

# 数据库 原理与应用

主 编 于艳东 姚玉阁 马占飞  
副主编 林 民 马志强 吴和群

湖南大学出版社·长沙

---

**图书在版编目(CIP)数据**

数据库原理与应用 / 于艳东, 姚玉阁, 马占飞主编.

长沙: 湖南大学出版社, 2025. 4. -- ISBN 978-7-5667-4189-9

I. TP311.13

中国国家版本馆 CIP 数据核字第 2025WM1017 号

---

**数据库原理与应用**

SHUJUKU YUANLI YU YINGYONG

---

**主 编:** 于艳东 姚玉阁 马占飞

**责任编辑:** 方雨轩

**印 装:** 天津市蓟县宏图印务有限公司

**开 本:** 787 mm×1092 mm 1/16

**印 张:** 14.5

**字 数:** 309 千字

**版 次:** 2025 年 4 月第 1 版

**印 次:** 2025 年 4 月第 1 次印刷

**书 号:** ISBN 978-7-5667-4189-9

**定 价:** 58.00 元

---

**出 版 人:** 李文邦

**出版发行:** 湖南大学出版社

**社 址:** 湖南·长沙·岳麓山

**邮 编:** 410082

**电 话:** 0731-88822559(营销部), 88821343(编辑室), 88821006(出版部)

**传 真:** 0731-88822264(总编室)

**网 址:** <http://press.hnu.edu.cn>

**电子邮箱:** 501267812@qq.com

---

版权所有, 盗版必究  
图书凡有印装差错, 请与营销部联系

本书介绍了 SQL Server 2022 数据库管理系统的基本原理和基本技术。在内容编排上,摒弃了孤立介绍知识点的编排模式,为培养学生的“大工程观”,以学生诉求和社会发展需求为观测点,主动融入时代元素,融入 CDIO (conceive-design-implement-operate, 构思-设计-实施-运作)工程教育理念,按“案例诠释理论、项目推动实践”的思路,以能力培养为主线,以案例教学为引导,通过项目任务带动知识点的学习,从而激发学生学习兴趣,真正实现“教、学、做”的有机融合。我们希望通过教材的不断改进来推动职业教育的教法改革。

### 一、本书特色

1) 基于产教融合理念,由校企“双元”合作开发,共同打造具有真实企业情景的项目化、任务驱动式理实一体化教材,推广项目化、情景化、模块化教学。

2) 配合混合式教学、在线学习等泛在教学改革的需要,采用“纸质教材+电子活页”的创新形式编写。纸质教材以项目和任务为载体,以工作过程为导向,以职业素养和职业能力培养为重点。电子活页以电子资源为主,包括视频、音频、作业、试卷、拓展资源等,充分利用云平台,融合信息技术,提高学生学习效率,实现碎片化学习。

3) 将“立德树人”基本要求贯彻于教材编写的全过程,以润物无声的方式将思政教育与专业知识培养有机融合。

### 二、本书内容设计

全书共分六个项目,项目一、项目二和项目三介绍了 SQL Server 2022 的基础知识,项目四介绍了 SQL 和人工智能的相关知识,项目五和项目六从具体实例出发,介绍了 SQL Server 2022 的具体应用。

六个项目的内容如下：

项目一：主要介绍数据库的创建和维护，并具体讲解了数据库表的创建、数据库表的修改和删除、主键和外键的创建和删除等知识。

项目二：主要介绍 SQL Server 2022 中数据处理与数据查询方面的基础知识，如利用 T-SQL 语句进行数据查询。

项目三：主要介绍 T-SQL 程序设计的相关知识，包括常量与变量、运算符、流程控制、存储过程、触发器等。

项目四：主要介绍 SQL 与人工智能的相关知识，从 SQL 在人工智能中的作用、利用人工智能增强 SQL 两方面进行具体介绍，并通过预测学生成绩变化趋势，介绍了 SQL Server 2022 中机器学习服务的使用方法。

项目五：主要介绍如何利用 SQL Server 2022 设计收视率调查问卷登记系统。

项目六：主要介绍如何利用 SQL Server 2022 设计学生信息管理系统。

华中科技大学秦磊华教授担任本书主审；集宁师范学院于艳东、姚玉阁，包头师范学院马占飞担任本书主编；内蒙古电子信息职业技术学院林民、内蒙古工业大学马志强、呼和浩特职业学院吴和群担任本书副主编；集宁师范学院陈素清、田箫，北京中晟华安信息技术有限公司兰助晴，北京中软卓越研究院王晓华等参与本书编写。具体编写分工：林民、吴和群负责编写项目一和项目二，马占飞、马志强负责编写项目三和项目四，于艳东、姚玉阁负责编写项目五和项目六。姚玉阁、陈素清、田箫、兰助晴、王晓华等负责全书实践案例部分的设计与编写。于艳东负责全书统稿、审校等工作。

本书内容翔实、示例丰富、结构合理、语言简洁流畅，可作为高等院校计算机及相关专业的数据库开发教材。

由于编者水平有限，书中可能存在不妥之处，敬请广大读者批评指正。

编 者

# CONTENTS

# 目 录

<b>项目一 创建和维护数据库</b>		任务描述 .....	31
		任务分析 .....	32
		准备知识 .....	32
		任务实施 .....	33
		1. 创建主键和外键 .....	33
		2. 删除主键和外键 .....	37
		<b>项目小结</b> .....	39
		<b>项目考核</b> .....	39
<b>项目二 数据处理与数据查询</b>		<b>任务：利用 SQL 语句进行数据查询</b> ..	42
		任务描述 .....	42
		任务分析 .....	43
		准备知识 .....	43
		1. 常用的 SQL Server 语句 .....	43
		2. 更新数据 .....	46
		3. SQL Server 常用函数 .....	52
		4. 创建与管理视图 .....	61
		任务实施 .....	63
		<b>项目小结</b> .....	82
		<b>项目考核</b> .....	82
<b>任务一：创建数据库表</b> .....	2		
任务描述 .....	2		
任务分析 .....	2		
准备知识 .....	2		
1. 数据库相关概念 .....	2		
2. SQL Server 2022 环境的搭建 .....	7		
3. 数据类型 .....	16		
4. Null 值 .....	18		
5. Identity 列 .....	18		
任务实施 .....	19		
1. 创建数据库表 .....	19		
2. 分离和附加数据库 .....	22		
<b>任务二：修改和删除数据库表</b> .....	25		
任务描述 .....	25		
任务分析 .....	25		
准备知识 .....	25		
任务实施 .....	26		
1. 修改表名 .....	26		
2. 设置字段属性 .....	27		
3. 添加表列 .....	28		
4. 删除表列 .....	30		
<b>任务三：创建、删除主键和外键</b> .....	31		

### 项目三 Transact-SQL 程序设计

<b>任务：应用存储过程和触发器</b> .....	85
任务描述 .....	85
任务分析 .....	85
准备知识 .....	85
1. 标识符 .....	85
2. 注释 .....	86
3. 常量与变量 .....	86
4. 运算符 .....	89
5. 运算符的优先级 .....	91
6. 流程控制 .....	91
任务实施 .....	98
1. 创建与管理存储过程 .....	98
2. 创建与管理触发器 .....	103
<b>项目小结</b> .....	116
<b>项目考核</b> .....	116

### 项目四 SQL 与人工智能

<b>任务：预测学生成绩变化趋势</b> .....	119
任务描述 .....	119
任务分析 .....	119
准备知识 .....	119
1. SQL 在 AI 中的作用 .....	119
2. 利用 AI 增强 SQL .....	122
任务实施 .....	122
1. 创建示例数据库和表 .....	122
2. 创建预测存储过程 .....	123
3. 执行预测查询 .....	124
<b>项目小结</b> .....	125
<b>项目考核</b> .....	125

### 项目五 设计收视率调查问卷登记系统

<b>任务：设计并验证收视率调查问卷登记系统</b> .....	127
任务描述 .....	127
任务分析 .....	127
准备知识 .....	127
1. 统计分析的目的和意义 .....	127
2. 系统分析与设计 .....	127
任务实施 .....	128
1. 设计数据库 .....	128
2. 设计系统界面 .....	131
3. 验证系统的运行结果 .....	171
<b>项目小结</b> .....	173
<b>项目考核</b> .....	173

### 项目六 设计学生信息管理系统

<b>任务：设计并验证学生信息管理系统</b> .....	175
任务描述 .....	175
任务分析 .....	175
准备知识 .....	175
1. 系统设计的目的与意义 .....	175
2. 系统分析与设计 .....	176
任务实施 .....	176
1. 设计并实现数据库 .....	176
2. 系统实现 .....	181
3. 验证系统的准确性 .....	216
<b>项目小结</b> .....	225
<b>项目考核</b> .....	225
<b>参考文献</b> .....	226

# 项目一

## 创建和维护数据库

### 知识目标

- 了解数据库、数据库管理系统等数据库相关概念。
- 掌握安装和连接 SQL Server 2022 数据库的方法。
- 掌握 SQL Server 支持的数据类型。
- 掌握创建数据库表的方法。
- 熟悉分离和附加数据库的方法。
- 掌握修改和删除数据库表的方法。
- 掌握创建、删除主键和外键的方法。

### 能力目标

- 能够安装 SQL Server 2022 数据库。
- 能够使用 SQL Server 创建、修改和删除数据库表。
- 能够使用 SQL Server 创建、删除主键和外键。

### 素质目标

- 明确数据库技术的重要地位,激发学生科技报国的家国情怀和使命担当。
- 鼓励学生努力学习专业知识,提升数据处理及管理能力,为构建数字中国做出贡献。

## 任务一:创建数据库表

### 任务描述

为了加强对学生的管理,学生处王老师需要统计在校学生人数及学生的相关信息。

### 任务分析

在 SQL Server 中,用户可以通过创建数据库表来进行数据的统计与整理。因此,王老师决定通过建立数据库表来进行统计。

### 准备知识

数据库是按照数据结构来组织、存储和管理数据的仓库。随着信息技术和市场的发展,特别是 20 世纪 90 年代以后,数据管理不再仅仅是存储和管理数据,而是转变成用户所需要的各种数据管理的方式。数据库有很多种类型,从最简单的存储各种数据的表格到能够进行海量数据存储的大型数据库系统,在各个方面都得到了广泛的应用。

## 1. 数据库相关概念

### (1)数据

数据(data)实际上就是描述事物的符号记录。

计算机中的数据一般分为两部分,其中一部分与程序仅有短时间的交互关系,并随着程序的结束而消亡,称为临时性(transient)数据,这类数据一般存放于计算机内存中;另一部分数据则对系统起着长期持久的作用,称为持久性(persistent)数据。数据库(database, DB)系统中处理的就是这种持久性数据。

软件系统中的数据是有一定结构的。首先,数据有型(type)与值(value)之分,数据的型给出了数据表示的类型,如整型、实型、字符型等;数据的值给出了符合给定型的值,如整型值 1。随着软件系统需求的扩大,数据的型有了进一步的扩大,包括了将多种相关数据以一定结构方式组合构成特定的数据框架,这样的数据框架称为数据结构(data structure);数据库领域中在特定条件下称数据结构为数据模式(data schema)。

