course_tab INNER JOIN score_tab ON course_tab.course_id
= score_tab.course_id INNER JOIN
        student_tab ON score_tab.stud_id =student_tab.stud_id
    where student_tab.stud_id=@studid
调用存储过程
declare
    @teacherid varchar(15),
    @name varchar(20),
    @xibi varchar(2),
    @ziche varchar(20),
    @xibu varchar(20)
set @teacherid='302'

execute pro_teacher
    @teacherid=@teacherid,
    @name= @name output,
    @xibi=@xibi output,
    @ziche= @ziche output,
    @xibu=@xibu output
SELECT @name as '姓名',
       @xibi as '性别',
       @ziche as '职称',
       @xibu as '系部'

```

| 姓名 | 性别 | 职称  | 系部  |
|----|----|-----|-----|
| 王海 | 男  | 副教授 | 信息系 |

### 考核标准（技能和素质考核）

#### 1、专业技能考核标准（共 80 分）

- (1) 正确完成视图的创建（20 分）
- (2) 正确完成视图的修改；（20 分）
- (3) 正确完成存储过程的创建；（30 分）
- (4) 正确完成存储过程的调用（10 分）

#### 2、学习态度考核标准（共 20 分）

- (1) 实训报告书质量（5 分）

认真对实训过程进行总结，发现和解决问题；认真按照要求填写；书面整

洁，字迹清楚。

(2) 工作职业道德 (5 分)

安全文明工作，职业操守较好；爱护计算机等公共设施；基本按照布置的工作任务和要求去完成。

(3) 团队合作精神 (5 分)

具有良好的团队合作精神，热心帮助班级或小组的其它成员；能与团队成员有效沟通；能合理分配小组成员的工作任务；

(4) 语言沟通能力 (5 分)

能用专业的语言正确流利地展示项目成果；能准确回答老师提出的相关问题

### 问题与思考

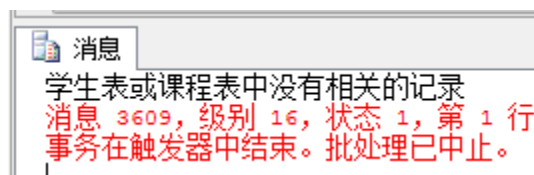
1. 使用 SQL 语句创建视图 “student\_view” 要求显示学生的学号，姓名，所选课程和成绩（代码附于工作报告中）。
2. 使用 SQL 语句修改视图 “student\_view” 要求显示学生的学号，姓名，所选课程和成绩，并按成绩降序排序，同成绩按学号升序排序（代码附于工作报告中）。
3. 创建带有两个参数的存储过程 PRO\_score0，输入一个整数 2，和一个 0-100 的整数，查询有 2（参数 1）门以上的课程是参数 2 分以上的学生学号（代码附于工作报告中）。

## 一体化课程项目工作任务单

| 名 称   | 触发器、游标  | 序号 | 6                |
|---|---|----|------------------|
| 目 标   | 1. 具备 insert 触发器创建的能力<br>2. 具备 update 触发器创建的能力<br>3. 具备 delete 触发器创建的能力<br>4. 具备游标创建的能力 |    |                  |
| 工作要求  | 1. 完成触发器的创建并测试<br>2. 完成游标的创建并测试<br>3. 根据工作过程写出实验报告                                      |    |                  |
| 企业标准  | T—SQL 代码必须符合基本编程规范  |    |                  |
| 工作条件  | 1. 操作系统: Windows 7<br>2. 软件环境: SQL Server 2008 数据库                                      |    |                  |
| 工作流程  |   |    | 注<br>意<br>事<br>项 |
| <p><b>一、工作背景</b></p> <p>abc 学校的信息中心在完成系统开发以后,进一步对系统进行完善,开发小组提出需要创建触发器、游标。</p> <p><b>二、工作内容</b></p> <p>任务一: 触发器的创建</p> <ol style="list-style-type: none"> <li>1. insert 触发器创建</li> <li>2. update 触发器创建</li> <li>3. delete 触发器创建</li> </ol> <p>任务二: 游标的创建</p> <ol style="list-style-type: none"> <li>1. 创建一个游标 crs_student_bri, 根据学生表中的身份证号得到学生的出生日期, 并根据得到的出生日期更新学生表中的 brithday</li> <li>2. 创建一个游标 crs_score_credit, 如果学生成绩在 60 分以上 (含 60), 则将计算所有合格课程的学分和保存于 credits 列</li> </ol> <p><b>三、工作流程</b></p> <p>任务一: 触发器的创建</p> <ol style="list-style-type: none"> <li>1. insert 触发器创建</li> </ol> <p>要求: 创建 insert 触发器 insert_score_tri, 实现向成绩表中添加记录时, 首先检查该行的学号列在学生表中和课程表中是否存在, 如有一项不成立, 则不允许插入, 并提示 "学生表或课程表中没有相关的记录"</p> |   |    |                  |

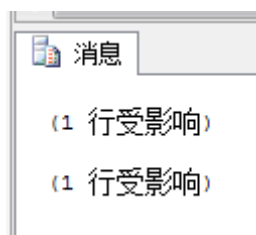
代码:

