普通高等教育“十三五”应用型人才培养规划教材

数据库技术及应用教程
(SQL Server 2008)

◎ 主编　吴慧婷　定会





机械工业出版社
CHINA MACHINE PRESS

需要全本请访问 www.ertongbook.com

普通高等教育“十三五”应用型人才培养规划教材

数据库技术及应用教程（SQL Server 2008）

主　编　吴慧婷　定　会
副主编　李吴松　喻　晓
参　编　孙　敏

常州大学图书馆
藏书章

机械工业出版社

本书结合流行的SQL Server 2008关系数据库管理系统，重点介绍了数据库的基本原理和技术方法。本书共分为12章，包括数据库系统概述、SQL Server 2008概述、T-SQL基础、创建和管理数据库、架构和数据表、数据查询、索引与视图、存储过程与触发器和用户自定义函数、事务与游标、数据库安全性、数据库备份与恢复和数据库设计等。

本书适合作为应用型本科院校计算机类、通信类、信息类和电子商务类等专业的数据库相关课程的教材，也可作为广大开发人员的参考教材。

本书配有电子课件，欢迎选用本书作教材的老师登录www.cmpedu.com注册下载，或发邮件至jinacmp@vip.163.com索取。

图书在版编目（CIP）数据

数据库技术及应用教程：SQL Server 2008/吴慧婷，定会主编. —北京：机械工业出版社，2018.6

普通高等教育“十三五”应用型人才培养规划教材

ISBN 978-7-111-59441-3

Ⅰ.①数… Ⅱ.①吴…②定… Ⅲ.①关系数据库系统－高等学校－教材 Ⅳ.①TP311.132.3

中国版本图书馆CIP数据核字（2018）第052118号

机械工业出版社（北京市百万庄大街22号 邮政编码100037）

策划编辑：吉 玲 责任编辑：吉 玲 刘丽敏

责任校对：刘 岚 封面设计：张 静

责任印制：孙 炜

北京玥实印刷有限公司印刷

2018年6月第1版第1次印刷

184mm×260mm · 17.75印张 · 429千字

标准书号：ISBN 978-7-111-59441-3

定价：45.00元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

服务咨询热线：010-88379833

读者购书热线：010-88379649

网络服务

机 工 官 网：www.cmpbook.com

机 工 官 博：weibo.com/cmp1952

教育服务网：www.cmpedu.com

金 书 网：www.golden-book.com

封面无防伪标均为盗版

前 言

数据库技术是数据管理的新技术，已经成为计算机系统里应用最广的技术之一。在部分本科地方院校向应用技术型转型的背景下，我们开始了本书的编写工作，并定位于应用技术型。本教材主要面向普通本科院校的学生编写，编者均来自院校的一线授课老师，在内容深度、系统结构、案例选择、编写方法等方面进行了深入细致的调研，以满足数据库技术的教学需要。

本书以学生成绩管理系统实例为主线，贯穿于各章的讲解，讲解过程循序渐进、深入浅出。本书介绍了数据库基础知识和数据库创建、表的操作、数据查询、索引、视图、存储过程、触发器、SQL Server 函数、事务与游标、数据库安全性、数据库备份与恢复和数据库设计等内容，并配有例题和练习题，学生通过本书可更好地学习和掌握数据库的开发和使用。

本书以介绍 SQL Server 2008 关系数据库管理系统为主，全书共分为 12 章。第 1 ~ 3 章由喻晓和孙敏编写，第 4 ~ 6 章由李吴松编写，第 7 ~ 9 章由吴慧婷编写，第 10 ~ 12 章由定会编写。吴慧婷、定会作为本书的主编，负责全书的策划和修改定稿工作。

本书以应用型本科人才培养为导向，在内容的选择、深度的把握上力求做到循序渐进。本书以帮助学生建立扎实的技术基础、培养学生将数据库技术运用到实际项目中的能力为编写目的，涉及的技术内容重难点突出。本书提供了大量翔实且便于融会贯通的实际案例，并附有运行结果，方便学生深入掌握数据库的基本原理和应用技术。每章均配有习题，相关章配有上机练习。本书条理清楚、重难点突出、实用性强，适合作为高等院校计算机类、通信类等专业数据库课程的教材，也可供广大技术人员及自学者参考。