过去的软件系统以程序为主体,数据以私有形式从属于程序,因此数据在系统中是分散、凌乱的,这也造成了数据管理的混乱,如出现数据冗余度高、数据一致性差及数据的安全性差等多种弊病。



### 知识链接

近 20 年来,数据在软件系统中的地位发生了变化,在数据库系统及数据库应用程序中数据已占主体地位,而程序已退居附属地位。在数据库系统中需要对数据进行集中、统一的管理,以达到数据被多个应用程序共享的目标。

## (2) 数据库

数据库是数据的集合,它具有统一的结构形式并存放于统一的存储介质内,是多种应用数据的集成,可以被多个应用程序所共享。

数据库是按照数据所提供的数据库模式存放数据的,它能构造复杂的数据结构以建立数据间的内在联系与复杂的关系,从而构成数据的全局结构模式。

数据库中的数据具有“集成”“共享”的特点。也就是说,数据库集中了多种应用的数据,并对其进行统一的构造与存储,从而使它们可以被不同应用程序使用。

## (3) 数据库管理系统

数据库管理系统(database management system, DBMS)是管理数据库的一种系统软件,负责数据库中的数据组织、数据操作、数据维护、数据控制及保护和数据服务等。数据库中的数据是海量级的数据,并且结构复杂,因此需要提供管理工具。数据库管理系统是数据库系统的核心,它主要有以下几方面的具体功能:

①数据模式定义。数据库管理系统负责为数据库构建模式,也就是为数据库构建其数据框架。

②数据存取的物理构建。数据库管理系统负责为数据模式的物理存取及构建提供有效的存取方法与手段。

③数据操作。数据库管理系统为用户使用数据库中的数据提供方便,它一般提供查询、插入、修改及删除数据等功能。此外,它自身还具有做简单算术运算及统计的功能,而且可以与某些过程性语言结合,使其具有强大的过程性操作功能。

④数据的完整性、安全性定义与检查。数据库中的数据具有内在语义上的关联性与一致性,它们构成了数据的完整性。数据的完整性是保证数据库中数据正确的必要条件,因此必须经常检查以维护数据的正确。



### 知识链接

数据库中的数据具有共享性,而数据共享可能会引发数据的非法使用,因此必须对数据正确使用做出必要的规定,并在使用时做检查,这就是数据的安全性。

数据完整性与安全性的维护是数据库管理系统的基本功能。

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06

⑤数据库的并发控制与故障恢复。数据库是一个集成、共享的数据集合体,它能为多个应用程序服务,所以存在多个应用程序对数据库进行同时操作,即并发操作。在并发操作中如果不加以控制和管理,多个应用程序间就会相互干扰,从而对数据库中的数据造成破坏。因此,数据库管理系统必须对多个应用程序的并发操作做必要的控制,以保证数据不受破坏,这就是数据库的并发控制。

数据库中的数据一旦遭受破坏,数据库管理系统必须有能力及时对数据进行恢复,这就是数据库的故障恢复。

⑥数据服务。数据库管理系统提供对数据库中数据的多种服务功能,如数据拷贝、转存、重组、性能监测、分析等。

为完成以上六个功能,数据库管理系统一般提供相应的数据语言(data language),它们是:

①数据定义语言(data definition language, DDL)。该语言负责数据的模式定义与数据的物理存取构建。

②数据操作语言(data manipulation language, DML)。该语言负责数据的操作,包括查询及增、删、改等操作。

③数据控制语言(data control language, DCL)。该语言具有负责数据的完整性、安全性的定义与检查和并发控制、故障恢复等功能,具体包括系统初启程序、文件读写与维护程序、存取路径管理程序、缓冲区管理程序、安全性控制程序、完整性检查程序、并发控制程序、事务管理程序、运行日志管理程序、数据库恢复程序等。

上述数据语言按其使用方式具有以下两种结构形式:

①交互式命令语言。这类语言比较简单,能在终端上即时操作,又称为自含型或自主型语言。

②宿主型语言。这些语言一般可嵌入某些宿主语言(host language)中,如 C、C++ 和 COBOL 等高级语言。

此外,数据库管理系统还有为用户提供服务的服务性(utility)程序,包括数据初始装入程序、数据转存程序、性能监测程序、数据库再组织程序、数据转换程序、通信程序等。

目前流行的数据库管理系统均为关系数据库管理系统(relational database management system, RDBMS),如 Oracle、PowerBuilder、DB2、SQL Server 等,它们均为严格意义上的数据库管理系统。



#### 知识链接

一些小型的数据库管理系统,如微软公司的 Access 等,它们只具备数据库管理系统的一些简单功能。

#### (4) 数据库管理员

由于数据库具有共享性,因此对数据库的规划、设计、维护、监视等需要有专人管理,他们被称为数据库管理员(database administrator, DBA),其主要工作如下:

①数据库设计(database design)。DBA 的主要任务之一是做数据库设计,具体地说是进行数据模式的设计。数据库的集成与共享性,决定了其需要有专门人员(即 DBA)对多个应用的数据需求进行全面的规划、设计与集成。

②数据库维护。DBA 必须对数据库中数据的安全性、完整性、并发控制及系统恢复、数据定期转存等进行实施与维护。

③改善系统性能,提高系统效率。DBA 必须随时监视数据库运行状态,不断调整其内部结构,使系统保持最佳状态与最高效率。



#### 知识链接

当系统效率下降时, DBA 需要采取适当的措施,如进行数据库的重组、重构等。

#### (5) 数据库系统

数据库系统(database system, DBS)由如下几部分组成:数据库(数据)、数据库管理系统(软件)、数据库管理员(人员)、系统平台之一——硬件平台(硬件)、系统平台之二——软件平台(软件)。这五个部分构成了一个以数据库为核心的完整的运行实体,称为数据库系统。

在数据库系统中,硬件平台包括:

①计算机。它是系统中硬件的基础平台,目前常用的有微型机、小型机、中型机、大型机和巨型机。

②网络。过去数据库系统一般建立在单机上,但是近年来它较多地建立在网络上,从目前形势看,数据库系统今后将以建立在网络上为主,而其结构形式又以客户端/服务器(C/S)方式和浏览器/服务器(B/S)方式为主。

在数据库系统中,软件平台包括:

①操作系统。它是系统的基础软件平台,目前常用的主要有 Linux 与 Windows Server 两种。

②数据库系统开发工具。它是指为开发数据库应用程序所提供的工具,包括高级语言,如 C、C++、C#、Python、Java 等,以及与 Internet 有关的 HTML、XML 等和一些专用开发工具。

③接口软件。在网络环境下,数据库系统中数据库与应用程序、数据库与网络间存在着多种接口,它们需要用接口软件进行连接,否则数据库系统整体就无法运作。这些接口软件包括 ODBC、JDBC、OLEDB、CORBA、COM、DCOM 等。



## 思政园地

在数字化时代,数据已成为我国基础性战略资源。数据库作为信息技术的基石,其性能与稳定性直接关系到各行各业的数字化转型进程。

在很长一段时间内,甲骨文、微软、IBM 等国际软件企业占据全球数据库绝大部分市场份额。但随着云计算等技术兴起,国产数据库迎来重大发展机遇。2024 年 10 月 22 日,国际事务处理性能委员会(Transaction Processing Performance Council, TPC)公布,在其开展的数据库“决策支持类”性能测试 TPC-DS 中,排名前四位的产品均来自中国企业。其中,腾讯云数据库 TDSQL 以 7260 万分的性能评分排名第一。

国产数据库的崛起为市场提供了丰富的应用选择,并逐渐获得市场青睐。国际数据公司(International Data Corporation, IDC)发布的数据显示,腾讯在我国关系数据库管理软件的市场份额从 2021 年的 10.31% 上升至 2023 年的 14.90%,华为在相关市场的份额从 2021 年的 10.15% 上升到 2023 年的 11.14%。

## (6) 数据库应用程序

数据库应用程序是由数据库系统加上应用软件、应用界面组成的,具体包括数据库、数据库管理系统、数据库管理员、硬件平台、软件平台、应用软件、应用界面。其中,应用软件由数据库系统所提供的数据库管理系统(软件)及数据库系统开发工具组成,而应用界面大多由相关的可视化工具开发而成。

数据库应用程序的七个部分以一定的逻辑层次结构组成一个有机的整体。如果不计数据库管理员(人员)并将应用软件与应用界面记成用户程序,则数据库应用程序的结构如图 1-1 所示。访问数据库数据的步骤如图 1-2 所示。

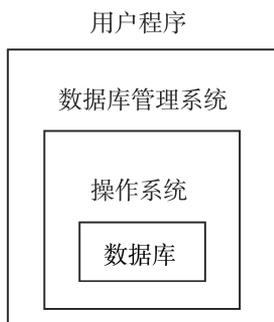


图 1-1 数据库应用程序的结构

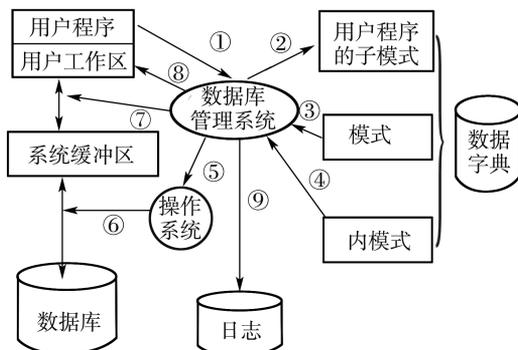


图 1-2 访问数据库数据的步骤

对图 1-2 中各个步骤的简单说明如下:

①用户程序中有一条读取数据库记录的 DML 语句,当计算机执行到该语句时,即向数据库管理系统发出读取相应记录的命令。

②数据库管理系统接到该命令后,首先访问该用户对应的子模式,检查该操作是否在合法授权范围内及欲读记录的正确性、有效性。若不合法,则拒绝执行,并向用户程序状态返回区发出回答状态信息;反之执行下一步。

③数据库管理系统读取模式描述并从子模式映射到全局模式,从而确定所需的逻辑记录类型。

④数据库管理系统从逻辑模式映射到存储模式,从而确定读入哪些物理记录及具体的地址信息。

⑤数据库管理系统向操作系统发出从指定地址读取记录的命令。

⑥操作系统执行读命令,按指定地址从数据库中把记录读入系统缓冲区,并在操作结束后向数据库管理系统做出回答。

⑦数据库管理系统按照模式将读入系统缓冲区中的内容映射成用户要求读取的逻辑记录,并将导出的逻辑记录送入用户工作区。

⑧数据库管理系统将操作执行情况的状态信息返回给用户程序。用户程序根据返回的状态信息决定是否利用该数据进行操作等。

⑨数据库管理系统将已执行的操作载入运行日志。



### 知识链接

如果用户是更新一个记录内容,则执行过程类似。首先读出目标记录,并在用户工作区中进行修改,然后向数据库管理系统发出“写回修改数据”的数据库指令即可。

## 2. SQL Server 2022 环境的搭建

SQL Server 2022 是微软公司于 2022 年 11 月推出的关系数据库管理系统,由一系列服务和工具组件构成,它们分别提供不同的服务功能和管理功能,如表 1-1 所示。