```
--检测触发器是否存在，触发器存放在系统表sysobjects中
if exists(select name from sysobjects where name='insert_score_tri'
and type='tr')
drop trigger insert_score_tri
go
create trigger insert_score_tri on score_tab for insert
as
    if exists(select * from inserted where inserted.stud_id not in
(select stud_id from student_tab) or inserted.course_id not in (select
course_id from course_tab))
    begin
        print('学生表或课程表中没有相关的记录')
        rollback
    end
测试
插入不满足条件的值
insert into score_tab values('105','1005',98)
```



插入满足条件的值

```
insert into score_tab values('105','100105',98)
```



## 2. update 触发器创建

要求：创建 update 触发器 update\_student\_tri，当更新了学生表中某位学生的学号时，自动更新成绩表中的学号

代码:

```
--检测触发器是否存在，触发器存放在系统表sysobjects中
if exists(select name from sysobjects where
name='update_student_tri' and type='tr')
drop trigger update_student_tri
go
create trigger update_student_tri on student_tab
for update
as
    declare @new_Stud_id varchar(15) --声明变量，更新后的学号
```

```

declare @Stud_id varchar(15)      --声明变量, 未更新的学号
select @Stud_id=deleted.stud_id,@new_Stud_id=inserted.stud_id
from deleted,inserted
where deleted.stud_name=inserted.stud_name
update score_tab
set stud_id=@new_Stud_id
where stud_id=@Stud_id
--提示信息
print '已将成绩表中的学生学号为'+@stud_id+'更新为' +@new_stud_id
测试:
将学生罗小红的学生号改为 205

```

```

update student_tab set stud_id='205' where stud_name='罗小红'

```

(1 行受影响)  
 已将成绩表中的学生学号为105更新为205  
 (1 行受影响)

分别查询学生表与成绩表的信息

| 消息      |           |           |           | stud_id | course_id | score |
|---------|-----------|-----------|-----------|---------|-----------|-------|
| stud_id | stud_name | gender_id | nation_id | 101     | 100103    | 55.00 |
| 101     | 张海        | 0         | 01        | 102     | 100101    | 88.00 |
| 102     | 王艳        | 1         | 01        | 102     | 100102    | 92.00 |
| 103     | 叶强        | 0         | 02        | 102     | 100103    | 76.00 |
| 104     | 刘兰        | 1         | 04        | 103     | 100105    | 85.00 |
| 106     | 林小华       | 0         | 03        | 103     | 100106    | 69.00 |
| 205     | 罗小红       | 1         | 01        | 104     | 100105    | 75.00 |
|         |           |           |           | 104     | 100106    | 85.00 |
|         |           |           |           | 205     | 100110    | 52.00 |

### 3.delete 触发器创建

要求: 创建 delete 触发器 del\_student\_tri, 当删除学生表中的学生信息时, 自动将成绩表的中学生成绩删除

代码:

```

if exists(select name from sysobjects where name='del_student_tri'
and type='tr')
drop trigger del_student_tri
go
create trigger del_student_tri on student_tab
for delete
as
declare @Stud_id varchar(15)
select @Stud_id=deleted.stud_id from deleted
delete from score_tab where stud_id in (select stud_id from deleted)
print '已删除学号为'+@stud_id+'的所有成绩信息'

```

测试

在学生表中删除姓名为张海的信息

```
delete from student_tab where stud_name='张海'
```

(3 行受影响)

已删除学号为101的所有成绩信息

(1 行受影响)

查看学生与成绩表的结果

| 结果 |         | 消息        |           |           |
|----|---------|-----------|-----------|-----------|
| id | stud_id | stud_name | gender_id | nation_id |
| 2  | 102     | 王艳        | 1         | 01        |
| 3  | 103     | 叶强        | 0         | 02        |
| 4  | 104     | 刘兰        | 1         | 04        |
| 6  | 106     | 林小华       | 0         | 03        |
| 5  | 205     | 罗小红       | 1         | 01        |

| stud_id | course_id | score |
|---------|-----------|-------|
| 102     | 100101    | 88.00 |
| 102     | 100102    | 92.00 |
| 102     | 100103    | 76.00 |
| 103     | 100105    | 85.00 |
| 103     | 100106    | 69.00 |
| 104     | 100105    | 75.00 |
| 104     | 100106    | 85.00 |
| 205     | 100110    | 52.00 |

任务二：游标的创建

1. 创建一个游标 crs\_student\_bri，根据学生表中的身份证号得到学生的出生日期，并根据得到的出生日期更新学生表中的 brithday。

代码：


