高职高专电子商务专业系列教材

网络数据库

主　编　乔冰琴
副主编　胡　振　谭海军

上海财经大学出版社

需要全本请访问 www.ertongbook.com

图书在版编目(CIP)数据

网络数据库/乔冰琴主编;胡振,谭海军副主编. —上海:上海财经大学出版社,2007.8

(高职高专电子商务专业系列教材)

ISBN 978-7-5642-0049-7/F·0049

Ⅰ.网… Ⅱ.①乔… ②胡… ③谭… Ⅲ.关系数据库-数据库管理系统-高等学校:技术学校-教材 Ⅳ.TP311.138

中国版本图书馆 CIP 数据核字(2007)第 125195 号

□ 责任编辑 刘 兵

□ 封面设计 周卫民

WANGLUO SHUJUKU

网络数据库

乔冰琴 主编

胡 振
谭海军 副主编

上海财经大学出版社出版发行

(上海市武东路 321 号乙 邮编 200434)

网 址:http://www.sufep.com

电子邮箱:webmaster@sufep.com

全国新华书店经销

上海崇明裕安印刷厂印刷装订

2007 年 8 月第 1 版 2007 年 8 月第 1 次印刷

787mm×960mm 1/16 27 印张 558 千字

印数:0 001—4 000 定价:33.00 元

前　言

Microsoft SQL Server 2005 是微软公司的最新数据库产品，其凭借在企业级数据管理、开发工作效率和商业智能方面的出色表现，赢得了众多客户青睐，成为目前惟一能够真正胜任从低端到高端任何数据应用的企业级数据平台。据互联网数据中心（Internet Data Center，IDC）统计，中国数据库市场平均年增长率保持在 10%～12%，而 MS SQL Server 2005 在发布的第一年就创下了 3 倍于市场平均增长率的增长速度，重新定义了企业级市场新格局。

为了帮助读者快速高效的学习掌握数据库基本原理和 MS SQL Server 2005 的基本使用方法和开发技巧，本书编写组组织了多所高校长期从事数据库原理和 MS SQL Server 2005 教学工作的教师编写本书。

本书可以分两个部分：第一部分为第 1～14 章，主要介绍数据库基本原理和 MS SQL Server 2005 使用方法和技巧；第二部分为第 15 章——案例实训，该章根据微软提供的数据库开发标准流程，从实战的角度指导读者开发一个简单的应用数据库，以便读者学以致用。

本书结构连贯，强调理论以够用为度，始终以介绍数据库的开发技能为主线。在编写过程中，编者总结并借鉴了多年实践教学经验，以“案例驱动”方式组稿全书。为了减少读者不断适应新的案例所花费的时间和精力，全书用一个小型的完整案例贯穿全书，理论联系实际，每一章节的实例都是案例实现的一部分。考虑到初学者对 MS SQL Server 2005 比较陌生，书中还提供了大量的图表，以便于引导读者阅读和理解。

为了使读者及时巩固学习效果，每个章节均配有相应的上机指导和难易程度不一的习题，读者可以在学习完每一个章节后，在上机指导的指导下上机实践，以加强动手能力。

教师可根据教学要求、课时多少、授课对象灵活选取授课内容，为了方便老师教学，本书提供配套的电子教案、案例数据库、案例源代码程序、习题参考答案、试题库等教学资料，一并在 http://www.jdkj.net/sql2005/ 下载，或发邮件至 qiaobingqin@sxftc.edu.cn

索取。

本书叙述详尽，由浅入深，层次分明，适合作为高职高专计算机类专业、与计算机相关类专业的数据库基础或 SQL Server 基础课程的教材或参考书，也可作为其他学习 MS SQL Server 2005 的工程技术人员自学的教材或参考书。