表 1-1 SQL Server 2022 中的基本服务

服务器组件	主要功能说明
SQL Server 数据库引擎	SQL Server 数据库引擎包括引擎、部分工具和“数据库引擎服务(DQS)”服务器。其中,引擎是用于存储、处理和保护数据的核心服务;工具用于管理数据库,支持关系数据与 XML 数据管理,并通过 PolyBase 集成访问 Hadoop 及其他异类数据源
分析服务	分析服务包括一些工具,可用于创建、管理联机分析处理(OLAP)和数据挖掘应用程序
报表服务	报表服务包括用于创建、管理和部署表格报表、矩阵报表、图形报表及自由格式报表的服务器和客户端组件,同时报表服务还是一个可用于开发报表应用程序的可扩展平台
集成服务	集成服务是一组图形工具和可编程对象,用于移动、复制和转换数据,可以与“数据库引擎服务”协同完成数据集成任务

Chapter 01

Chapter 02

Chapter 03

Chapter 04

Chapter 05

Chapter 06

(续表)

服务器组件	主要功能说明
主数据服务	主数据服务(MDS)是针对主数据管理的 SQL Server 解决方案。可以通过配置 MDS 来管理任何数据域(产品、客户、账户);MDS 中可包括层次结构、各种级别的安全性、事务、数据版本控制和业务规则等核心功能,以及 Excel 插件支持
机器学习服务(数据库内)	允许在数据库中执行 Python 和 R 脚本,通过 Python 和 R 脚本准备和清理数据、执行特征工程,以及在数据库中训练、评估和部署机器学习模型
PolyBase	PolyBase 提供数据虚拟化能力,从 SQL Server 查询异构数据源
Azure 连接服务	提供以下扩展功能:Azure SynapseLink、Microsoft Purview 访问策略、SQL Server 的 Azure 扩展、即用即付计费模式,以及 SQL 托管实例的连接功能

### (1) SQL Server 2022 的安装

下面介绍在 Windows 10 操作系统中安装 SQL Server 2022 的步骤。

**STEP 1** 打开 SQL Server 2022 安装文件所在的文件夹,双击其中的 setup.exe 文件,打开“SQL Server 安装中心”窗口,如图 1-3 所示。



图 1-3 “SQL Server 安装中心”窗口

**STEP 2** 在窗口左侧单击“安装”,然后在窗口右侧单击“全新 SQL Server 独立安装或向现有安装添加功能”选项,打开“版本”窗口,如图 1-4 所示。

**STEP 3** 选中“指定可用版本”单选按钮,然后在其下的下拉列表框中选择 SQL Server 2022 版本“Developer”,再单击“下一步”按钮,打开“许可条款”窗口。



图 1-4 “版本”窗口

**STEP 4** 勾选“我接受许可条款和隐私声明”复选框，然后单击“下一步”按钮，打开“全局规则”窗口。该窗口中可以检测在安装 SQL Server 安装程序支持文件时可能发生的问题，检测完成后单击“下一步”按钮，打开“Microsoft 更新”窗口，如图 1-5 所示。

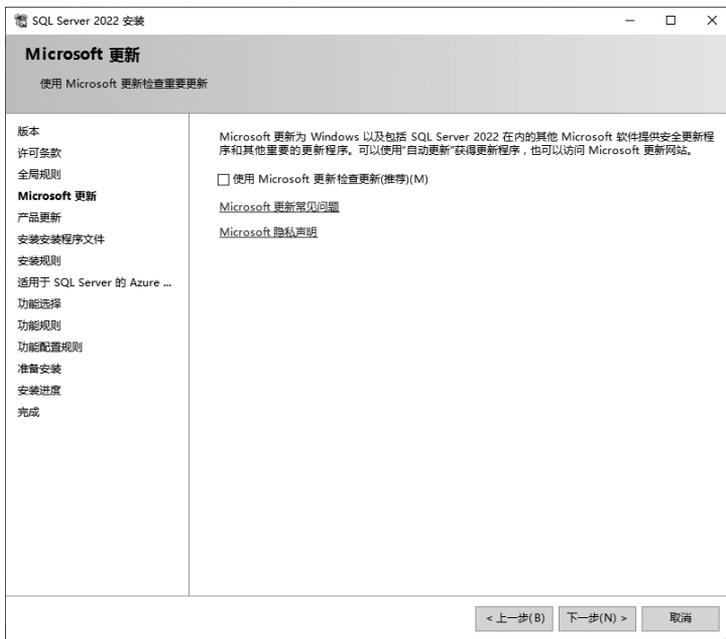


图 1-5 “Microsoft 更新”窗口

**STEP 5** 不勾选“使用 Microsoft 更新检查更新(推荐)”复选框，直接单击“下一步”按钮。

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06



## 知识链接

“产品更新”窗口中显示最近提供的 SQL Server 产品更新,如果未发现任何产品更新,安装程序将不会显示此窗口并自动跳转到下一窗口。

“安装安装程序文件”窗口提供下载、提取和安装安装程序文件的进度。如果找到了针对 SQL Server 安装程序的更新,并且指定包括该更新,则将安装该更新;如果未找到任何更新,将自动跳转到下一窗口。

**STEP 6** 如图 1-6 所示,打开“安装规则”窗口,验证计算机的系统状态;单击“下一步”按钮,打开“适用于 SQL Server 的 Azure 扩展”窗口,如图 1-7 所示。



图 1-6 “安装规则”窗口



图 1-7 “适用于 SQL Server 的 Azure 扩展”窗口

**STEP 7** 取消勾选“适用于 SQL Server 的 Azure”复选框,单击“下一步”按钮,打开“功能选择”窗口,如图 1-8 所示。



图 1-8 “功能选择”窗口

**STEP 8** 选择要安装的功能,不更改窗口底部的目录,单击“下一步”按钮,打开“实例配置”窗口,如图 1-9 所示。

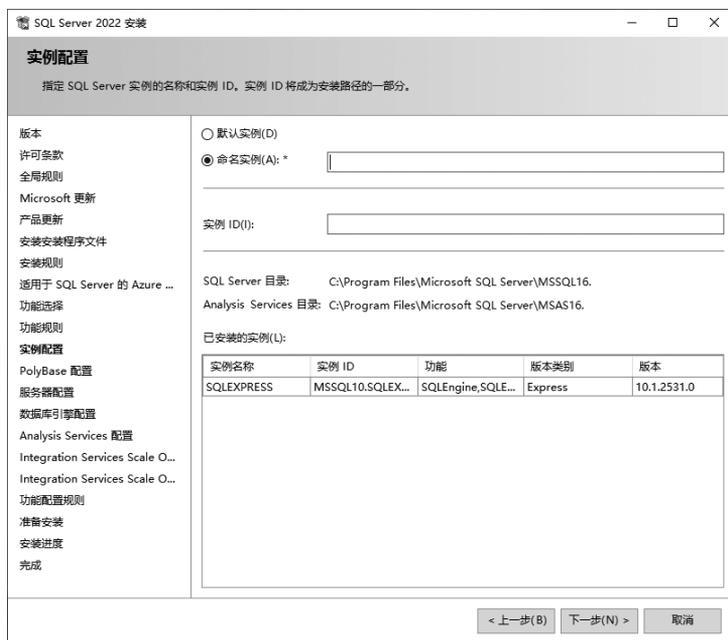


图 1-9 “实例配置”窗口

Chapter  
01

Chapter  
02

Chapter  
03

Chapter  
04

Chapter  
05

Chapter  
06

**STEP 9** 选中“默认实例”单选按钮,单击“下一步”按钮。



### 知识链接

若选中“命名实例”单选按钮,则可以指定实例名称等参数。SQL Server 支持在单个服务器或处理器上安装多个(可以为不同版本、服务、功能和安装目录)SQL Server 实例,每个实例必须有唯一的名称。不论是何版本,服务器都只能安装一个 SQL Server 默认实例。

**STEP 10** 在打开的“PolyBase 配置”窗口中指定 PolyBase 扩大选项和端口范围,保持默认值不变,单击“下一步”按钮,打开“服务器配置”窗口,如图 1-10 所示。

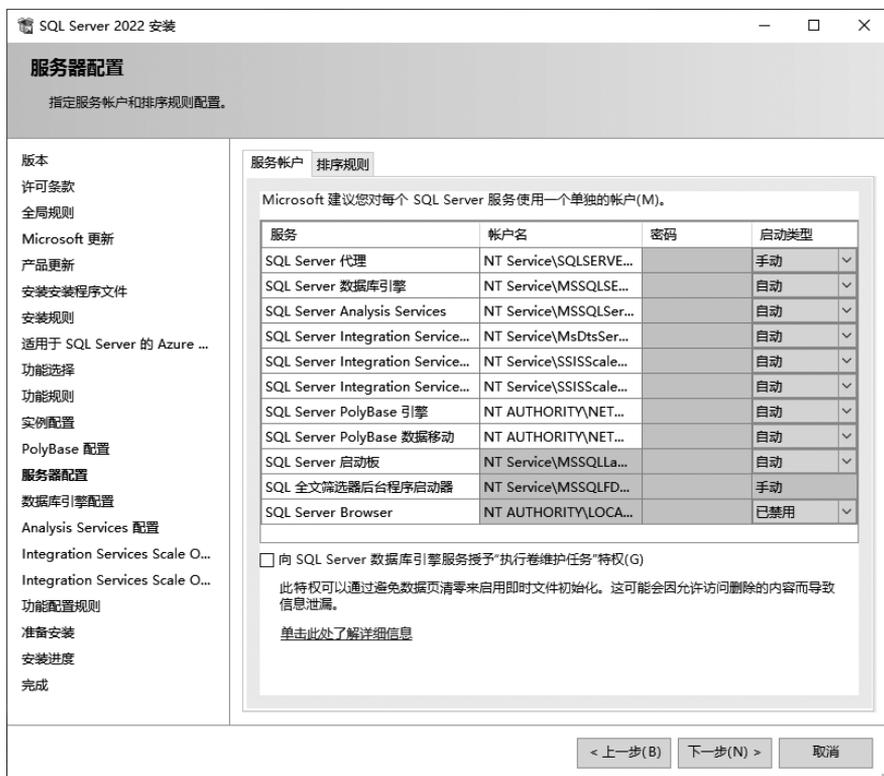


图 1-10 “服务器配置”窗口

**STEP 11** 保持默认值不变,单击“下一步”按钮,打开“数据库引擎配置”窗口,如图 1-11 所示。



图 1-11 “数据库引擎配置”窗口

**STEP 12** 选中“混合模式(SQL Server 身份验证和 Windows 身份验证)”单选按钮,设置 sa 账户的密码,并单击“添加当前用户”按钮,然后单击“下一步”按钮。



### 知识链接

混合模式身份验证是 SQL Server 中常用的一种身份验证模式,它结合了 Windows 身份验证和 SQL Server 身份验证两种方式。这意味着用户既可以使用 Windows 操作系统中的用户账户进行身份验证(直接通过信任连接访问 SQL Server,无须重复输入凭据),也可以使用 SQL Server 中独立的用户账户和密码进行身份验证。