由于作者水平有限，书中难免存在错误或不足之处，敬请读者批评指正。

编 者

目 录 Contents

前 言

第1章 数据库系统概述 …… 1

1.1 初识数据库 …… 1

1.1.1 数据、信息与数据处理 …… 1

1.1.2 数据管理技术的发展 …… 2

1.1.3 数据库系统的组成 …… 3

1.2 数据模型 …… 4

1.2.1 数据模型概述 …… 4

1.2.2 概念层数据模型 …… 4

1.2.3 组织层数据模型 …… 7

1.3 关系模型 …… 8

1.3.1 关系模型的数据结构 …… 8

1.3.2 关系模型的数据操作 …… 10

1.3.3 关系模型的完整性约束 …… 10

1.4 关系代数基础 …… 11

1.4.1 传统的集合运算 …… 12

1.4.2 专门的关系运算 …… 13

1.4.3 用关系代数实现关系查询 …… 15

1.5 数据库系统的体系结构 …… 16

1.5.1 数据库系统的内部体系结构 …… 16

1.5.2 数据库系统的外部体系结构 …… 18

1.6 常见的关系数据库 …… 18

1.6.1 Access 数据库 …… 18

1.6.2 SQL Server 数据库 …… 19

1.6.3 MySQL 数据库 …… 19

1.6.4 Oracle 数据库 …… 19

习题 …… 19

第2章 SQL Server 2008 概述 …… 21

2.1 SQL Server 2008 简介 …… 21

2.1.1 SQL Server 的发展 …… 21

2.1.2 SQL Server 2008 版本及系统需求 …… 22

2.1.3 SQL Server 2008 的体系结构 …… 23

2.1.4 实例 …… 23

2.2 SQL Server 2008 的安装与启动 …… 23

2.2.1 SQL Server 2008 的安装 …… 23

2.2.2 SQL Server 2008 服务器服务的启动和停止 …… 36

2.3 SQL Server 2008 的主要管理工具 …… 37

2.3.1 SQL Server Management Studio …… 38

2.3.2 SQL Server Analysis Services …… 39

2.3.3 SQL Server 配置管理器 …… 39

2.3.4 SQL Server 文档和教程 …… 39

习题 …… 39

第3章 T-SQL 基础 …… 40

3.1 T-SQL 概述 …… 40

3.1.1 SQL 概述 …… 40

3.1.2 T-SQL 的发展 …… 40

3.1.3 T-SQL 的特点 …… 41

3.1.4 T-SQL 的分类 …… 41

3.2 系统提供的数据类型 …… 41

3.2.1 数值类型 …… 41

3.2.2 字符串类型 …… 42

3.2.3 日期和时间类型 …… 43

3.2.4 货币类型 …… 44

3.3 用户自定义数据类型 …… 44

3.3.1 创建用户自定义数据类型 …… 44

3.3.2 删除用户自定义数据类型 …… 45

3.4 T-SQL 语法要素 …… 45

3.4.1 标识符 …… 45

3.4.2 常量 …… 45

3.4.3 变量 …… 45

3.4.4 注释 …… 47

3.4.5 批处理 …… 48

3.4.6 脚本 …… 48

3.4.7 运算符与表达式 …… 48

3.5 T-SQL 函数 …… 49

3.5.1 数学函数 …… 49

3.5.2 聚合函数 …… 49

3.5.3 时间日期函数 …… 49

3.5.4 字符串函数 …… 50

3.5.5 转换函数 …… 51

需要全本请访问 www.ertongbook.com

3.5.6 其他函数 …… 51
3.6 T-SQL 流程控制语句 …… 51
3.6.1 BEGIN...END 语句 …… 52
3.6.2 IF...ELSE 语句 …… 52
3.6.3 CASE 语句 …… 52
3.6.4 WHILE、BREAK 和 CONTINUE 语句 …… 54
习题 …… 54
第 4 章 创建和管理数据库 …… 56
4.1 SQL Server 数据库概述 …… 56
4.1.1 系统数据库 …… 56
4.1.2 数据库的文件和文件组 …… 57
4.1.3 数据库文件的属性 …… 58
4.1.4 常见数据库对象 …… 58
4.2 创建数据库 …… 60
4.2.1 用图形化方法创建数据库 …… 60
4.2.2 用 T-SQL 语句创建数据库 …… 65
4.2.3 查看和设置数据库选项 …… 69
4.3 修改数据库 …… 72
4.3.1 扩展数据库 …… 72
4.3.2 收缩数据库 …… 73
4.3.3 创建和更改文件组 …… 77
4.3.4 删除数据库 …… 80
4.3.5 分离和附加数据库 …… 81
习题 …… 83
第 5 章 架构和数据表 …… 85
5.1 创建和管理架构 …… 85
5.1.1 创建架构 …… 85
5.1.2 在架构间传输对象 …… 86
5.1.3 删除架构 …… 86
5.2 创建和管理数据表 …… 87
5.2.1 设计表结构 …… 87
5.2.2 创建数据表 …… 88
5.2.3 修改表结构 …… 91
5.2.4 删除表 …… 92
5.2.5 完整性与约束 …… 93
5.3 分区表 …… 98
5.3.1 基本概念 …… 98
5.3.2 创建分区表 …… 99
习题 …… 103
第 6 章 数据查询 …… 105
6.1 单表查询 …… 105
6.1.1 查询语句的基本结构 …… 105
6.1.2 对列的查询 …… 106
6.1.3 对行的查询 …… 108
6.1.4 对查询结果排序 …… 112
6.1.5 使用聚合函数 …… 112
6.1.6 对查询结果分组 …… 113
6.1.7 HAVING 子句 …… 114
6.1.8 COMPUTE 和 COMPUTE BY 子句 …… 115
6.1.9 使用查询结果创建新表 …… 116
6.2 连接查询 …… 117
6.2.1 内连接 …… 117
6.2.2 自连接 …… 119
6.2.3 外连接 …… 120
6.3 子查询 …… 122
6.3.1 嵌套子查询 …… 122
6.3.2 相关子查询 …… 126
6.3.3 其他形式的子查询 …… 128
6.4 集合查询 …… 129
6.4.1 并运算 …… 130
6.4.2 交运算 …… 131
6.4.3 差运算 …… 131
6.5 添加和管理表数据 …… 132
6.5.1 使用图形化界面管理表数据 …… 132
6.5.2 使用 T-SQL 语句插入表数据 …… 132
6.5.3 使用 T-SQL 语句更新表数据 …… 134
6.5.4 使用 T-SQL 语句删除表数据 …… 135
习题 …… 135
第 7 章 索引与视图 …… 137
7.1 索引 …… 137
7.1.1 索引的概念与作用 …… 137
7.1.2 索引的存储结构及分类 …… 138
7.1.3 创建索引 …… 140
7.1.4 查看与修改索引 …… 144
7.1.5 删除索引 …… 146
7.2 视图 …… 147
7.2.1 视图的概念与作用 …… 147
7.2.2 创建视图 …… 147
7.2.3 修改视图 …… 153
7.2.4 删除视图 …… 154
7.2.5 使用视图 …… 155
习题 …… 160
第 8 章 存储过程与触发器和用户自定义函数 …… 161
8.1 存储过程 …… 161
8.1.1 存储过程概述 …… 161