本书由乔冰琴担任主编，胡振、谭海军担任副主编。各章编写分工如下：第 1 章、第 15 章由山西省财政税务专科学校乔冰琴编写，第 2 章、第 6 章由天津农学院甄爱军编写，第 3 章由山西省财政税务专科学校蔺建霞编写，第 4 章、第 5 章由重庆工业职业技术学院刘毅编写，第 7 章、第 8 章由重庆电子职业技术学院胡振编写，第 9 章由成都东软信息技术学院熊永福编写，第 10 章由山西省财政税务专科学校李琳编写，第 11 章、第 12 章由重庆电子职业技术学院陈磊编写，第 13 章、第 14 章由长江师范学院谭海军编写。

本书承蒙上海财经大学出版社编辑的审阅，并提出了许多宝贵的意见，在此向他们表示最衷心、最诚挚的谢意！

由于本书涉及面广，加之作者水平、经验有限，书中难免存在一些错误和缺点，敬请读者批评指正！

编　者

2007 年

第 8 章　表数据操作

前一章讨论了使用 SELECT 语句在表中查询数据，但是 SELECT 语句只能检索数据，不能在表中添加、修改、删除数据，若要实现这些功能，则需要通过 INSERT、UPDATE 和 DELETE 等语句来实现，这些操作也称为插入查询、更新查询和删除查询，统称为操作查询。操作查询能够对数据库的内容进行更改，但不能返回结果集。

8.1　添加数据

数据库中的表建好了以后，可以使用 INSERT 语句添加单条记录，也可以使用 INSERT …SELECT 语句添加子查询结果集。无论采用哪种方式添加数据，添加的数据都必须符合相应列的数据类型、约束等，以保证表中数据的完整性。

8.1.1　INSERT 语句

使用 INSERT 语句可以向已建好的表中每次添加一条记录。该语句的基本语法格式如下。

```
INSERT [INTO] <目标表名> [(列名列表)] VALUES (<值列表>)
```

其中：

- 目标表名，是指接受数据的表名。
- 列名列表，欲插入数据的各个列。这些列必须在目标表中存在，顺序可以任意；当对目标表中所有列都添加数据时，列名列表可以省略。列名列表必须用圆括号括起来。
- 值列表，欲添加的数据值。各数据值与<列名列表>中列的顺序一一对应，且数据都必须符合相应列的数据类型、约束等，以保证表中数据的完整性。若对新记录的所有列都添加数据，并省略了列名列表，则值列表中数据的排列顺序要与目标表中列的顺序一致。值列表也要用圆括号括起来。

使用 INSERT 语句时要注意以下几点。

需要全本请访问 www.ertongbook.com

(1)必须为主键、不允许为空的列添加数据。

(2)添加的数据必须符合相应列的数据类型、约束,以保证表中数据的完整性。

(3)若添加的数据是 char、varchar、datetime 数据类型,则该值要用单引号括起来。

(4)对于指定有默认值的列,若要使该列获得其默认值,则在值列表的相应位置上使用 DEFAULT 关键字。

下面分别来看一下添加数据的几种情况。

1. 为新记录的每个列都添加数据

为新记录的每个列都添加数据,可以省略列名列表。若省略了列名列表,则值列表中数据的排列顺序要与目标表设计的字段顺序一致;若不省略列名列表,在列名列表中各列的排列顺序可以是任意的,但值列表中数据要与列名列表中各列一一对应。

例 8－1　在学生表中添加学号为 15003、姓名是张三的男生、汉族、生日 1987-12-1、入学时间 2004-9-1、所在班级为 2004015、重庆籍、备注信息为空的学生信息。

分析:题目要求在案例数据库的表 Students 中添加数据,该表中的所有字段及其顺序是 Student_id、Student_name、Student_sex、Student_nation、Student_birthday、Student_time、Student_classid、Student_home、Student_else,要求输入的数据涉及到了所有的字段,即要为新记录的所有字段添加数据,则<列名列表>可以省略,<值列表>中数据的顺序要与上述字段的先后顺序一致。语句及执行结果如图 8－1 所示,从图中可以看到在消息区有"(1 行受影响)"的提示,即表示成功添加了一条记录。

