



普通高等教育“十二五”规划教材

SQL Server 2012 数据库应用与实训

李萍 黄可望 黄能耿 编著



机械工业出版社
CHINA MACHINE PRESS



配电子课件

普通高等教育“十二五”规划教材

SQL Server 2012 数据库应用与实训

李 萍 黄可望 黄能耿 编著



机械工业出版社

本书根据当前市场对数据库人才的需求,以培养应用型和创新型数据库技术人才为目标,以 SQL Server 2012 为系统平台,重点介绍数据库的基本概念,数据库、数据表和各种约束的创建,数据操纵与数据查询,视图、函数、存储过程和触发器等数据库编程,以及数据库安全和日常维护管理等内容。

本书引入“学生成绩管理系统”“在线电子商店系统”和“图书借阅管理系统”3个案例,分别从理论、实训和综合案例的角度介绍数据库结构的设计思想,并将数据库原理融入到实际工程案例中,由浅入深,由易到难,循序渐进,强调实践性,突出实用性。在每章的实训部分,紧扣理论知识点,采用自主研发的 Jitor 实训指导软件,指导读者一步一步地进行操作,并及时检查操作完成情况,创新性强且特色鲜明。

本书内容广泛、充实,实用性强,既可作为应用型本科及高职高专院校的数据库课程教材,又可作为数据库应用开发人员的参考资料或培训教材。

为方便教学,本书配备电子课件等教学资源。凡选用本书作为教材的教师均可登录机械工业出版社教育服务网 www.cmpedu.com 免费下载。如有问题请致信 cmpgaozhi@sina.com,或致电 010-88379375 联系营销人员。

图书在版编目(CIP)数据

SQL Server 2012 数据库应用与实训 / 李萍, 黄可望,
黄能耿编著. —北京: 机械工业出版社, 2015. 7

普通高等教育“十二五”规划教材

ISBN 978-7-111-50508-2

I. ①S… II. ①李… ②黄… ③黄… III. ①关系数
据库系统—高等学校—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2015)第 129149 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: 刘子峰 责任编辑: 刘子峰 陈瑞文

责任校对: 张 征 封面设计: 陈 沛

责任印制: 刘 岚

北京富生印刷厂印刷

2015 年 7 月第 1 版·第 1 次印刷

184mm×260mm·18.75 印张·429 千字

0 001—3 000 册

标准书号: ISBN 978-7-111-50508-2

定价: 39.00 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

封面无防伪标均为盗版

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

前 言

“数据库原理与应用”是计算机软件技术、网络技术、物联网应用技术等专业的一门专业主干课程。通过本课程的学习，学生能够了解数据库设计的过程，掌握利用 SQL Sever 数据库管理系统实施数据库、进行数据操纵与数据查询、编写数据库对象、维护和管理数据库等技能。随着当前数据库技术日新月异的发展以及数据库应用在日常生活中的不断普及，作为现代大学生，特别是计算机专业的学生，学习和掌握数据库知识是非常必要的。

本书根据高等职业教育的教学特点，结合作者多年教学改革和应用实践经验编写而成。全书采用“项目导向、任务驱动”的组织模式，将“学生成绩管理系统”的实现与数据库应用技术的教学实施结合在一起。基于数据库系统的设计与实施过程划分为本书前 7 章的内容，分别为数据库基础、数据库和表的创建与维护、数据操纵、数据查询、数据库编程、数据库安全管理、数据库维护。每章结束附有小结、习题和实训项目，供读者及时消化章节内容，以更加深入地进行学习。

本书在各章实训部分和最后一章的数据库开发案例中还分别引入了“在线电子商店系统”和“图书借阅管理系统”，将数据库原理与应用技术融入到实际工程案例中。其中，实训部分紧扣理论知识点，采用自主研发的 Jitor 实训指导软件，指导读者一步一步地进行操作，并及时检查操作完成情况。综合案例部分让读者再次系统地体验了一个工程案例从数据库规划设计到数据库运行维护的全过程。

本书遵循学习者的认知和技能形成规律，使用通俗易懂的语言，采用图示法、类比法等多种适合学习者的讲解形式，由浅入深、循序渐进地介绍各章节内容，尽可能将知识融于形象的案例中，使读者易于学习和掌握，逐步培养读者的学习兴趣。

本书建议讲解学时为 64~80 学时，64 学时的学时分配见下表。

章节名	参考学时
第 1 章 数据库基础	8
第 2 章 数据库和表的创建与维护	10
第 3 章 数据操纵	6
第 4 章 数据查询	10
第 5 章 数据库编程	14
第 6 章 数据库安全管理	6
第 7 章 数据库维护	6
第 8 章 数据库开发案例——图书借阅管理系统	4