8.1.2 创建存储过程 …… 162
8.1.3 调用存储过程 …… 165
8.1.4 管理存储过程 …… 167
8.1.5 常见的存储过程 …… 169
8.2 触发器 …… 169
8.2.1 触发器概述 …… 169
8.2.2 创建触发器 …… 170
8.2.3 DML 触发器示例 …… 173
8.2.4 DDL 触发器示例 …… 174
8.2.5 管理触发器 …… 174
8.3 用户自定义函数 …… 176
8.3.1 基本概念 …… 176
8.3.2 创建和调用标量值函数 …… 177
8.3.3 创建和调用内联表值函数 …… 179
8.3.4 创建和调用多语句表值函数 …… 179
8.3.5 查看、修改和删除用户自定义函数 …… 181
习题 …… 183
第 9 章 事务与游标 …… 184
9.1 事务 …… 184
9.1.1 事务概述 …… 184
9.1.2 管理事务 …… 184
9.1.3 并发操作与数据不一致性 …… 188
9.1.4 锁机制 …… 188
9.2 游标 …… 190
9.2.1 游标的概念 …… 190
9.2.2 游标的操作 …… 190
9.2.3 利用游标修改和删除表数据 …… 194
9.2.4 游标的示例 …… 195
习题 …… 197
第 10 章 数据库安全性 …… 198
10.1 身份验证模式 …… 198
10.1.1 Windows 身份验证 …… 198
10.1.2 混合身份验证 …… 198
10.1.3 设置身份验证模式 …… 199
10.2 登录账户管理 …… 200
10.3 用户管理 …… 208
10.4 权限管理 …… 211
10.4.1 对象权限的管理 …… 212
10.4.2 语句权限的管理 …… 218
10.5 角色管理 …… 227
10.5.1 固定服务器角色 …… 228
10.5.2 固定数据库角色 …… 229
10.5.3 自定义数据库角色 …… 230
10.5.4 应用程序角色 …… 237
习题 …… 237
第 11 章 数据库备份与恢复 …… 238
11.1 数据库备份 …… 238
11.1.1 备份类型 …… 238
11.1.2 备份策略 …… 239
11.1.3 创建备份设备 …… 240
11.1.4 备份操作 …… 242
11.2 数据库恢复 …… 246
11.2.1 故障类型及恢复模式 …… 246
11.2.2 恢复操作 …… 247
习题 …… 252
第 12 章 数据库设计 …… 253
12.1 数据库设计概述 …… 253
12.2 需求分析 …… 254
12.2.1 需求分析的方法与步骤 …… 254
12.2.2 数据流图 …… 255
12.2.3 数据字典 …… 256
12.3 概念结构设计 …… 258
12.3.1 概念结构设计的方法与步骤 …… 258
12.3.2 局部 E-R 模型设计 …… 258
12.3.3 全局 E-R 模型设计 …… 259
12.3.4 优化全局 E-R 模型 …… 260
12.4 逻辑结构设计 …… 261
12.4.1 E-R 模型向关系模型的转换 …… 261
12.4.2 关系模型的优化 …… 263
12.4.3 设计外模式 …… 263
12.5 函数依赖与关系的规范化 …… 263
12.5.1 问题的提出 …… 264
12.5.2 函数依赖 …… 265
12.5.3 关系规范化 …… 267
12.6 物理结构设计 …… 270
12.6.1 确定数据库存取方法 …… 270
12.6.2 确定数据库存储结构 …… 271
12.7 数据库的实施及维护 …… 272
12.7.1 数据库的实施 …… 272
12.7.2 数据库的运行和维护 …… 272
习题 …… 273
参考文献 …… 275

第 1 章 数据库系统概述

本章首先对数据库的基本概念、数据管理技术发展的三个阶段、数据库系统的组成、数据模型等进行介绍；然后讲解关系代数基础，内容包括传统的集合运算和专门的关系运算；最后简单介绍常用的关系数据库管理系统，包括 Access 数据库、SQL Server 数据库、MySQL 数据库和 Oracle 数据库。

1.1 初识数据库

数据库技术产生于 20 世纪 60 年代末，是数据管理的最新技术，是计算机科学的重要分支。数据库技术是信息系统的核心和基础，它的出现极大地促进了计算机应用向各行各业的渗透。数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家信息化程度的重要标志。在进入数据库应用之前，首先来了解数据库中最常用的概念和术语。