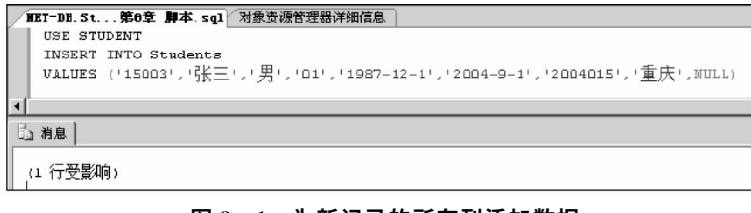


图 8－1　为新记录的所有列添加数据

可用图 8－2 中语句查看目的表中新插入的记录。语句执行结果如图 8－2 所示。

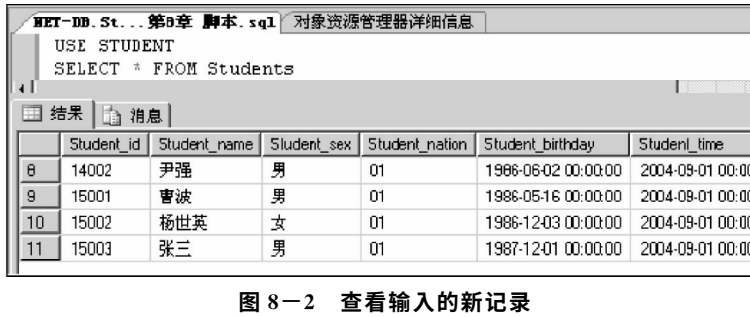


图 8－2　查看输入的新记录

2. 给新记录的部分列添加数据

也可以只为新记录的部分列添加数据，对于没有添加数据的列，系统将按照下面的规则为其填充数据。

(1)设置有默认值的列填充其默认值。

(2)具有 IDENTITY 属性的列，且该表 IDENTITY_INSERT 选项的参数是 OFF(默认值是 OFF)，使用下一个增量标识值。

(3)数据类型为 timestamp 的列，不必为其添加数据，系统自动使用当前的时间戳值。

(4)若该列允许为空，则使用空值 NULL。

若上述四条都不符合，则记录添加失败，新记录不能添加到目标表中。

例 8－2　在学生表中添加学号为 15004、姓名是李四的男生、所在班级为 2004015 的学生信息。

分析：要求在案例数据库的表 Students 中添加数据，题目提供了学号、姓名、性别、所在班级的信息，即要为新记录的部分字段添加数据，在列名列表中填写上述字段名(Student_id，Student_name，Student_sex，Student_classid)，顺序不限，但在＜值列表＞中数据的顺序要与＜列名列表＞中字段的先后顺序一一对应。语句如下。

```
USE Student
INSERT INTO Students
    (Student_id,Student_name,Student_sex,Student_classid)
    VALUES ('15004','李四','男','2004015')
```

3. 给新记录的每个字段都使用默认值

如果想在新行的全部列中都使用创建表时所指定的默认值，则 INSERT 语句可以写成下面的形式。

```
INSERT INTO ＜目标表名＞ DEFAULT VALUES
```

4. 为标识列添加数值

若欲添加新记录的目标表有标识列，则要注意该表的 IDENTITY_INSERT 选项，该选项用于控制能否为标识列指定数值插入，每个表的 IDENTITY_INSERT 选项的默认值都是 OFF。

若表的 IDENTITY_INSERT 选项的参数是 OFF，表示不能为标识列输入指定数值，只能由系统自动填充，用 INSERT 语句添加记录时，在值列表中直接忽略标识列的数值。

自动填充标识列数据的规则：每次添加新记录时，系统会在前一次自动填充数值的基础上(注意：不是表中自动标识列的最大值)，按标识列数值递增的规律计算出一个新值作为新记录的标识列的填充数据。