```
declare @sfzh varchar(18)--定义变量，身份证号
declare @brithday date--定义变量，出生日期
--用游标将crs_student_bri表中的数据读出
declare crs_student_bri cursor --crs_student_bri游标名
for
select stud_sfz --定义选择的字段
from student_tab order by stud_id --从哪个表中
open crs_student_bri --打开定义的游标
fetch next from crs_student_bri into @sfzh
while @@fetch_status=0
begin
set
@brithday=substring(@sfzh,7,4)+'/'+substring(@sfzh,11,2)+'/'+sub
string(@sfzh,13,2)
update student_tab set brithday=@brithday where stud_sfz=@sfzh
fetch next from crs_student_bri into @sfzh
end
```



```
close crs_student_bri
deallocate crs_student_bri
测试
```

```
select * from student tab
```

结果

消息

| id | stud_id | stud_name | gender_id | nation_id | brithday   |
|----|---------|-----------|-----------|-----------|------------|
| 2  | 102     | 王艳        | 1         | 01        | 1995-05-06 |
| 3  | 103     | 叶强        | 0         | 02        | 1996-06-14 |
| 4  | 104     | 刘兰        | 1         | 04        | 1998-08-05 |
| 6  | 106     | 林小华       | 0         | 03        | 1995-12-12 |
| 5  | 205     | 罗小红       | 1         | 01        | 1996-11-12 |

2. 创建一个游标 crs\_score\_credit, 如果学生成绩在 60 分以上 (含 60), 则将计算所有合格课程的学分和保存于 credits 列。

```
update student_tab set credits=0
declare @credit int
declare @stud_id varchar(15)
--用游标将crs_score_credit表中的数据读出
declare crs_score_credit cursor --crs_score_credit游标名
for
    SELECT student_tab.stud_id,sum(course_tab.credits)
    FROM course_tab INNER JOIN score_tab ON course_tab.course_id =
score_tab.course_id INNER JOIN
        student_tab ON score_tab.stud_id =student_tab.stud_id
    where score_tab.score>=60
    group by student_tab.stud_id
open crs_score_credit --打开定义的游标
fetch next from crs_score_credit into @stud_id, @credit
while @@fetch_status=0
begin

update student_tab set credits=@credit where stud_id=@stud_id
fetch next from crs_score_credit into @stud_id, @credit
end
close crs_score_credit
deallocate crs_score_credit
```

测试:

以学生王艳为例

第一步: 查看学生的学分信息

```
select stud_id,stud_name,credits from student tab
```

结果

| stud_id | stud_name | credits |
|---------|-----------|---------|
| 102     | 王艳        | 11.0    |
| 103     | 叶强        | 6.0     |
| 104     | 刘兰        | 6.0     |
| 106     | 林小华       | 0.0     |
| 205     | 罗小红       | 0.0     |

从表中得知，王艳的学分为 11 分

第二步：验证学生的学分信息是否正确

```
select * from score_tab
```

| stud_id | course_id | score |
|---------|-----------|-------|
| 102     | 100101    | 88.00 |
| 102     | 100102    | 92.00 |
| 102     | 100103    | 76.00 |

从表中得知，王艳选择了 100101, 100101, 100103 三门课程并且均合格，再查询这三门课程的学分

```
select course_id,credits from course_tab where course_id in('100101','100102','100103')
```

| course_id | credits |
|-----------|---------|
| 100101    | 3.0     |
| 100102    | 4.0     |
| 100103    | 4.0     |

所以三门课程的学分总共是  $3+4+4=11$ ，程序正确

### 考核标准（技能和素质考核）

#### 1、专业技能考核标准（共 80 分）

（1）正确完成触发器的创建并测试（50 分）

（2）正确完成游标的创建并测试；（30 分）

#### 2、学习态度考核标准（共 20 分）

（1）实训报告书质量（5 分）

认真对实训过程进行总结，发现和解决问题；认真按照要求填写；书面整洁，字迹清楚。

（2）工作职业道德（5 分）

安全文明工作，职业操守较好；爱护计算机等公共设施；基本按照布置的工作任务和要求去完成。

（3）团队合作精神（5 分）

具有良好的团队合作精神，热心帮助班级或小组的其它成员；能与团队成员有效沟通；能合理分配小组成员的工作任务；

（4）语言沟通能力（5 分）

能用专业的语言正确流利地展示项目成果；能准确回答老师提出的相关问题。

### 问题与思考

1. 创建 insert 触发器 up\_credit\_tri, 实现向成绩表中添加记录时, 如果学生成绩在 60 分以上 (含 60), 则学生的学分自动累加到学生表中 (代码附于工作报告中)。
2. 创建一个游标 crs\_student\_mail, 根据学生表中 qq 得到学生的 qq 邮箱, 并根据得到的 qq 邮箱更新学生表中的 e\_mail 列, 请注意邮箱的格式 (代码附于工作报告中)。