1.1.1 数据、信息与数据处理

1. 数据

数据（Data）是数据库中存储的基本对象。数据是描述客观事物特征的一系列符号，包括数字、文本、图形、图像、音频、视频、档案记录等不同形式。从计算机的角度来看，数据是计算机能够识别并处理的符号。数据的含义称为数据的语义，数据与其语义是不可分的。例如，99 是一个数据，其语义可以表示一个学生的计算机考试成绩，也可以表示计算机系学生的人数。

2. 信息

通过解释、分析归纳、演绎推导等手段从数据中抽离出对人有意义和价值的数据称为信息。信息是人们能够理解的经过加工后的数据。在信息社会中，信息是对人有用的一种重要资源。

3. 数据处理

数据处理是将数据转换为信息的过程。数据处理的基本环节包括对数据的收集、存储、加工、分类、检索、传播等系列活动。从数据处理的角度来说，信息是一种被加工成特定形式的数据，可以得到以下的表达式：

$$\text{信息} = \text{数据} + \text{数据处理}$$

数据处理工作分为三类：数据管理、数据加工和数据传播。

（1）数据管理

数据管理的主要任务是收集信息，用数据表示信息并按类别组织保存，为各种数据处理

快速、正确地提供必要的数据。

(2) 数据加工

数据加工的主要任务是对数据进行变换、抽取和运算，得到更有用的数据。

(3) 数据传播

通过数据传播，信息在空间或时间上以各种形式传递。数据传播过程中，数据的结构、性质和内容不变。

数据处理的中心问题是数据管理，数据管理是其他数据处理的核心和基础。数据管理工作包括组织和保存数据、进行数据维护、提供数据查询和数据统计功能。随着计算机硬件及软件的发展，计算机在数据管理方面经历了从低级到高级的发展过程，数据管理技术的优劣将直接影响数据处理的效率。

1.1.2 数据管理技术的发展

随着应用需求的不断发展，数据管理技术的发展经历了三个阶段：人工管理阶段、文件系统阶段和数据库系统阶段。

1. 人工管理阶段

20 世纪 40 年代中期至 50 年代中期，由于计算机的应用主要是进行科学计算，没有直接存取存储设备，在硬件之上也没有操作系统，采用批处理的方式处理数据。人工管理数据阶段具有以下特点：

1) 数据不保存。当时计算机用于科学计算，数据不需要长期保存，所以只需要计算某个课题时将数据输入，程序结束时撤销数据。

2) 应用程序控制管理数据，程序员负担重。在这种管理方式下，数据由应用程序管理，数据的输入/输出方式、结构和格式、存储方式都需要在程序中设定，没有操作系统软件进行数据管理。

3) 无共享、冗余度极大。数据是面向应用的，则一批数据只针对一个应用程序，多个应用程序之间没有通信或消息传递，数据不共享，各应用程序间存在大量重复数据，即冗余度极大。

4) 不独立，完全依赖于程序。数据与应用程序具有一一对应的关系。在人工管理阶段，数据变化时需要修改程序，数据完全依赖于程序，数据不具有独立性。

人工管理阶段应用程序与数据之间的对应关系如图 1-1 所示。

2. 文件系统阶段

20 世纪 50 年代末期至 60 年代中期，随着硬件设备的发展，存储设备出现了磁盘和磁鼓等，软件方面出现了高级语言和操作系统软件，这时的计算机不仅用于科学计算，也开始用于企业管理。

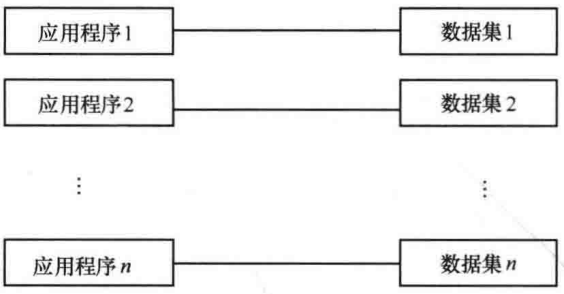

图 1-1 人工管理阶段应用程序与数据之间的对应关系

文件系统阶段数据处理的特点如下：

1) 数据可长期保存。

2) 由文件系统管理数据。

3) 共享性差、冗余度大。

4）数据的独立性差，数据的逻辑结构改变必须修改应用程序。

5）数据由应用程序控制。

文件系统阶段应用程序与数据之间的对应关系如图 1-2 所示。

3. 数据库系统阶段

20 世纪 60 年代末以来，随着计算机应用领域不断在广度和深度进行扩展，数据处理技术不仅限于文件管理系统，并且随着磁盘技术取得突飞猛进的发展，数据库系统成为了存储数据、管理数据和分析数据的主要工具。