例 8－3　在学生成绩表中添加一条学号为“11001”、第 3 个学期开设的课程号为

“4001”的成绩为 85 的新成绩记录。

分析：成绩表中有一字段 SC_id 是标识列，在默认情况下，IDENTITY_INSERT 选项的默认值是 OFF，因添加的数据涉及到除了标识列以外的所有字段，故可以省略列名列表，但在＜值列表＞中不用为标识列 SC_id 指定数据。语句如下。

```
USE Student
INSERT INTO Student_course
        VALUES ('11001','4001',85,3)
```

若表的 IDENTITY_INSERT 选项的参数值设置为 ON，则添加新记录时必须为标识列指定要输入的数值，系统不再为其自动填充。

可用下面的语句设置表的 IDENTITY_INSERT 选项的参数。

```
SET IDENTITY_INSERT ＜表名＞ ON|OFF
```

8.1.2　INSERT …SELECT 语句

在 INSERT 语句中使用 SELECT 子查询，可以将子查询结果集中的记录全部插入到目标表中。

其基本语法格式如下。

```
INSERT [INTO] ＜目标表名＞
[(＜列名列表＞)]
SELECT ＜选择列表＞ FROM 源表名
[WHERE ＜逻辑表达式＞]
```

使用 INSERT …SELECT 语句时，应注意下面几点。

(1)INSERT 的＜列名列表＞要放在圆括号里面，且不使用 VALUES。

(2)不要将 SELECT 子查询写在圆括号里面，这一点和前面介绍的子查询不一样，要特别注意。

(3)若 SELECT 子句的＜选择列表＞和目标表的结构完全一样，则可以省略 INSERT 子句的＜列名列表＞。

(4)INSERT 子句的＜列名列表＞和 SELECT 子句的＜选择列表＞要注意按顺序一一对应，若省略了 INSERT 的＜列名列表＞，则 SELECT 子句的＜选择列表＞要和目标表的字段的顺序一一对应，相互对应字段的数据类型、长度、精度等要一致。

例 8－4　在 Student 数据库中建立一个新表 ABC，在新表中添加学生表所有记录的学号、姓名、性别、班级信息。

分析：新表 ABC 中存放学生的学号、姓名、性别、班级等信息，则表 ABC 只需要这几个字段即可，而且这些字段和学生表 Students 中对应字段的数据类型等要一致，比较便

捷的方法就是先用 SELECT 语句，以表 Students 为源表，建一个空表，语句如下。

```
USE Student
SELECT Student_id,Student_name,Student_sex,Student_classid
INTO ABC FROM Students
        WHERE 1=2
```

语句中条件“1=2”显然不成立，所以结果集为空，即建了一个空表。然后用下面的语句添加记录。

```
USE Student
INSERT INTO ABC
SELECT Student_id,Student_name,Student_sex,Student_classid
       FROM Students
```

8.2　修改数据

有时需要修改表中现有的数据，在 SQL Server 2005 中，可以用 UPDATE 语句或 UPDATE…FROM 语句来实现这个功能。

8.2.1　UPDATE 语句

UPDATE 语句可以修改表中的一条或者多条记录，其语法格式如下。

```
UPDATE <目标表名>
SET {列名=表达式|NULL|DEFAULT} [,...n]
[WHERE 搜索条件]
```

其中：

- <目标表名>是欲修改记录的表。
- SET 子句指定要修改的列和所用的数据。修改后的数据由表达式确定，但是必须要符合相应字段的数据类型、长度、精度和约束，以保证数据的完整性。另外，对具有默认值的列可以使用 DEFAULT 修改为默认值；对允许使用空值的列，也可以使用 NULL 修改为空值。
- WHERE 子句用来筛选记录，只对符合搜索条件的记录修改，当省略 WHERE 子句时，将对表中的所有记录都修改。