**STEP 13** 在打开的“Analysis Services 配置”窗口中根据提示进行设置,完成后单击“下一步”按钮,打开“Integration Services Scale Out 配置-主节点”窗口。

**STEP 14** 保持默认值不变,单击“下一步”按钮,打开“Integration Services Scale Out 配置-辅助角色节点”窗口。

**STEP 15** 保持默认值不变,单击“下一步”按钮,打开“准备安装”窗口,如图 1-12 所示。单击“安装”按钮,根据提示完成安装。

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06



图 1-12 “准备安装”窗口

## (2) SQL Server Management Studio 的安装

安装好 SQL Server 2022 服务器后,要使用可视化工具管理 SQL Server 2022,可以安装 SQL Server Management Studio(SSMS)管理工具。其安装步骤如下。

**STEP 1** 在图 1-3 中单击“安装 SQL Server 管理工具”选项,在打开的官方页面中下载 SSMS 安装文件。

**STEP 2** 双击下载的安装文件“SSMS-Setup-CHS.exe”,打开 SSMS 安装向导,如图 1-13 所示。设置 SSMS 的安装路径,单击“安装”按钮,根据提示完成安装。



图 1-13 SSMS 的安装向导

### (3)使用 SSMS 连接 SQL Server 服务器

安装好 SQL Server 2022 和 SSMS 后,启动 SSMS 并连接到 SQL Server 服务器,即可在其中配置数据库系统、建立或删除数据库对象、设置或取消用户的访问权限等。使用 SSMS 连接 SQL Server 服务器的步骤如下。

**STEP 1** 选择“开始”|Microsoft SQL Server Tools 20|SQL Server Management Studio 20,打开“连接到服务器”对话框,如图 1-14 所示。



图 1-14 “连接到服务器”对话框

**STEP 2** 在“连接到服务器”对话框中选择服务器名称(通常为默认)和身份验证方式。如果选择“Windows 身份验证”,可以直接单击“连接”按钮;如果选择“SQL Server 身份验证”,则需要输入安装 SQL Server 2022 数据库时设置的登录名和密码,其中登录名通常为 sa,密码由用户设置。单击“连接”按钮后,即可进入 SSMS 主界面,并连接到 SQL Server 服务器,如图 1-15 所示。

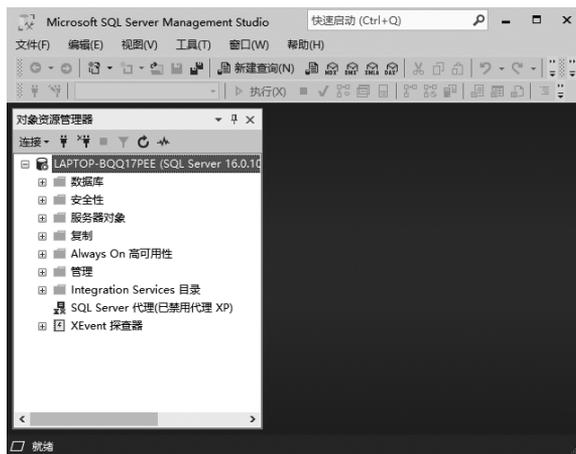


图 1-15 SSMS 主界面

Chapter  
01

Chapter  
02

Chapter  
03

Chapter  
04

Chapter  
05

Chapter  
06

### 3. 数据类型

在 SQL Server 2022 中,每个列、局部变量、表达式和参数都具有一个相关的数据类型。数据类型是一种属性,用于指定对象可保存的数据的类型,如整数数据、字符数据、货币数据、日期和时间数据、二进制字符串等。数据库表按行和列存储数据,在创建表时将涉及定义数据类型。数据类型决定了每个列存储数据的范围。为一个列选择数据类型时,应选择允许你期望存储的所有数据值的数据类型,同时使所需的空间量最小。使用一个长度恰当、合适的数据类型有利于数据校验以及更好地利用存储空间,并提升性能。因此,在决定要使用哪种数据类型时,可以参照以下原则。

①为列选择一个合适的长度。

②如果属性值的长度不会大幅改变,就使用固定长度数据类型(char 和 nchar),如存储身份证号码、邮政编码等。如果属性值的长度会大幅变化,就使用变长数据类型(varchar 和 nvarchar),如存储学生简历等。

SQL Server 2022 的数据类型很丰富,如表 1-2 所示。

表 1-2 SQL Server 2022 的数据类型

数据类型分类	数据类型	取值范围及应用说明	
精确数字	bit	1(真)或 0(假),字符串“True”转换为 1,任何非 0 值转换为 1,字符串“False”转换为 0,用于逻辑数据	
	tinyint	很小的整数,0~255	
	整数	smallint	短整数, $-32\ 768(-2^{15})\sim 32\ 767(2^{15}-1)$
		int	整数, $-2\ 147\ 483\ 648(-2^{31})\sim 2\ 147\ 483\ 647(2^{31}-1)$
		bigint	长整数, $-9\ 223\ 372\ 036\ 854\ 775\ 808(-2^{63})\sim 9\ 223\ 372\ 036\ 854\ 775\ 807(2^{63}-1)$
	小数	decimal[(p[,s])]	定点,有效值范围为 $-10^{38}+1\sim 10^{38}-1$ ;精度 p(1~38)定义数字总位数,默认为 18,包括小数点左边和右边的位数;s ( $0\leq s\leq p$ )定义小数位数,默认为 0
		numeric[(p[,s])]	等价于 decimal
	货币	money	$-922\ 337\ 203\ 685\ 477.580\ 8\sim 922\ 337\ 203\ 685\ 477.580\ 7$ (精确到 4 位小数)
		smallmoney	$-214\ 748.364\ 8\sim 214\ 748.364\ 7$ (精确到 4 位小数)
	近似数字	小数	real
float[(n)]			浮点, $-1.79E+308\sim -2.23E-308, 0, 2.23E-308\sim 1.79E+308$ ;若 n 为 1~24,n 视为 24,等价于 real 类型;若 n 为 25~53,n 视为 53,15 位有效数字

(续表)

数据类型分类	数据类型	取值范围及应用说明
日期时间	datetime2	0001-01-01 00:00:00.0000000 ~ 9999-12-31 23:59:59.9999999,精确到 100ns
	datetimeoffset	与可识别时区(基于协调世界时(coordinated universal time, UTC))的一日内时间相组合的日期。YYYY-MM-DD hh:mm:ss[.nnnnnnn][{+ -}hh:mm],其中{+ -}hh:mm为时区偏移量,如北京时间 2023-10-01 13:30:00+08:00,+08:00表示比 UTC 早 8 小时
	date	0001-01-01~9999-12-31(仅存日期)
	time	00:00:00.0000000~23:59:59.9999999(仅存时间),精确到 100ns
字符串 (单字节)	char[(n)]	定长,n(1~8 000)定义字符串长度,默认为 1,实际字符串短 时用空格补足
	varchar[(n max)]	变长,n(1~8 000)定义字符串长度,默认为 1;使用 varchar (max)表示其长度可足够大(达 $2^{31}-1$ 字节,2 GB),替代原 来的 text
Unicode 字符串 (双字节)	nchar[(n)]	定长,n(1~4 000)定义字符串长度,默认为 1,实际字符串短 时用空格补足,用于非英语语言,如中文
	nvarchar[(n max)]	变长,n(1~4 000)定义字符串长度,默认为 1;使用 nvarchar (max)表示其长度可足够大(达 $2^{31}-1$ 字节,2 GB),替代原 来的 ntext
二进制 字符串	binary[(n)]	定长,n(1~8 000)字节,用于图像、视频、音频等数据
	varbinary[(n max)]	变长,n(1~8 000)字节,varbinary(max)表示其作为 LOB(大 对象),最大存储大小为 $2^{31}-1$ 字节,替代原来的 image,用 途同上
树形结构	hierarchyid	用于表示节点在树形层次结构中的位置,该类型的列不会自 动表示树,而是由应用程序生成和分配其值,使行与行之间的 所需关系反映在这些值中
唯一标识	rowversion	自动生成的唯一的二进制数,当对包含 rowversion 列的表执 行插入数据行或更新数据操作时,该列值递增;每个表只能 有一个 rowversion 列
	uniqueidentifier	全球唯一标识(GUID),十六进制数字,由网卡、处理器 ID 及 日期和时间等字符串常量产生,或者由 NEWID()函数产生, 用法同上
空间数据	geometry	欧几里得(平面地球)坐标系中点、直线、曲线、多边形数据, 用于确定地理位置
	geography	为空间数据提供了一个由经度和纬度联合定义的存储结构, 诸如 GPS 纬度和经度坐标之类的椭球体(圆形地球)数据, 用于确定地理位置
扩展标 记语言	xml	XML 数据(文档或片段),可以在 xml 类型的列或者变量中 存储 xml 实例,实例大小不能超过 2 GB

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06

(续表)

数据类型分类	数据类型	取值范围及应用说明
程序中的数据类型	cursor	变量或存储过程 OUTPUT 参数的一种数据类型,这些参数包含对游标的引用
	table	用于存储表操作的结果集,以便进行后续处理,通常作为用户定义函数返回值的数据类型
	sql_variant	各种数据类型的值,其列可能包含不同数据类型的行
用户自定义	用户自行命名	用户可创建自定义的数据类型

## 4. Null 值

Null 值是一个未知值,将该值引用为 Null。列值的“为空性”是指该列接受或拒绝 Null 值的能力。一列中的 Null 值通常表明对于一个特殊的数据行,该列中没有输入项,因为该值既不为空也不为 0,其真实的值是未知的。



### 知识链接

如果用户需要的信息仍然未知,则可能需要设为空列。例如,客户信息的初始化,有些信息在初始化时可能未知,需要以后补充,这些需要以后补充的信息字段可以设为 Null 值。

一般来说,应该避免使用 Null 值,因为对这些值的更新或查询,以及对有这些值的列的一些选项的设置都会更加复杂。例如,主键和 Identity 属性都不能在为空的列中使用。

如果定义了允许 Null 值的列,则可以通过以下方法将 Null 值输入到该列中。

①如果将行插入到该表,但没有为这个为空列指定数据值,则 SQL Server 将为该列指定为 Null 值。

②用户可以输入 Null 这个词(Null 不加引号是为了和加引号的字符串“Null”区别开)。

## 5. Identity 列

在创建表时,可以通过向列定义中添加 Identity 属性来将某一列指定为标识列。创建带有 Identity 属性的列时,SQL Server 将根据种子值(seed)和增量值(increment)自动为该列生成一个行值。默认情况下,不允许手动为 Identity 列插入行值,除非将标识列的 IDENTITY\_INSERT 属性设置为 ON。种子值是插入到表中第一行的标识值。增量值是 SQL Server 为了连续插入行而递增标识值的量。当每次插入行时,SQL Server 都将为该行的标识列指定一个当前的标识值。被插入的下一行收到这个标识值,它是一个比当前最大标识值更大的增量。这样,每个被插入的行都将收到一个唯一的标识符。为了唯一地标识某一行,通常都将标识列用作表中的主键