数据库系统阶段数据处理的特点如下：

1）数据结构化，采用数据模型表示复杂的数据结构。

2）共享性高、冗余度小、易扩充。

3）有较高的数据独立性。

4）数据由数据库管理系统统一管理和控制。

数据库系统阶段应用程序与数据之间的对应关系如图 1-3 所示。

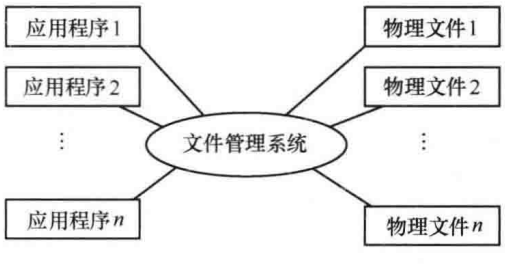

图 1-2　文件系统阶段应用程序与数据之间的对应关系

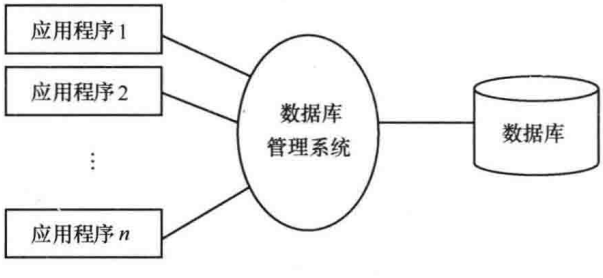

图 1-3　数据库系统阶段应用程序与数据之间的关系

1.1.3　数据库系统的组成

数据库系统（Database System，DBS）是一个按照数据库方式存储、维护和向应用系统提供数据支持的系统，是在计算机系统中引入数据库后的系统。数据库系统一般由数据库、数据库管理系统、应用系统、数据库管理员和用户组成，也可将数据库系统视为数据库、软件、硬件和数据库管理员的集合体。

数据库（Database，DB）是长期存储在计算机内有组织的、可共享的大量相关数据的集合。数据库中的数据按一定的数据模型组织、描述和储存，具有较小冗余度、数据间联系紧密而又有较高的数据独立性等特点。

数据库管理系统（Database Management System，DBMS）是位于用户与操作系统之间的一层数据管理软件，其功能包括数据库的定义、数据库的操纵、数据库的运行管理、数据库的建立和维护。

计算机的硬件环境包括数据库对硬件的要求，如内存容量、硬盘大小；软件环境包括支持数据库运行的操作系统软件、具有数据库支持的编译环境和应用开发工具。

数据库系统相关的人员包括数据库管理员和用户等。数据库管理员（Database Administrator，DBA）是监督和管理数据库系统的一组人员，负责数据库系统的全面管理和控制。用户是通过应用程序界面使用数据库的人员。

1.2 数据模型

1.2.1 数据模型概述

日常生活中在玩具店能够看到飞机或汽车模型，在售楼部可以看到商品楼房的模型。模型是对现实生活中的事物进行抽象模拟后的形态。数据库中的数据是按一定的方式存储在一起的，其组织结构称为数据模型，它决定了数据库中数据之间关联的表达方式，是对数据的一种抽象表达。数据模型要求能比较真实地模拟现实世界，容易被人理解，并且便于在计算机上实现。

数据模型可以分为以下两类：

第一类是概念层数据模型：从数据的语义视角来抽取模型。此模型是按用户的观点来对数据和信息进行建模，主要用在数据库的设计阶段。

第二类是组织层数据模型：从数据的组织层次来描述数据，主要包括层次模型、网状模型、关系模型以及面向对象模型。此模型是从计算机系统的观点对数据进行建模，主要用于DBMS的实现。

数据模型是数据库系统的核心和基础。DBMS都基于某种数据模型。为了把现实世界中的具体事物抽象、组织为某一具体DBMS支持的数据模型，通常首先将现实世界抽象为信息世界（概念层模型），然后再将信息世界转换为机器世界（组织层模型）。概念层模型与DBMS无关，组织层模型与DBMS有关。

数据模型应满足以下三个条件：

1）能比较真实地模拟现实世界；

2）容易被人们理解；

3）便于在计算机上实现。

数据模型一般由数据结构、数据操作和数据完整性约束三部分组成，这三部分就称为数据模型的三要素。

数据结构包括两类：一类是与数据类型、内容、性质有关的对象，如关系模型中的域、属性和关系等；另一类是与数据之间联系有关的对象，它从数据组织层表达数据的结构。数据结构是对系统静态特性的描述。

数据操作是对系统动态特性的描述，包括数据查询和数据更改。数据查询是在数据集合中提取用户感兴趣的内容，不改变数据结构与数据值。数据更改包括插入、删除和修改数据，此类操作改变数据的值。

数据完整性约束是对系统静态特性的描述，是一组完整性规则的集合。完整性规则是数据及其联系所具有的制约和依存规则，用以保证数据的正确、有效和相容，使数据库中的数据值与现实情况相符。

1.2.2 概念层数据模型

概念层数据模型也称信息模型，它是按用户的观点来对数据和信息建模。它完全不涉及信息在计算机系统中的表示，是现实世界的抽象，只是描述某个信息结构。概念层数据模型

需要全本请访问 www.ertongbook.com

是理解数据库和设计数据库的基础，也是用户和数据库设计人员交流的工具。概念层数据模型用于信息世界的建模，是现实世界到信息世界的第一层抽象，也是面向用户、面向现实世界的数据模型，与 DBMS 无关。

目前描述概念层数据模型最常用的方法是实体-联系（Entity-Relationship）方法，即 E-R 方法，使用的工具称为 E-R 图。E-R 图所描述的现实世界的信息结构称为实体-联系模型（E-R 模型），所涉及的基本概念有实体（Entity）、属性（Attribute）、键（Key）、域（Domain）、联系（Relationship）等。