例 8－5　将成绩表中课程代码为 4001 的所有成绩在 70 分以下（不含 70 分）的成绩都加 5 分，语句如下。

```
USE Student
UPDATE Student_course
SET Student_Grade=Student_Grade+5
        WHERE Course_id='4001' AND Student_Grade<70
```

8.2.2　UPDATE…FROM 语句

UPDATE…FROM 语句可扩展 UPDATE 语句的功能，可扩展功能主要表现在以下两个方面：

(1)搜索条件可以扩大到其他表中。

(2)在 SET 子句的表达式中可以从其他表中来获取数据。

UPDATE…FROM 语句的语法如下。

```
UPDATE <目标表名>
SET {列名=表达式|NULL|DEFAULT} [,...n]
FROM <表名>
[WHERE 搜索条件]
```

其中，FROM 子句用来指定其他表。

例 8-6　将姓名为叶海平的学生的电子商务基础课程的成绩改为 88 分。

分析：要在成绩表 Student_course 中修改成绩，但是姓名在学生表 Students 中，课程名在课程表 Courses 中，所以在 UPDATE 语句中需要用到 FROM 子句。语句如下。

```
USE Student
UPDATE Student_course
SET Student_grade=88
        FROM Students AS ST
        INNER JOIN Student_course AS SC
            ON SC.Student_id=ST.Student_id
        INNER JOIN Courses AS CS
            ON CS.Course_id=SC.Course_id
         WHERE CS.Course_name='电子商务基础' AND ST.Student_name=
'叶海平'
```

8.3　删除数据

对于表中已经确定无用的记录，可以用 DELETE 或者 TRUNCATE TABLE 语句

删除。

8.3.1　DELETE 语句

DELETE 语句可以从表中删除一条或者多条记录，语法格式如下。

```
DELETE [FROM] <目标表名>
[WHERE <搜索条件>]
```

其中：

- FROM 关键字可选。
- <目标表名>是要从中删除记录的表。
- WHERE 子句用来筛选满足<搜索条件>的记录。该子句可选：若不选，则表示删除表中的所有记录；若选，则删除满足条件的记录。

例 8－7　在成绩表中删除学号为 13001 的学生的所有成绩。

分析：在成绩表 Student_course 中删除数据，但不是删除所有数据，是删除学号为 13001 的同学的所有成绩，所以需要使用 WHERE 子句来筛选。语句如下。

```
USE Student
DELETE Student_course
        WHERE Student_id='13001'
```

8.3.2　DELETE…FROM 语句

有时候要删除记录的搜索条件不在目标表中，而是在其他表中，则需要在 DELETE 语句中使用 FROM 子句指定其他表，以便扩充搜索记录的条件，基本语法如下。

```
DELETE [FROM] <目标表名>
FROM <表名>
[WHERE <搜索条件>]
```

其中，FROM 子句用来指定其他表，以便扩充搜索记录的条件，其他说明与 DELETE 语法中的一样，在此不再复述。

注意：不要将 FROM 关键字和 FROM 子句混淆。

例 8－8　在成绩表中删除安静同学的所有成绩。

分析：题目要删除姓名是安静的同学的所有成绩，但是在成绩表 Student_course 中没有学生姓名字段，而在学生表 Students 中有学生姓名字段，所以需要使用 FROM 子句来扩充 WHERE 子句的搜索条件。在学生表 Student_course 和成绩表 Students 中，有两个相同意义的字段——学号 Student_id，故这两个表可以通过学号字段进行内连接，以便扩充搜索条件，语句如下。

需要全本请访问 www.ertongbook.com

```
USE Student
DELETE Student_course
        FROM Student_course AS SC INNER JOIN Students AS ST
        ON SC. Student_id=ST. Student_id
        WHERE Student_name='安静'
```

8.3.3　TRUNCATE TABLE 语句

可以使用 TRUNCATE TABLE 语句删除表中的所有记录，该语句的语法格式如下。

```
TRUNCATE TABLE <目标表名>
```

TRUNCATE TABLE 语句只是删除目标表中的所有记录，执行的结果和下面的语句（不带 WHERE 的 DELETE）是等效的。

```
DELETE <目标表名>
```