约束。

例如,为标识列指定了 Identity(1,1),其中第一个参数是种子值,第二个参数是增量值,那么被插入的第一行将得到值为 1 的标识列,第 2 行得到的值为 2,而第三行得到的值为 3,依次类推。

## 任务实施

### 1. 创建数据库表

可以使用 SSMS 工具创建数据库表,也可以通过编码的方式使用 T-SQL 创建数据库表。具体操作步骤如下。

**STEP 1** 打开 SSMS 并连接到数据库服务器,在“对象资源管理器”窗口中展开“数据库”,接着展开“数据库”|test 节点,在下拉菜单中选择“表”并右击,如图 1-16 所示。

**STEP 2** 在弹出的快捷菜单中选择“新建”|“表”命令,将出现如图 1-17 所示的界面。

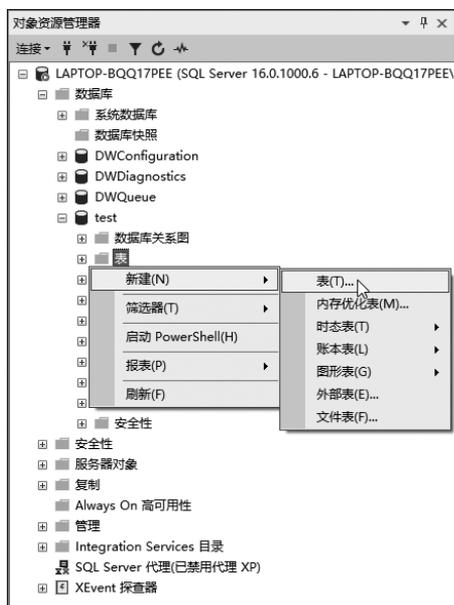


图 1-16 创建表



图 1-17 添加表列

**STEP 3** 在如图 1-17 所示的界面中添加构成表的字段,为每个字段指定数据类型并设置是否允许为空。在下半部分的“列属性”中可以为字段设置更多的属性,如名称、长度、数据类型及标识规范等。添加完表所需的所有字段后,单击“保存”按钮,将打开如图 1-18 所示的保存表对话框,在“输入表名称”文本框中输入表的名称后单击“确定”按钮即可。

**STEP 4** 在创建表的时候,可以为表指定主键。方法非常简单,在如图 1-17 所示的界面中,选择要设置为主键的字段,然后单击工具栏上的“设置主键”按钮,如图 1-19 所示。



图 1-18 保存表



图 1-19 设置主键

**STEP 5** 在如图 1-17 所示的界面中,在“列名”列中输入 AreaID,“数据类型”列中选择 int 数据类型,去掉“允许 Null 值”列中的“√”,在“列属性”中展开“标识规范”,设置“(是标识)”属性值为“是”,“标识增量”属性为 1(默认值为 1),“标识种子”属性为 1(默认值为 1)。同样地,增加列 Area,分别指定列名“Area”,数据类型 varchar(20),允许 Null 值。将字段 AreaID 设置为主键,完成后保存表名为 Area\_Tbl。表 Area\_Tbl 的结构如图 1-20 所示。

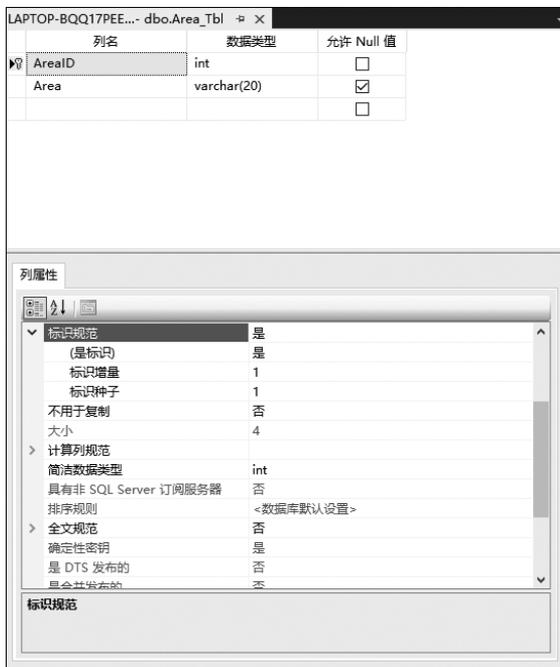


图 1-20 表 Area\_Tbl

**STEP 6** 可以采用编码的方式,使用 T-SQL 来创建表。创建表的 T-SQL 语法如下。

```
CREATE TABLE
    [ database_name. [ owner ] . | owner. ] table_name
    ( { < column_definition >
      | column_name AS computed_column_expression
```

```

        | < table_constraint > ::= [ CONSTRAINT constraint_name ] }
        | [ { PRIMARY KEY | UNIQUE } [ ,...n ]
    )
    [ ON { filegroup | DEFAULT } ]
< column_definition > ::= { column_name data_type }
    [ COLLATE < collation_name > ]
    [ [ DEFAULT constant_expression ]
      | [ IDENTITY ( seed , increment ) [ NOT FOR REPLICATION ] ] ]
    ]
    [ ROWGUIDCOL ]
    [ < column_constraint > ] [ ...n ]

```

相关参数说明如下：

- CREATE TABLE: 主关键字。table\_name: 表示要创建的表的名称。
- column\_definition: 表示列的定义, 包括名称、数据类型、长度、是否为空等。

以创建表 Area\_Tbl 为例, 看看用 T-SQL 创建表的代码。创建表 Area\_Tbl 的代码如下。

```

USE [DataCollationSys]
GO

CREATE TABLE [dbo].[Area_Tbl](
    [AreaID] [int] IDENTITY(1,1) NOT NULL,
    [Area] [varchar](20) COLLATE Chinese_PRC_CI_AS NULL,
    CONSTRAINT [PK_AREA_TBL] PRIMARY KEY CLUSTERED
    (
        [AreaID] ASC
    )WITH (IGNORE_DUP_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

```

相关参数说明如下：

- USE [DataCollationSys]: 指定了要在其中创建表的库。
- [dbo]: 指定表的拥有者。[Area\_Tbl]是表名。AreaID 是标识列字段。
- Area: 该字段的字符集是 Chinese\_PRC\_CI\_AS。
- WITH (IGNORE\_DUP\_KEY = OFF): 说明表主键不会忽略重复值。

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06



**STEP 2** 弹出“分离数据库”对话框,如图 1-22 所示,列出了要分离的数据库 fenlitest,单击“确定”按钮,指定数据库将被分离。

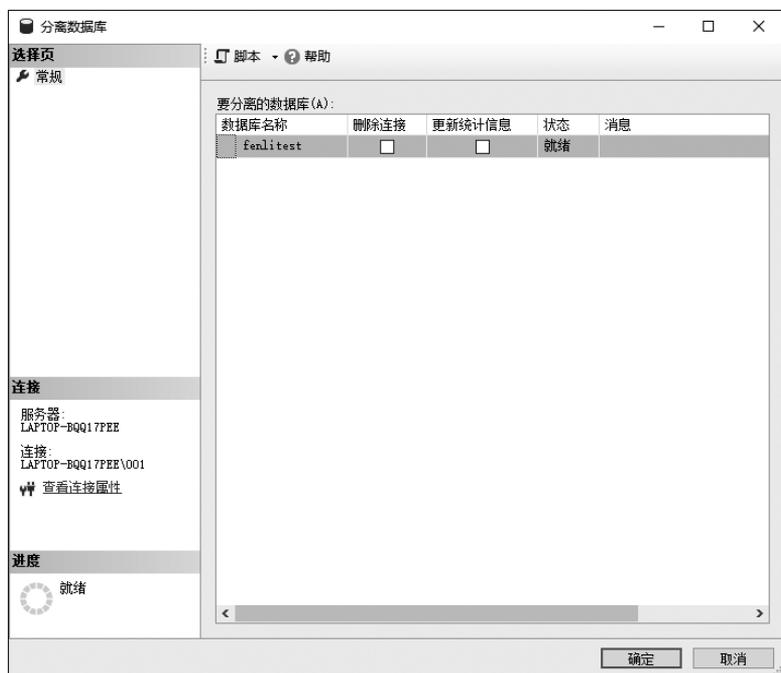


图 1-22 “分离数据库”对话框

“分离数据库”对话框中相关选项的说明如下:

- 删除连接:如果还没有删除所有的用户连接,则必须勾选这个复选框来删除用户连接。
- 更新统计信息:默认情况下,分离操作将在分离数据库时保留过期的优化统计信息;如果需要更新现有的优化统计信息,则勾选这个复选框。
- 状态:显示当前数据库的状态(“就绪”或“未就绪”)。
- 消息:数据库有活动连接时,“消息”列将显示活动连接的个数。

2)使用 T-SQL 命令分离数据库。

①sp\_detach\_db 用于分离数据库,其语法格式如下:

```
sp_detach_db 数据库名
```

②使用 sp\_detach\_db 分离数据库 fenlitest 的 T-SQL 命令如下:

```
sp_detach_db fenlitest
```

## (2)附加数据库

在 SQL Server 中,用户可以对 SQL Server 实例附加被分离的数据库。附加时,数据库引擎会启动数据库,并将数据库重置为分离或复制时的状态。附加数据库 fenlitest 也有以下两种方法。

1)使用 SSMS 附加数据库。

**STEP 1** 在“对象资源管理器”窗口中,右击“数据库”节点,在弹出的快捷菜单中选择“附加”命令,如图 1-23 所示。

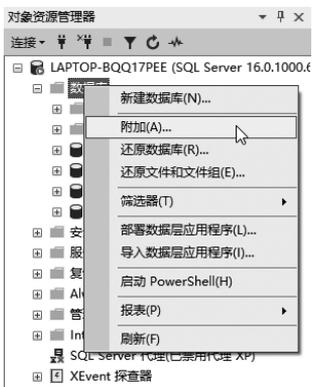


图 1-23 选择“附加”命令

**STEP 2** 在“附加数据库”对话框中,单击“添加”按钮,添加准备附加的数据库,在弹出的“定位数据库文件”对话框中选择该数据库数据文件 fenlitest.mdf 所在的路径“C:\database\fenlitest.mdf”、日志文件 fenlitest\_log.ldf 所在的路径“C:\database\fenlitest\_log.ldf”。