本书由无锡职业技术学院的李萍、黄可望、黄能耿编著。其中，整体设计由李萍完

成,第1、2、6、7、8章由李萍编写,第3、4、5章由黄可望编写,各章的实训内容与Jitor实训指导软件由黄能耿编写。全书由李萍统稿,由无锡职业技术学院的刘德强副教授主审。

本书在编写过程中得到了编者所在院系领导和同事的帮助和大力支持,刘培林、史荧中、王想实、汪菊琴、周薇老师提供了大量的资料并进行了教学模式与教学方法的探究,在此对他们的工作深表感谢。同时,在本书的编写过程中,参考了目前国内外优秀的数据库原理与应用方面的书籍、资料,在此谨向有关作者表示感谢。

限于编者水平,书中错误与不足之处在所难免,敬请广大读者批评指正。

编 者

目 录

前 言

第1章 数据库基础.....	1	2.1.4 创建数据库.....	40
1.1 数据库与数据库系统	1	2.1.5 修改数据库.....	43
1.1.1 数据库	1	2.1.6 删除数据库.....	44
1.1.2 数据库管理系统	1	2.2 创建和管理数据表.....	45
1.1.3 数据库系统	2	2.2.1 数据表的设计	45
1.2 数据模型	3	2.2.2 数据类型	45
1.2.1 概念数据模型	3	2.2.3 创建数据表	47
1.2.2 结构化数据模型	5	2.2.4 数据完整性约束	49
1.2.3 关系数据库管理系统	6	2.2.5 修改数据表	54
1.3 关系规范化设计.....	10	2.2.6 删除数据表	56
1.3.1 关系中的异常.....	10	2.3 索引.....	56
1.3.2 函数依赖.....	12	2.3.1 索引的概念.....	56
1.3.3 范式理论.....	13	2.3.2 索引的类型.....	57
1.3.4 注意事项.....	15	2.3.3 创建索引.....	58
1.4 数据库设计.....	15	2.3.4 管理索引.....	59
1.4.1 数据库设计概述.....	15	本章小结	60
1.4.2 数据结构设计.....	17	习题2	61
1.4.3 数据结构设计工具软件.....	21	实训2 在线电子商店数据库	
1.4.4 数据结构设计实例.....	22	和表的创建	62
1.5 SQL Server 2012 入门.....	28	第3章 数据操纵	64
1.5.1 SQL Server 2012 系统简介.....	28	3.1 数据插入语句.....	64
1.5.2 SQL Server 2012 的安装.....	29	3.1.1 使用 Values 关键字的	
1.5.3 SQL Server 2012 的使用.....	32	Insert 语句	64
本章小结	33	3.1.2 使用 Select 子句的	
习题1	34	Insert 语句	65
实训1 在线电子商店的设计	34	3.1.3 数据插入中的完整性约束.....	66
第2章 数据库和表的创建与维护	38	3.2 数据更新语句.....	67
2.1 创建和管理数据库.....	38	3.2.1 数据更新基本语句	67
2.1.1 数据库常用对象.....	38	3.2.2 数据更新中的完整性约束.....	68
2.1.2 数据库的组成.....	39	3.3 数据删除语句.....	69
2.1.3 系统数据库.....	40	3.3.1 Delete 语句	69

3.3.2 Truncate 语句	69	5.2.3 注释	118
3.3.3 数据删除中的完整性约束	69	5.3 游标	119
本章小结	70	5.3.1 游标的概念	119
习题3	70	5.3.2 游标的基本操作	120
实训3 在线电子商店的数据初始化 ...	71	5.3.3 使用游标更新数据	123
第4章 数据查询	75	5.4 视图	124
4.1 数据查询语句的基本结构	75	5.4.1 视图的概念	124
4.2 简单查询	75	5.4.2 创建视图	125
4.2.1 Select...From 子句	75	5.4.3 使用视图查询数据	127
4.2.2 Where 子句	78	5.4.4 修改视图	128
4.2.3 Order By 子句	81	5.4.5 删除视图	128
4.2.4 Group By 子句	82	5.4.6 视图的应用	128
4.2.5 联合查询	85	5.5 函数	132
4.3 连接查询	86	5.5.1 标准函数 (内置函数)	132
4.3.1 交叉连接	86	5.5.2 自定义函数	134
4.3.2 内连接与等值连接	87	5.5.3 自定义函数的维护	139
4.3.3 外连接	90	5.6 事务和锁	139
4.3.4 自连接	94	5.6.1 事务	139
4.4 子查询	95	5.6.2 锁机制	143
4.4.1 嵌套子查询	95	5.6.3 丢失更新	148
4.4.2 相关子查询	100	5.7 存储过程	152
4.5 基于数据查询的数据操纵	103	5.7.1 存储过程的概念	152
4.5.1 联合使用 Select 和 Insert 语句	103	5.7.2 系统存储过程	153
4.5.2 联合使用 Select 和 Update 语句	104	5.7.3 用户自定义存储过程	154
4.5.3 联合使用 Select 和 Delete 语句	105	5.7.4 管理存储过程	159
本章小结	105	5.8 触发器	160
习题4	106	5.8.1 触发器的基本概念	161
实训4 在线电子商店数据库的查询 ...	107	5.8.2 创建 DML 触发器	161
第5章 数据库编程	110	5.8.3 创建 DDL 触发器	168
5.1 变量与流程控制语句	110	5.8.4 管理触发器	168
5.1.1 变量	110	本章小结	169
5.1.2 流程控制语句	113	习题5	171
5.2 脚本文件	117	实训5 在线电子商店数据库的编程 ...	172
5.2.1 SQL 脚本文件的概念	117	第6章 数据库安全管理	179
5.2.2 批处理	118	6.1 安全体系结构	179
		6.1.1 客户机操作系统的安全管理 ...	179
		6.1.2 SQL Server 服务器的 安全管理	180