1. 实体

实体通常是客观存在并且可以互相区分的事务。其可以是实际的事务，如一名学生、一本书等；也可以是抽象的事件，如一个创意等。

2. 属性

属性是描述对象的某个特性，如学生的学号、姓名、性别、出生日期、籍贯等。

3. 键

键又称为码、关键字、关键码等，是区别实体的唯一标识，如学号、图书证号等。

4. 域

域是实体中相应属性的取值范围，如性别属性的域为｛男，女｝。

5. 联系

联系是实体间的相互关系，反映了客观事物间相互依存的状态。概念模型中主要解决的问题是实体间的联系。基本联系有三种，即实体间一对一的联系、一对多的联系和多对多的联系。

1）一对一联系（1∶1）：如果对于实体集 A 中的每一个实体，实体集 B 中至多有一个实体与之联系，反之亦然，则称实体集 A 和实体集 B 具有一对一联系，记为 1∶1。例如，观众与座位、乘客与车票、病人与病床等都是一对一联系。

2）一对多联系（1∶n）：如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体（n≥0）与之联系，而对于实体集 B 中的每一个实体，实体集 A 中至多只有一个实体与之联系，则称实体集 A 和实体集 B 具有一对多联系，记为 1∶n。例如，学院与教师、班级与学生等都是一对多的联系。

3）多对多联系（m∶n）：如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体（n≥0）与之联系，而对于实体集 B 中的每一个实体，实体集 A 中也有 m（m≥0）个实体与之联系，则称实体集 A 和实体集 B 具有多对多联系，记为 m∶n。例如，学生与课程、商品与顾客等都是多对多联系。

E-R 方法用来描述现实世界的概念模型。E-R 图提供了表示实体、属性和联系的方法。E-R 模型可以描述多个实体之间的关系。

实体：用矩形表示，矩形框内写明实体名。矩形框中写明学生，表示学生实体，如图 1-4 所示。

属性：用椭圆形表示，并用无向边将其与相应的实体连接起来，如图 1-5 所示。

如图 1-5 所示，椭圆形中的学号、姓名、性别、系和年龄都是学生实体的属性，所以使用无向边将其与学生实体连接起来。实体的一个属性或属性组的唯一标识称为实体的码。学号就是学生实体的码。

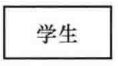

图 1-4　实体示例

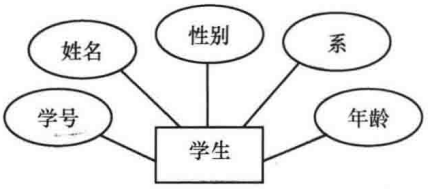

图 1-5　实体属性示例

联系：用菱形表示，菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时在无向边旁标上联系的类型（1∶1、1∶n、m∶n），如图 1-6 所示。

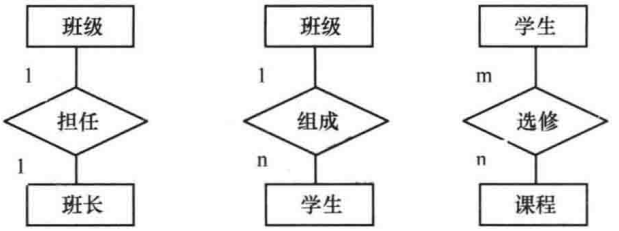

图 1-6　不同类型的联系

E-R 模型不仅能描述两个实体间的联系，还可以描述两个以上实体间的联系。比如，课程、教师、参考书 3 个实体，并有以下语义：每门课程由多名教师讲授，每门课程可以有多本参考书，一名教师讲授课程时可以使用多本参考书，一本参考书可以供多名教师使用，将联系名命名为讲授，如图 1-7a 所示。又比如，供应商、项目和零件 3 个实体，并有以下语义：每个供应商可以参与多个项目并供应多个零件，每个项目可以由多个供应商承包并供应多个零件，每个零件可以从多个供应商供应用于多个项目，将联系名命名为供应，如图 1-7b 所示。

同一个实体集内的各实体之间也可以存在一对一、一对多、多对多的联系。例如，多名教师中只有一位为系主任，同时系主任也是教师，如图 1-8 所示。

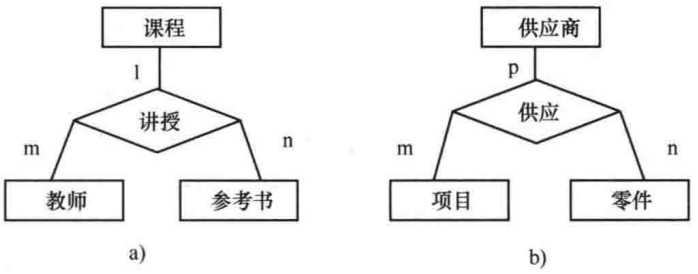

图 1-7　两个以上实体间的联系示例

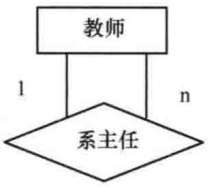

图 1-8　单个实体内部 1∶n 的联系

设计 E-R 图的过程包括以下 5 个步骤：

1）确定实体；

2）确定联系；

3）把实体和联系组合成 E-R 图；

4）确定实体和联系的属性；

5）确定实体的码。

在教师授课示例中，有教师、学生和课程3个实体，并有以下语义：每个教师可以教授多门课程，每门课程可以有多个学生选修，每个学生也可以选修多门课程。因此，教师与课程之间是一对多的联系（1:n），课程与学生之间是多对多的联系（m:n）。教师与课程间的关系是教课，学生与课程间的关系是学习，在菱形框内写明教课和学习，分别代表教课关系和学习关系，使用无向边把教师、课程和学生实体与教课、学习关系连接起来，并在无向边旁标上关系类型，如图1-9所示。