**STEP 3** 选择数据库文件后,单击“确定”按钮,返回“附加数据库”对话框,如图 1-24 所示。

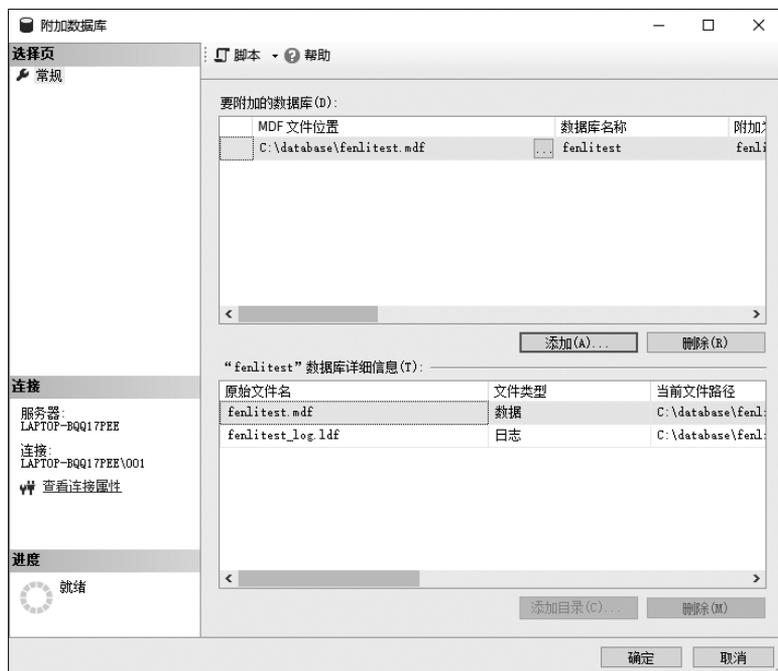


图 1-24 “附加数据库”对话框

**STEP 4** 准备好附加数据库后,单击“确定”按钮,数据库 fenlitest 就被附加到 SQL Server 实例上了。

注意:如果附加不成功,可以在操作系统下查看要附加数据库文件的权限,在文件属性对话框的“安全”选项卡中单击“编辑”按钮,将 Authenticated Users 组的权限设置为“完全控制”。

2)使用 T-SQL 命令附加数据库。

①sp\_attach\_db 用于附加数据库,其语法格式如下:

```
sp_attach_db 数据库名,数据库文件列表
```

相关参数说明如下:

- 数据库文件列表至少应包括主数据文件。主数据文件包括指向数据库中其他文件的系统表。
- 数据库文件列表还必须包括数据库分离后所有被移动的文件。

②使用 sp\_attach\_db 附加数据库 fenlitest 的 T-SQL 命令如下:

```
sp_attach_db 'fenlitest','C:\database\fenlitest.mdf','C:\database\fenlitest_log.ldf'
```

## 任务二:修改和删除数据库表

### 任务描述

小李是某酒店的大堂经理,为了能够统一规划酒店所承办的各种宴会及宴会的类型、时间、准备工作等事项,小李需要确认酒店的数据库表。

### 任务分析

在创建数据库表以后,有可能根据业务系统和存储数据内容的变化,对表做相应的修改。因此,小李决定根据实际情况对数据库表做相应的修改或删除。

### 准备知识

修改数据库表涉及的内容包括修改表名、字段名,添加或删除字段,以及修改表字段的属性,等等。

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06

## 任务实施

### 1. 修改表名

1) 通过 SSMS 修改表名。

打开 SSMS 并连接到数据库服务器,在“对象资源管理器”中展开“数据库”节点,接着展开数据库“test”节点,展开“表”文件夹,右击要更改名字的表,选择快捷菜单中的“重命名”命令,如图 1-25 所示。然后输入新的表名,这样就完成了表名的修改。

2) 使用 T-SQL 更改表名。

可以使用系统存储过程 `sp_rename` 修改表名。`sp_rename`的语法格式如下:

```
sp_rename [ @objname = ] 'object_name',
          [ @newname = ] 'new_name'
          [ , [ @objtype = ] 'object_type' ]
```

相关参数说明如下:

- @objname: 要修改的对象名字。
- @newname: 新的数据库对象名。
- @objtype: 要修改的数据库对象的类型,这里是“object”。

将以 `Area_Tbl` 表名修改为 `Tbl_Area` 为例,说明 `sp_rename` 的用法,如下所示:

```
USE DataCollationSys
Go
exec sp_rename 'Area_Tbl','Tbl_Area','Object'
```

这样可以将表 `Area_Tbl` 更名为 `Tbl_Area`。因为更改表名,所以类型为“Object”。



### 拓展提高

修改表名会引起使用该表的视图、存储过程或函数产生“找不到对象”的错误,因此在更改表名前,必须确定是否有视图、存储过程或函数引用了该表。如果有,而且一定要更改该表名,要记得更改引用该表的视图、存储过程或函数中该表表名为对应的新表的表名。

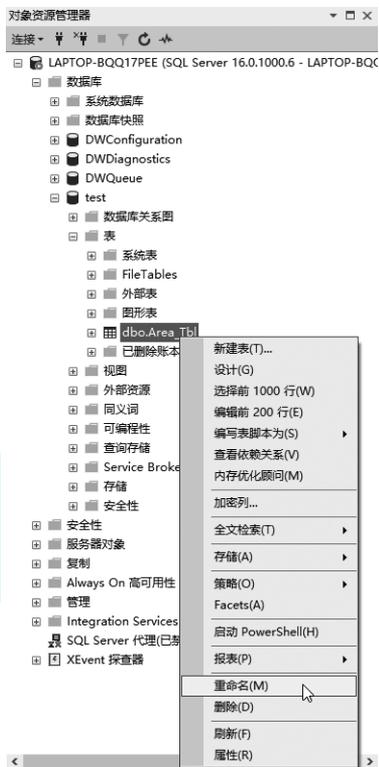


图 1-25 修改表名

## 2. 设置字段属性

1) 在 SSMS 中修改字段名及其相关属性。

打开 SSMS 并连接到数据库服务器,在“对象资源管理器”中展开“数据库”节点,接着展开“数据库”|test 节点,展开“表”文件夹,右击要更改名字的表 Area\_Tbl,在快捷菜单中选择“设计”命令,如图 1-26 所示。

在如图 1-27 所示的界面中可以修改字段名、数据类型及字段的其他相关属性。



图 1-26 展开表字段



图 1-27 修改表字段属性

在图 1-27 中,可以修改字段 Area 的名字为 AreaName,并将字段数据类型改为 char,长度改为 15,并且不允许该字段为 Null。在“列名”列中单击 Area 字段,然后输入 AreaName,将 Area 改为 AreaName,在“数据类型”列中选择 char 并输入 15,将数据类型改为 char,长度改为 15,在“允许 Null 值”列中去掉“√”,将该字段设置为不允许为空,然后单击工具栏上的“保存”按钮保存修改,这样就修改了表字段及其相关的属性。

2) 使用 T-SQL 来修改表字段名及其相关的属性。

使用存储过程 sp\_rename 修改字段名。sp\_rename 的语法在上面修改表名中已介绍,这里不再介绍,直接看使用 sp\_rename 存储过程修改字段 Area 为 AreaName 的 T-SQL 语句,如下所示:

```
USE DataCollationSys
GO
EXEC sp_rename 'Area_Tbl.Area','AreaName','COLUMN'
```

因为是修改表字段,所以必须在第一个参数中指明要修改的字段属于哪个表,在第三个参数中指定修改的对象类型为 COLUMN。



## 拓展提高

同修改表名一样,修改表字段属性也会影响使用该字段的视图、存储过程或函数。因此在修改了字段名后,别忘了将引用该字段的视图、存储过程或函数中相应的字段名改成新的字段名。

使用 T-SQL 修改字段数据类型、长度及其他属性的 T-SQL 语法如下:

```
ALTER TABLE table
{ [ ALTER COLUMN column_name
  { new_data_type [ ( precision [ , scale ] ) ]
    [ COLLATE < collation_name > ]
    [ NULL | NOT NULL ]
    | {ADD | DROP } ROWGUIDCOL }
  ] }
```

相关参数说明如下:

- ALTER TABLE 和 ALTER COLUMN:主关键字。
- table:修改字段的表的名字。column\_name:要修改的表字段名。
- new\_data\_type:指定新的数据类型以及相关的长度。
- COLLATE:指定字段使用的字符集。NULL|NOT NULL:指明字段是否允许为空。

例如,修改 Area\_Tbl 字段 Area 的 T-SQL 语句如下:

```
USE DataCollationSys
Go
ALTER TABLE Area_Tbl
ALTER COLUMN Area char(15) NOT NULL
```

注意:不是所有的列都可以改变。通常,下列类型的列不能修改:

- 属于主键或外键约束的列。
- 用于复制的列。
- 具有 text、ntext、image 或 timestamp 数据类型的列。
- 在索引中使用的列。
- 用于检查或约束的列。
- 用于计算的列。
- 通过明确地执行 CREATE STATISTICS 语句创建统计的列。

## 3. 添加表列

1)在 SSMS 中添加列。

可以在如图 1-17 所示的添加表列界面中添加表列,直接在“列名”列中输入要新添加的列名,在“数据类型”列中为新添加的列选择数据类型,并指定数据长度,在“允许 Null 值”列中设定该列是否允许为空。用户也可以在 SSMS 中连接数据库服务器后,依次展开“数据库”节点、要添加列的表所属的数据库(这里是 test 数据库)、“表”文件夹、要修改的表(这里是 Area\_Tbl 表)、“列”文件夹,右击“列”文件夹,在弹出的快捷菜单中选择“新建列”命令,如图 1-28 所示。接着会出现如图 1-17 所示的添加表列界面,在该界面中进行添加表列。

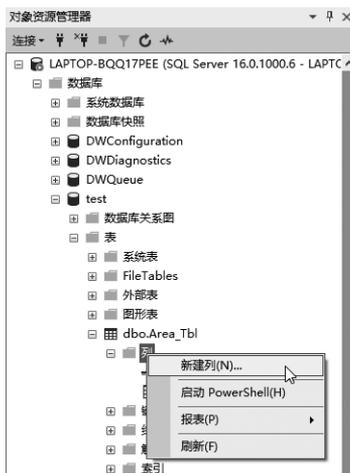


图 1-28 新建列

2) 采用编码的方式来为表添加新列。

添加表列的 T-SQL 语法如下:

```
ALTER TABLE table_name
ADD column_name datatype NULL|NOT NULL
DEFAULT default_value
```

相关参数说明如下:

- ALTER TABLE、ADD 和 DEFAULT: 主关键字。
- table\_name: 指定要添加表列的表。column\_name: 新添加的表列的名字。
- datatype: 新添加的表列的数据类型。NULL|NOT NULL: 指明字段是否允许为空。
- default\_value: 为该字段指定默认值,当向该表插入数据时,如果没有为该字段指定要插入的数据,将向该字段插入默认值,默认值项是可选的。

下面的 SQL 语句实现向表 Area\_Tbl 中添加字段“memo”:

```
USE DataCollationSys
GO
ALTER TABLE Area_Tbl
ADD memo varchar(200) NULL
```

## 4. 删除表列

1) 在 SSMS 中删除表列。

打开 SSMS, 连接到数据库服务器, 依次展开“数据库”节点、要删除列的表所属的数据库(这里是 test 数据库)、“表”文件夹、要修改的表(这里是 Area\_Tbl 表)、“列”文件夹, 右击要删除的字段, 在弹出的快捷菜单中选择“删除”命令, 如图 1-29 所示。在接着出现的界面中单击“确认”按钮, 删除表列。也可以在如图 1-30 所示的界面中, 选中要删除的列, 右击该列, 在弹出的快捷菜单中选择“删除列”命令, 删除表列。