6.1.3 数据库的安全管理	180	7.4.3 约束的创建与维护	230
6.1.4 数据库对象的安全管理	180	7.4.4 索引的创建与维护	233
6.2 客户机安全认证	181	本章小结	233
6.2.1 创建 Windows 用户	181	习题 7	234
6.2.2 客户机身份验证	182	实训 7 在线电子商店数据库的维护 ..	235
6.3 服务器安全认证	183	第 8 章 数据库开发案例——图书借阅	
6.3.1 建立服务器登录账号	184	管理系统	236
6.3.2 设置账号的服务器角色	189	8.1 开发环境与开发工具	236
6.3.3 管理与使用登录账号	190	8.2 系统需求分析	237
6.4 数据库安全认证	192	8.2.1 总体需求	237
6.4.1 创建数据库用户	192	8.2.2 业务分析	238
6.4.2 维护数据库用户	193	8.2.3 非功能性需求	238
6.5 数据库对象安全认证	195	8.2.4 功能分析	238
6.5.1 角色管理	196	8.3 数据结构设计	239
6.5.2 权限管理	198	8.3.1 结构分析	239
本章小结	204	8.3.2 物理数据模型设计	239
习题 6	205	8.3.3 数据字典	240
实训 6 在线电子商店数据库的		8.4 数据库实施	242
安全管理	205	8.4.1 生成数据库	242
第 7 章 数据库维护	207	8.4.2 数据初始化	245
7.1 数据库的分离与附加	207	8.4.3 数据查询核对	248
7.1.1 数据库的分离	207	8.5 数据库应用程序编制	249
7.1.2 数据库的附加	208	8.5.1 ADO.NET 数据库编程接口 ..	249
7.2 备份与恢复	209	8.5.2 数据库连接通用模块	255
7.2.1 常见数据库故障	209	8.5.3 C/S 结构程序设计	259
7.2.2 数据库备份设备	209	8.5.4 B/S 结构程序设计	273
7.2.3 备份方式	210	8.6 数据库开发案例小结	278
7.2.4 恢复模式	210	附录	279
7.2.5 备份与恢复方案	212	附录 A 学生成绩管理系统	
7.2.6 数据库备份与恢复	213	数据表结构	279
7.2.7 建立自动备份的维护计划 ..	217	附录 B 在线电子商店数据表结构	281
7.3 数据转换	220	附录 C SQL 常用函数	283
7.3.1 数据转换的基本概念	220	附录 D Jitor 实训指导软件	
7.3.2 数据的导出	221	使用说明	287
7.3.3 数据的导入	223	参考文献	292
7.4 数据库的日常维护操作	224		
7.4.1 数据库的创建与维护	225		
7.4.2 数据表的创建与维护	227		

第1章 数据库基础

近年来,数据库技术在多个领域都得到了广泛应用,无论是应用的数量还是重要性都在不断增长。医院、学校、企业、图书馆、商场、超市、网站等都使用数据库来存储、操作和检索数据。本章从数据库的基本知识起步,介绍 SQL Server 2012 数据库管理系统的安装和简单使用;从数据库设计的模型、基本原则和基本步骤出发,以学生成绩管理系统为开发项目,详细讲解数据库的设计过程,为后续数据库各类应用的学习奠定基础。

1.1 数据库与数据库系统

1.1.1 数据库

数据库(Database, DB),简单地说,就是存储数据的仓库。它是储存在计算机内有组织、可共享的逻辑数据集合,也是现实世界中互相关联的大量逻辑数据及数据间关系的集合。数据库中的数据按一定的数据模型进行组织和描述,并储存在计算机的硬盘中,具有较小的冗余度、较高的数据独立性和易扩展性,并可为各种用户共享。数据库中不仅存放数据,而且存放数据之间的关系或联系。