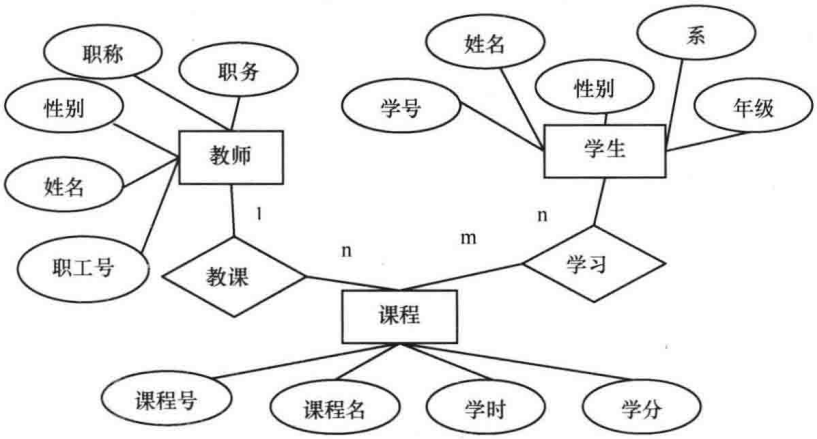

图1-9　E-R图

如图1-9所示，其中学号为学生实体的主码，职工号为教师实体的主码，课程号为课程实体的主码。

1.2.3　组织层数据模型

组织层数据模型是按计算机系统的观点对数据进行建模，主要包括层次模型、网状模型、关系模型、面向对象模型等，用于数据库管理系统的实现。其中，层次模型和网状模型又称为非关系模型。

1. 层次模型

层次模型用树形结构来表示各类实体以及实体间的联系，如图1-10所示。树形结构中有且只有一个根结点，其余为子孙结点，每个结点（除根结点）只能有一个父结点（也称双亲结点），但是可以有一个或多个子结点，无子结点的结点称作叶。

层次模型的主要优点是模型比较简单，对于实体间联系是固定的，提供了良好的完整性支持；主要缺点是很难表示现实中事物间非层次性的联系，查询子结点必须通过父结点，对插入和删除操作的限制较多。

2. 网状模型

网状模型是一个网状结构模型，是对层次模型的扩展，允许一个以上的结点无双亲，同时也允许一个结点有多个双亲，如图1-11所示。层次模型是网状模型中的一种简单的情况。

网状模型的主要优点是具有良好的性能，存取效率高，能够更加直接地描述现实世界；主要缺点是结构比较复杂，不利于最终用户使用。

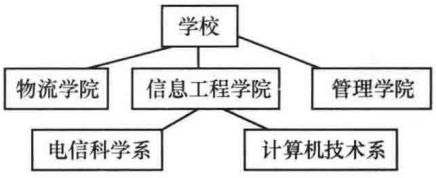

图 1-10　学校组织结构的层次模型

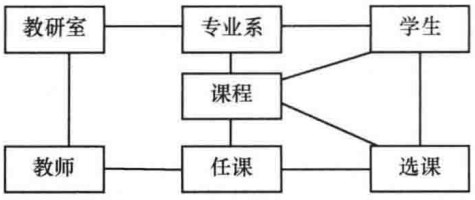

图 1-11　网状模型示例

3. 关系模型

关系模型是一种简单的二维表结构，每个二维表称作一个关系，关系中每一行数据称作一条记录，又称作元组，每一列数据称作属性，列标题称作属性名，见表 1-1。

表 1-1　课程表

课 程 号	课 程 名	类　别	学　时	学　分
001	计算机基础（一）	必修	48	3
003	编译原理	选修	32	2
103	数据库原理	必修	56	3.5

表 1-1 中的每一行称为一门课程记录，每一列称为课程的属性，课程号、课程名、类别、学时和学分为属性名。

关系模型的主要优点是建立在严格的数学概念基础上，其数据结构简单、清晰，用户容易掌握，存取路径对用户透明，具有更高的数据独立性和更好的安全保密性；主要缺点是查询效率不如非关系数据模型。

4. 面向对象模型

面向对象模型是用面向对象观点描述实体的逻辑组织、对象间限制及联系等模型。共享相同属性和方法集的所有对象构成一个对象类，称为类，而对象是类的实例。

1.3　关系模型

关系数据模型就是用关系表示现实世界中实体以及实体之间联系的数据模型。关系数据模型（下面简称关系模型）是目前最重要的数据模型之一。关系数据库就是采用关系模型作为数据的组织方式。关系模型的三要素包括数据结构、数据操作和完整性约束。下面从关系模型的三要素来介绍关系模型的特点。

1.3.1　关系模型的数据结构

数据结构是描述系统的静态特征，是刻画数据模型性质最重要的方面。把数据看成二维表中的元素，而这个二维表就是关系。用关系（表格数据）表示实体和实体之间联系的模型称为关系数据模型。关系结构的数学模型称作关系模型。在用户看来，一个关系模型的逻辑结构是一张二维表，见表 1-2，它由行和列组成。关系模型中的域、属性和关系等都属于数据结构。

表 1-2　关系模型示例

学　号	姓　名	民　族	性　别	出生日期
0711001	张然	汉	男	1988 年 8 月 8 日
0711002	许汐	汉	女	1988 年 7 月 4 日
0711003	李星星	满	男	1988 年 8 月 7 日