但是 TRUNCATE TABLE 语句执行的速度要更快，占用的日志更少，这是因为 TRUNCATE TABLE 直接释放数据页，并且在事务日志中也只记录数据页的释放，而 DELETE 是一行一行地删除，在事务日志中也要记录每一条记录的删除。

TRUNCATE TABLE 不能完全代替不带 WHERE 的 DELETE 命令，不能替代的情况有如下几种。

（1）要保留标识的情况下不能用 TRUNCATE TABLE，因为 TRUNCATE TABLE 会重置标识。

（2）需要使用触发器的情况下不能使用 TRUNCATE TABLE ，它不会激发触发器。

（3）对于由 FOREIGN KEY 约束引用的表（即主键所在的表，不是外键所在的表）不能使用 TRUNCATE TABLE。

（4）对于参与了索引视图的表不能使用 TRUNCATE TABLE 。注意：是指索引视图，并非普通视图。

例 8－9　删除表 ABC 中的所有记录，语句如下。

```
USE Student
TRUNCATE TABLE ABC
```

也可以用如下语句，效果一样。

```
USE Student
DELETE ABC
```

8.4　数据的导入与导出

任何数据库都需要和外界交换数据，数据的导入与导出是数据库管理的基本要求。在 SQL Server 2005 中，可以将 SQL Server 中的数据导出到其他的数据源中，例如 Access 数据库、Excel 工作表、文本文件等，也可以将其他数据源中的数据导入到 SQL Server 中的数据库中。在 SQL Server 2005 中，把从外界其他数据源输入数据称为导入，从 SQL Server 的数据库中输出数据到其他数据源称为导出。

下面以将 SQL Server 中的数据导出到 Excel 工作表中为例，介绍 SQL Server 的数据导出和导入方法。

例 8－10　将学生信息表 Students 中的所有学生信息导出到 Excel 工作表中。

具体操作步骤如下。

(1)从【开始】菜单上选择【程序】|【Microsoft SQL Server 2005】|【SQL Server Management Studio】。

(2)使用【windows 身份验证】建立连接，在【对象资源管理器】中展开【数据库】。

(3)用鼠标选中案例数据库 Student，单击右键，在右键菜单中选择【任务】命令，在其子菜单中执行【导出数据(X)…】，弹出如图 8－3 所示的【SQL Server 导入和导出向导】对话框，单击【下一步】按钮，弹出如图 8－4 所示的【选择数据源】对话框。



图 8－3　SQL Server 导入和导出向导对话框



图 8－4　选择数据源

(4)在图 8－4 所示的对话框中选择数据源，即导入数据的来源，在本例中数据源使用默认的“SQL Native Client”，数据库选择本书案例数据库“Student”，单击【下一步】按钮，弹出如图 8－5 所示的【选择目标】对话框。



图 8－5　选择目标

(5)在图 8－5 的对话框中，可设置目标数据库的相关选项。因要将数据导出在 Excel 表中，故目标选择“Microsoft Excel”，在【Excel 连接设置】中指定目标 Excel 文件的路径、文件名和版本(可单击【浏览】按钮，用对话框的形式指定目标 Excel 文件的路径和文件

名)。若选中复选框【首行包含列名称】,则在 Excel 文件的第一行出现相应的字段名,各项设置好后单击【下一步】按钮,弹出如图 8－6 所示的【指定表复制或查询】对话框。



图 8－6　指定表复制或查询

(6)在图 8－6 所示的对话框中有两个选项:【复制一个或多个表或视图的数据】,此选项用于复制源数据库中现有表或视图的全部数据;【编写查询以查询指定要传输的数据】,此选项用于编写 SQL 查询,以便对复制操作的源数据进行操纵或限制,进行有条件的导入导出。本例在此选项卡中选择第一个选项,单击【下一步】按钮,弹出如图 8－7 所示【选择源表和源视图】对话框。