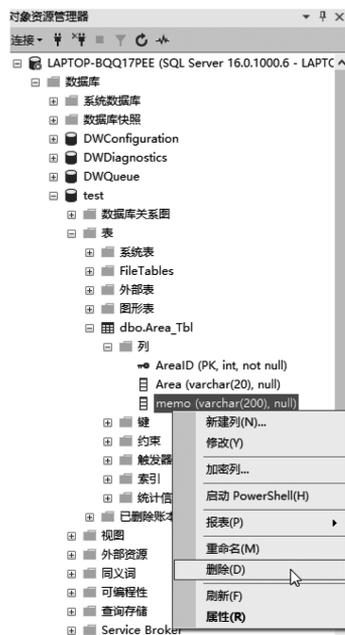


图 1-29 删除表列

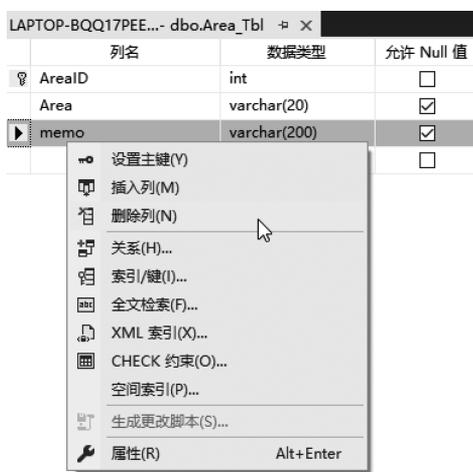


图 1-30 在表设计器中删除列

2) 采用编码的方式编写 T-SQL 脚本删除表列。

删除表列的 T-SQL 语法如下:

```
ALTER TABLE table_name
DROP COLUMN column_name
```

相关参数说明如下:

- ALTER TABLE 和 DROP COLUMN: 主关键字。
- table\_name: 要删除字段的表。column\_name: 要删除的字段名。

下面的 T-SQL 语句用于删除表 Area\_Tbl 中的字段 memo(先前新添加的列):

```
USE DataCollationSys
GO
ALTER TABLE Area_Tbl
DROP COLUMN memo
```

注意:删除列后,列中的数据内容也会一起删除。通常,下列类型的字段不能删除:

- 用于主键或外键的字段。
- 用于复制的字段。
- 用作索引中的列(除非先删除索引)。
- 符合规则的列。
- 与默认值关联的列。

删除数据库表非常简单,打开 SSMS,连接数据库服务器,依次展开“数据库”节点、表所在的数据库、“表”文件夹,右击要删除的表,在弹出的快捷菜单中选择“删除”命令,接着单击“确定”按钮,完成表的删除,如图 1-31 所示。

用户也可以在查询分析器中编写 T-SQL 脚本删除表。删除表的 T-SQL 语法如下:

```
DROP TABLE table_name
```

相关参数说明如下:

- DROP TABLE:表示要删除表。
- table\_name:要删除的表的名称。

下面 T-SQL 语句用于删除 City\_Tbl 表:

```
DROP TABLE City_Tbl
```



图 1-31 删除表

Chapter  
01

Chapter  
02

Chapter  
03

Chapter  
04

Chapter  
05

Chapter  
06

## 任务三:创建、删除主键和外键

### 任务描述

张老师是某学校高二年级的年级主任,期中考试后,张老师想创建数据库表来统

计学生的考试成绩,要求能够根据学生成绩的数据库表对全校学生进行排名。

## 任务分析

在 SQL Server 的关系数据库中,表与表间通过关系进行关联,最重要的关系就是主键和外键。因此,张老师决定通过创建或删除主键和外键来完成这一任务。

## 准备知识

主键是表中唯一标识一行的约束。主键通常定义在一列中,有时也定义在几列中,通过几列组合在一起唯一标识一行。通过主键可强制表的实体完整性,即确保数据库中所代表的任何事物都不存在重复的数据,每一个表只能有一个主键约束,而且主键不接受空值。

## 拓展提高

由于主键约束确保唯一数据,因此经常用来定义标识列。所谓标识列,指表中已指派了标识属性的列。标识属性生成唯一数字。

例如,表 Area\_Tbl 中 AreaID 是主键,因此字段 AreaID 唯一标识一行,如表 1-3 所示。

表 1-3 Area\_Tbl 样例数据

AreaID	Area
2	华东区
3	华中区
4	华北区
5	西南区
7	华南区

外键是用于建立和加强两个表之间数据关联的一列或多列。外键一般建立在一列上,有时也通过几列组合起来建立外键。通过将一个表中的主键值的一列或多列添加到另一个表,可以创建两个表之间的链接,这个列就称为第二个表的外键。

外键约束并不仅仅只可以与另一个表的主键约束相链接,它还可以定义为引用另一个表的唯一约束。外键约束不允许为空值,但是如果任何组合外键的列包含空值,则跳过外键约束的校验。

图 1-32 所示的表 Area\_Tbl 和 Province\_Tbl,表 Province\_Tbl 中 AreaID 是表 Area\_Tbl 的主键,把 AreaID 字段加入表 Province\_Tbl 中,那么在表 Province\_Tbl 中 AreaID 字段就是外键,表 Area\_Tbl 和 Province\_Tbl 通过 AreaID 字段关联。

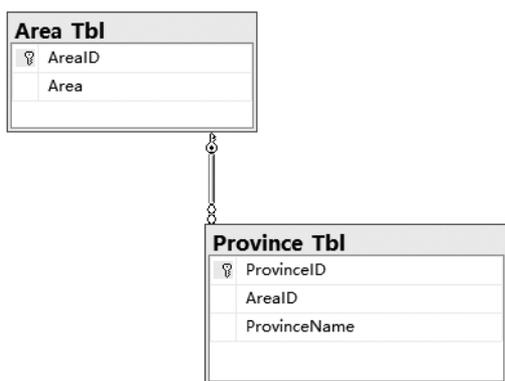


图 1-32 主键和外键关联示例



## 拓展提高

尽管外键的主要目的是控制存储在外键表中的数据,但它还可以控制对主表数据的修改。

例如,图 1-32 中,如果删除了表 Area\_Tbl 中的数据,那就破坏了表 Area\_Tbl 与 Province\_Tbl 关联的完整性,如果删除了表 Area\_Tbl 中的一条记录数据,那么表 Province\_Tbl 中与该条记录关联的数据就因为没有关联而变得孤立。外键约束就是防止这种情况的发生,如果有外键约束,要删除表 Area\_Tbl 中某条与表 Province\_Tbl 关联的数据,那么删除时将发生错误。



## 任务实施

### 1. 创建主键和外键

像创建其他数据库对象一样,既可以在 SSMS 中创建主键和外键,也可以通过编码的方式,用 T-SQL 在查询分析器中创建主键和外键。此外,既可以在创建表的同时创建主键和外键,也可以在创建表后,将表的某个字段或某几个字段设置为主键和外键。

1) 在 SSMS 中创建主键和外键。

**STEP 1** 打开 SSMS, 连接到数据库服务器, 依次展开“数据库”节点、要添加主键和外键的表所属的数据库(这里是 test 数据库)、“表”文件夹, 如果在新建表的同时创建主键和外键, 右击并在快捷菜单中选择“新建”|“表”命令, 如图 1-16 所示。在出现如图 1-17 所示的界面中, 右击要作为主键或外键的列, 在弹出的快捷菜单中选择“设置主键”命令, 如图 1-33 所示, 这样就将选择的列设置为主键了。

**STEP 2** 创建外键的过程稍微要复杂一些, 在如图 1-33 所示的界面中, 选择“关系”命令, 在出现的界面中单击“添加”按钮添加关系, 如图 1-34 所示。

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06



图 1-33 创建表主键

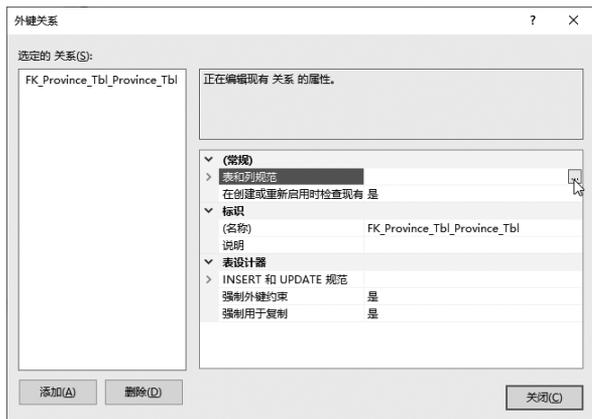


图 1-34 外键关系

**STEP 3** 在图 1-34 中,选择新添加的关系,在右边的关系属性中,单击“表和列规范”编辑框旁的...按钮,出现如图 1-35 所示的界面。在“主键表”下拉列表框中选择主键所在的表,如 Area\_Tbl,当鼠标移到下面的编辑框中时,将出现一个下拉列表框,列出所选主键表中的所有字段,选择主键,如 AreaID,然后将鼠标移动到右边“外键表”下面的编辑框中,在出现的下拉列表框中选择相应的字段,如 AreaID,单击“确定”按钮,这样外键表中的选中字段就成为外键,外键就这样被创建了。

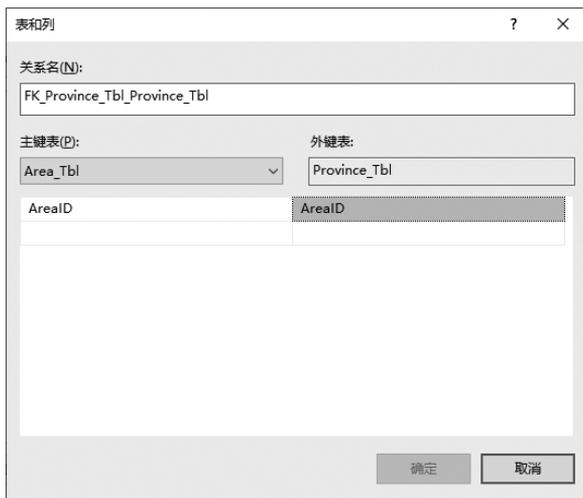


图 1-35 表和列规范

**STEP 4** 通过修改表来设置表的主键,操作方式和创建表时创建主键一样。在 SSMS 的“对象资源管理器”窗口中,依次展开“数据库”节点、要创建主键和外键的表所属的数据库(这里是 test 数据库)、“表”文件夹,右击要创建主键和外键的表(这里是 Province\_Tbl),在弹出的快捷菜单中选择“设计”命令,出现表设计器(参考前面修改表的项目)。此后,创建主键和外键的方法同新建表时创建主键和外键的方法一样。

下面来看看如何通过编码的方式在查询分析器中编写 T-SQL 语句创建主键和外键。以下是在创建表 Province\_Tbl 的同时创建外键 AreaID 和主键 ProvinceID 的

T-SQL语句:

```
CREATE TABLE [dbo].[Province_Tbl](
    [ProvinceId] [int] IDENTITY(1,1) NOT NULL,
    [AreaID] [int] NULL,
    [ProvinceName] [varchar](50) NULL,
CONSTRAINT [PK_Province_Tbl] PRIMARY KEY CLUSTERED
(
    [ProvinceId] ASC
)
) ON [PRIMARY]
```