表 1-2 中的每一行称为一条学生记录，每一列称为属性，学号、姓名、民族、性别和出生日期分别是属性名。

以下介绍关系模型中的基本术语。

1）关系：关系就是二维表，见表 1-2。其满足以下性质：关系表中的每一列都是不可再分的基本属性；表中的行、列次序并不重要；表中的各属性不能重名。

2）元组：表中的每一行称作一个元组，它相当于一个记录值。例如，表 1-2 所示的学生关系中的元祖有：

（0711001，张然，汉，男，1988 年 8 月 8 日）

（0711002，许汐，汉，男，1988 年 7 月 4 日）

（0711003，李星星，满，男，1988 年 8 月 7 日）

3）属性：表中的每一列是一个属性值集，列可以命名，称为属性名。二维表中对应某一列的值称为属性值，列的个数称为关系的元数。如果一个二维表有 n 列，则称其为 n 元关系。表 1-2 所示的学生关系有学号、姓名、民族、性别、出生日期 5 个属性，是一个五元关系。

在关系模型中有两套标准术语，一套用的是表、行、列，而另外一套用关系（对应表）、元组（对应行）、属性（对应列）。

4）值域：属性的取值范围。例如，“性别”的值域是｛男，女｝。

5）候选键：又称候选码。如果一个属性或属性集的值能够唯一标识一个关系的元组而又不包含多余的属性，则称该属性或属性集为候选键。

6）主键：又称主码，是表中的属性或属性组，用于唯一地确定一个元组（从候选码中选择一个作为主码）。若某个关系的主键由多个属性共同组成，则用括号将这些属性括起来。例如，学生关系中，学号就是主码，因为学号的一个取值可以唯一确定一个学生；而成绩表中的主码就是由学号和课程号共同组成的，因为一个学生有多门课程的成绩，一门课程有多个学生参加考试，所以只有学号和课程号组合起来才能共同确定一行记录，称由多个属性共同组成的主码或主键为复合主键，则成绩关系中的主键是（学号，课程号）。

7）主属性：候选码所包含的属性。例如，在学生关系中学号为主属性。

8）非主属性：主属性之外的其他属性。例如，在学生关系中姓名、民族、性别、出生日期都是非主属性。

9）关系模式：对关系的描述。

关系模式可以形式化地表示为 R（U，D，DOM，F）。其中，R 为关系名，U 为组成该关系的属性名集合，D 为属性组 U 中属性所来自的域，DOM 为属性向域的映像集合，F 为属性间的数据依赖关系集合。

注：域名及属性向域的映像常常直接说明为属性的类型、长度。

关系模式通常可以简记为 R（U）或 R（A1，A2，…，An）。其中，R 为关系名，A1，A2，…，An 为 属性名。

也可以表示为关系名（属性 1，属性 2，…，属性 n）。

在关系中，主属性用下划线标识。

如表 1-1 所示，对应的关系为课程（课程号，课程名，类别，学时，学分）。

如表 1-2 所示，对应的关系为学生（学号，姓名，民族，性别，出生日期）。

10）外码：又称作外键。如果模式 R 中的某属性是其他模式的主码，那么该属性集为模式 R 的外码。

11）分量：元组中的一个属性值。

在关系模型中，实体以及实体间的联系都是用关系（二维表）来表示的，即用表格结构表达实体集，用外码表达实体间的联系。关系模型是由若干个关系模式组成的集合。关系模型是型，关系是值。关系的每一个分量必须是一个不可分的数据项，即不允许表中还有表。关系中不允许出现相同的元组。

1.3.2 关系模型的数据操作

数据操作是描述系统的动态特征，是指对数据库中各种对象的实例执行操作的集合。具体来说，包括操作和有关操作规则。数据操作是集合操作，操作对象和操作结果都是关系，即若干元组的集合。数据库中主要有两大类操作：检索和修改，修改包括插入、删除和更新。

1.3.3 关系模型的完整性约束

数据完整性是指数据库中存储的数据是有意义的或正确的。数据完整性约束是一组完整性规则的集合。为维护数据与现实世界的一致性，在关系模型中的完整性约束有三类：实体完整性约束、参照完整性约束和用户自定义完整性约束。在关系模型中，创建的关系必须满足实体完整性和参照完整性约束。

1. 实体完整性约束

实体完整性规定基本关系中所有主键的属性不能取空值。

实体完整性指的是关系数据库中的所有表都必须有主键，而且表中不允许存在无主键值的记录和主键值相同的记录。实体完整性通过主键实现。

例如，一个表对应存放一个实体，表中的每一行代表实体中的一个实例。如学生表用来存放学生实体，该表中的一行代表一个学生，一个学生有唯一的一个学号。如果主键值学号为空值，表明该学生的学号不能确定，也就是说没有唯一确定的学号，这只能说明他/她不是学生，因此就不应该存放在学生表中。

2. 参照完整性约束

参照完整性有时也称为引用完整性，要求引用存在的实体，不引用不存在的实体，主要考虑不同关系间或同一关系内不同元组之间的制约。现实世界中的数据之间存在某种联系，一些数据的取值可能需要参照另一些数据的取值范围，这种数据之间的参照关系就称为参照完整性。

需要全本请访问 www.ertongbook.com