图 8－7　选择源表和源视图

(7)在图 8－7 所示的对话框中,可以选择要导出数据的表或者视图。选中“Students”表,单击【编辑映射】按钮,弹出如图 8－8 所示的【列映射】对话框。

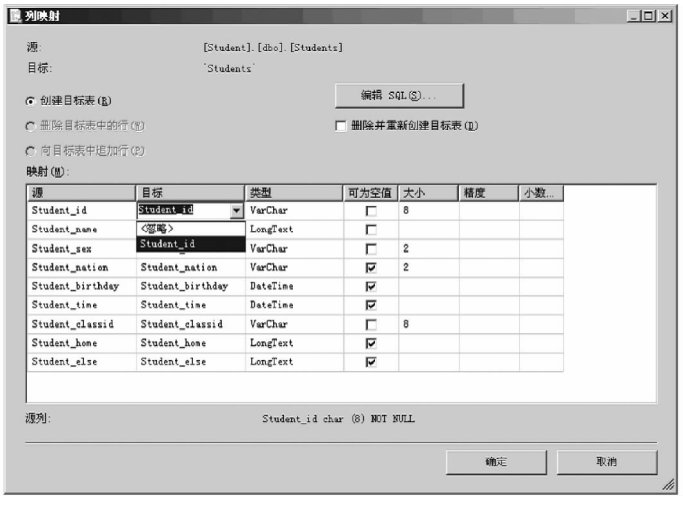


图 8－8　编辑列映射

(8)在图 8－8 所示的对话框中,可以编辑源表和目标表的列映射。【源】列是数据源表或源视图的字段名,【目标】列是对应输出到目标表中的字段,若不输出某字段,则可在对应的【目标】字段名上选择“忽略”,也可修改目标表的类型、可为空值等,修改各项参数后单击【确定】,可回到图 8－7 所示的对话框。

(9)在图 8－7 所示的对话框中,单击【预览】按钮,可预览从源表或源视图中导出的数据,如图 8－9 所示。单击【确定】,可回到图 8－7 所示的对话框。



图 8－9　预览数据

(10)在图 8－7 所示的对话框中选择【下一步】,可弹出如图 8－10 所示的【保存并执行包】对话框,在此对话框中选择【立即执行】,单击【下一步】,弹出如图 8－11 所示的【完成该向导】对话框。



图 8－10　保存并执行包

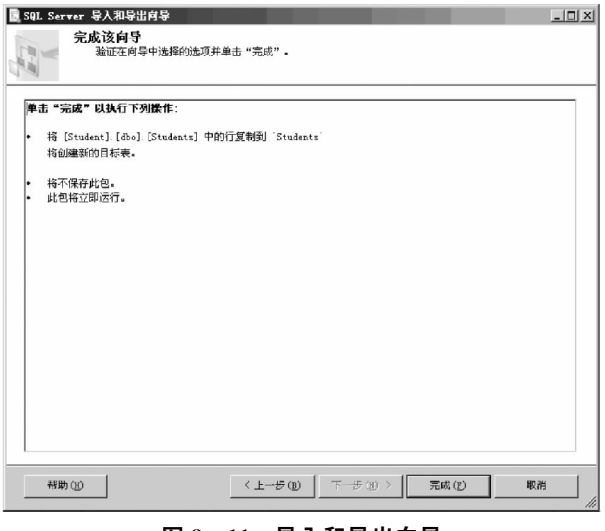


图 8－11　导入和导出向导

(11)在图 8－11 所示的对话框中,验证向导中选择的选项,若还有需要修改的选项,则可单击【上一步】,回到相应的对话框中继续修改选项;若确认无误,可单击【完成】按钮,执行数据导出操作。

需要全本请访问 www.ertongbook.com