相关参数说明如下:

- CONSTRAINT:表示要创建一个约束。
- PRIMARY KEY:表示该约束是一个主键约束。

创建外键的 T-SQL 语句如下:

```
ALTER TABLE [dbo].[Province_Tbl] WITH CHECK
ADD CONSTRAINT [FK_Province_Tbl_Area_Tbl] FOREIGN KEY([AreaID])
REFERENCES [dbo].[Area_Tbl] ([AreaID])
```

创建外键都是通过修改表,向表添加外键约束来实现的。

通过修改表实现创建主键和外键。创建主键的 T-SQL 语法如下:

```
ALTER TABLE table
[ WITH CHECK | WITH NOCHECK ]
    ADD CONSTRAINT constraint_name
PRIMARY KEY [ CLUSTERED | NONCLUSTERED ]
( column [ ,...n ] )
[ WITH FILLFACTOR = fillfactor ]
[ ON { filegroup | DEFAULT } ]
]
```

相关参数说明如下:

- ALTER TABLE:表示要修改表。table 是要修改的表名。
- WITH CHECK|WITH NOCHECK:为可选项,表示在创建主键时是否要检验表中已有数据是否符合主键约束的要求,即数据的唯一性。
- ADD CONSTRAINT:表示要添加约束。constraint\_name:要添加的约束的名称。
- PRIMARY KEY:标明该约束是主键约束。
- column:要作为主键的字段,可以是多个字段,字段间用逗号分隔。
- WITH FILLFACTOR = fillfactor:为可选项,为主键索引指定填充因子。

例如,在创建表 Province\_Tbl 后,为表 Province\_Tbl 创建主键 PK\_Province\_Tbl,T-SQL 语句如下:

```
ALTER TABLE [dbo].[Province_Tbl]
WITH CHECK ADD CONSTRAINT [PK_Province_Tbl]
PRIMARY KEY([ProvinceId])
```

创建外键的 T-SQL 语句如下:

```
ALTER TABLE table
[ WITH CHECK | WITH NOCHECK ]
ADD CONSTRAINT constraint_name
FOREIGN KEY
    ( ( column [ ,...n ] ) )
REFERENCES ref_table [ ( ref_column [ ,...n ] ) ]
[ ON DELETE { CASCADE | NO ACTION } ]
[ ON UPDATE { CASCADE | NO ACTION } ]
[ NOT FOR REPLICATION ]
```

相关参数说明如下:

- ALTER TABLE:为主关键字,表示要修改表。table:所要修改的表的名称。
- WITH CHECK | WITH NOCHECK:为可选项,表示在创建外键约束时是否对表已有数据进行规则检验。
- ADD CONSTRAINT:为关键字,表示要添加约束。constraint\_name:要添加的约束的名称。FOREIGN KEY:为关键字,表示添加外键。
- column:要作为外键的表字段,可以是多个字段,字段间用逗号分隔。
- REFERENCES:为关键字,表示要参照表。
- ref\_table:表示要参照的表的名称。ref\_column:表示要参照的字段。
- [ ON DELETE { CASCADE | NO ACTION } ],[ ON UPDATE { CASCADE | NO ACTION } ],[ NOT FOR REPLICATION ]:为可选项,分别表示对表数据做删除、更新和复制时采取的级联动作。

例如,在表 Province\_Tbl 创建好后,要在表字段 AreaID 上创建外键 FK\_Province\_Tbl\_Area\_Tbl 和表 Area\_Tbl 关联,T-SQL 语句如下:

```
ALTER TABLE [dbo].[Province_Tbl]
WITH CHECK ADD CONSTRAINT [FK_Province_Tbl_Area_Tbl] FOREIGN KEY
([AreaID])
REFERENCES [dbo].[Area_Tbl] ([AreaID])
```

## 2. 删除主键和外键

当要删除对列或列组合中的输入值的唯一性要求时,应删除主键约束。当要删除强制引用完整性要求时,应删除外键。

1)在 SSMS 中删除主键的步骤如下:

**STEP 1** 打开 SSMS,连接到数据库服务器,在“对象资源管理器”窗口中展开“数据库”|test|“表”文件夹,右击要删除主键的表 Area\_Tbl,选择“设计”命令,如图 1-36所示。

**STEP 2** 在出现的界面中,右击主键所在的字段,在弹出的快捷菜单中选择“删除主键”命令,然后保存对表的修改,这样主键就从表中删除了,如图 1-37 所示。



图 1-36 修改表

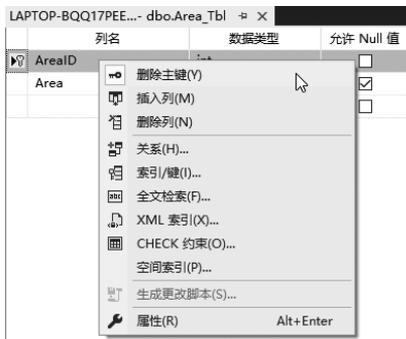


图 1-37 删除主键

2)在 SSMS 中删除外键约束的步骤如下:

**STEP 1** 在如图 1-36 所示的界面中,展开表下面的“键”节点并右击,在弹出的快捷菜单中选择“删除”命令,如图 1-38 所示。

**STEP 2** 在接着出现的界面中单击“确定”按钮,删除外键约束,如图 1-39 所示。



图 1-38 删除外键约束

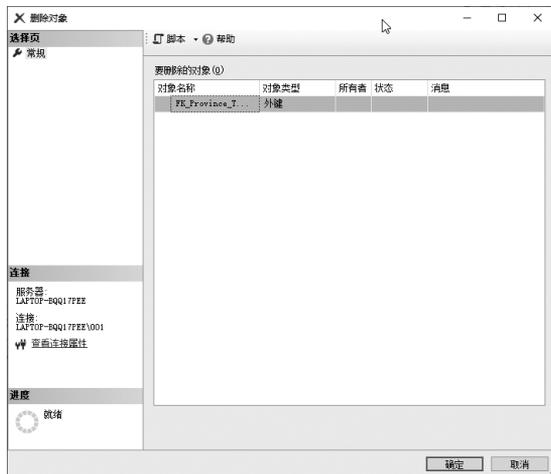


图 1-39 确认删除对象

同样的,删除主键、外键约束也可以在查询分析器中通过编写 T-SQL 语句来进行。删除主键、外键约束的 T-SQL 语句如下:

```
ALTER TABLE table
DROP [ CONSTRAINT ] constraint_name
```

相关参数说明如下:

- ALTER TABLE: 为主关键字,表示要修改表。table: 要修改的表名。
- DROP [ CONSTRAINT ]: 表示要删除约束,CONSTRAINT 是可选的关键字。
- constraint\_name: 要删除的约束的名称。

例如,删除表 Area\_Tbl 中的主键 PK\_Area\_Tbl, T-SQL 语句如下:

```
ALTER TABLE Area_Tbl
DROP CONSTRAINT PK_Area_Tbl
```

删除表 Area\_Tbl 中的外键约束 FK\_Province\_Tbl\_Area\_Tbl, T-SQL 语句如下:

```
ALTER TABLE Area_Tbl
DROP CONSTRAINT FK_Province_Tbl_Area_Tbl
```



## 思政园地

2022年11月29日23时08分,搭载神舟十五号载人飞船的长征二号F遥十五运载火箭在酒泉卫星发射中心点火发射,约10分钟后,神舟十五号载人飞船与火箭成功分离,进入预定轨道,发射取得成功。11月30日7时33分,神舟十五号3名航天员顺利进驻中国空间站,与神舟十四号航天员乘组首次实现“太空会师”。你知道神舟十五号的任务是什么吗?神舟十五号的飞行任务有6项,其中一项就是对空间站开展常态化的平台测试、维护及站务管理工作。空间站上的设施和设备需要维护和修理,同样地,数据库中的表也需要维护。

## 项目小结

本项目主要讲解了 SQL Server 2022 环境的搭建,以及在 SQL Server 2022 中如何创建和维护数据库,并从创建数据库表、修改和删除数据库表、认识主键和外键等方面做了具体介绍。

## 项目考核



## 填空题

1. 数据实际上就是\_\_\_\_\_。
2. 数据库是数据的集合,它具有统一的\_\_\_\_\_并存放于统一的\_\_\_\_\_内,是多种应用数据的集成,并可被各个应用程序所共享。
3. 由于\_\_\_\_\_约束确保唯一数据,因此经常用来定义标识列。



## 简答题

1. 简述数据和数据库的概念。
2. 简述创建数据库表的意义。
3. 简述主键和外键在数据库中的地位,并简述主键和外键的创建过程。



## 实验题

假设有一个在线教育平台的数据库 schoolManagement,用于记录课程信息、班级信息、教师信息、学生信息及学生成绩信息。schoolManagement 中有以下五个表。

课程表(Courses):CourseID(课程编号,主键),CourseName(课程名称)。

班级表(Class):ClassID(班级编号,主键),ClassName(班级名称)。

教师表(Teachers):TeacherID(教师编号,主键),TeacherName(教师姓名),Class-

Chapter  
01Chapter  
02Chapter  
03Chapter  
04Chapter  
05Chapter  
06

sID(所属班级编号,外键),CourseID(讲授课程编号,外键),Department(所属部门)。

学生表(Students):StudentID(学生编号,主键),StudentName(学生姓名),Gender(性别),Birthday(出生日期),Telephone(电话),ClassID(所属班级编号,外键)。

成绩表(Grades):GradeID(成绩编号,主键),StudentID(学生编号,外键),CourseID(课程编号,外键),Grade(成绩)。

实验要求:请根据以下任务要求,编写相应的 T-SQL 语句,并在数据库中进行验证。

任务 1:创建表。

按上述描述创建五个表,其中教师表的外键字段 ClassID 和 CourseID 分别关联班级表的 ClassID 和课程表的 CourseID;学生表的外键字段 ClassID 关联班级表的 ClassID;成绩表的外键字段 StudentID 和 CourseID 分别关联学生表的 StudentID 和课程表的 CourseID。

任务 2:删除外键约束。

(1)删除成绩表中的 StudentID、CourseID 外键约束。

(2)删除学生表中的 ClassID 外键约束。

提示:删除外键约束前,可使用 SHOW CREATE TABLE 语句确认外键约束名。删除外键约束后,可使用 SHOW CREATE TABLE 语句验证外键约束是否已删除。

任务 3:重新创建外键约束。

(1)使用 ALTER TABLE 语句重新为成绩表的 StudentID、CourseID 字段添加外键约束。

(2)使用 ALTER TABLE 语句重新为学生表的 ClassID 字段添加外键约束。

提示:重新创建外键约束后,可使用 SHOW CREATE TABLE 语句验证外键约束是否正确添加。

任务 4:删除主键约束并重新创建。

(1)使用 ALTER TABLE 语句删除课程表的主键约束。

(2)使用 ALTER TABLE 语句重新为课程表的 CourseID 字段创建主键约束。