1.1.2 数据库管理系统

数据库管理系统(Database Management System, DBMS)是位于用户与操作系统之间用于管理数据库的系统软件,具备如下功能:

(1) 数据定义功能

DBMS 提供数据定义语言(Data Definition Language, DDL),用 DDL 可以定义数据库中的数据对象。例如,定义数据库、数据表、视图和索引等,还可以定义数据的完整性与安全性等约束条件。

(2) 数据操纵功能

DBMS 提供数据操纵语言(Data Manipulation Language, DML),用 DML 可以操纵数据库中的数据。例如,用 SQL 语句实现对数据库中数据的查询、插入、修改与删除等操作。

(3) 数据库管理功能

数据库管理功能是 DBMS 的核心功能,由控制程序实现。控制程序的主要功能有:对数据库的完整性约束条件的检查和执行、安全性检查和并发性控制、数据库维护。所有数据库的操作都是在控制程序的统一管理和统一控制下进行的,以确保数据库的安全性、完整性、并发性,并能在发生故障时恢复系统。

(4) 数据库维护功能

数据库维护功能主要包括数据库中数据的输入、转换、转储、恢复、性能监视、分析

等，由 DBMS 实用程序提供给数据库管理员实现上述功能。

1.1.3 数据库系统

数据库系统（Database System，DBS）由计算机软、硬件系统、数据库、数据库管理系统、数据库应用程序、数据库管理员与用户组成，如图 1-1 所示。

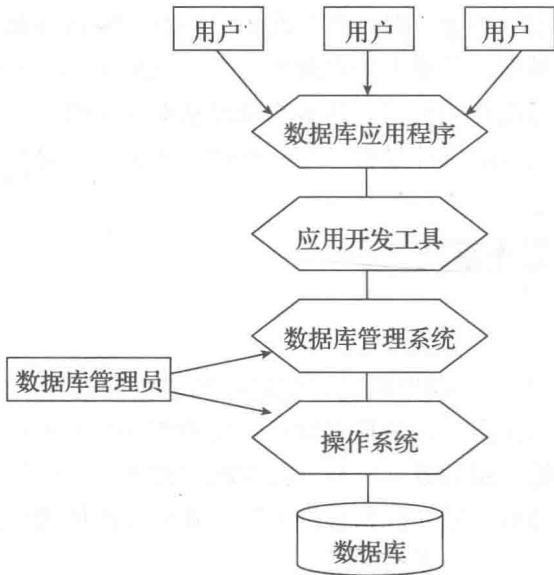


图 1-1 数据库系统的构成

- 1. 支持数据库运行的软、硬件环境
每种数据库管理系统都有其要求的软、硬件环境。一般对硬件要说明所需的基本配置，对软件则要说明其适用于哪些底层软件、与哪些软件兼容等。
- 2. 数据库管理系统
目前，市场占有率较高的数据库管理系统主要有高端用户使用的甲骨文公司的 Oracle、中低端用户使用的微软公司的 SQL Server 以及开源数据库 MySQL。
- 3. 数据库应用程序
数据库应用程序是一个允许用户插入、修改、删除并显示数据库中数据的计算机程序，由程序员用某种程序设计语言编写。数据库应用程序可通过 DBMS 对数据库中的数据进行操作。常用的数据库应用程序开发工具有 Visual C#、Visual Basic、Java、C++、Delphi 等。
- 4. 系统人员与用户
 - (1) 数据库管理员
数据库管理员（Database Administrator，DBA）是管理、维护数据库系统的人员，其具体职责如下：决定数据库中的信息内容和结构；决定数据库的存储结构和存取策略；定义数据

的安全性要求与完整性约束条件；监督和控制数据库的使用与运行；数据库的改进与重组；数据库的备份与灾难恢复等。

(2) 应用程序员

应用程序员根据系统分析员对系统的需求分析与功能分析的结果，设计和编写数据库应用程序中各功能模块的界面与程序代码，并测试程序。

(3) 用户

用户指最终使用数据库应用程序的人员。

1.2 数据模型

如何将现实世界中要处理事物转换成能用计算机处理的数据？通常的解决方法是先对现实世界事物进行分析，建立某种能描述客观事物的概念数据模型，然后根据结构化数据模型建立数据库结构，并编写数据库应用程序。

1.2.1 概念数据模型

1. 概念数据模型中的常用术语

(1) 实体 (Entity)

现实世界中客观存在并可相互区别的事物称为实体，如学生“李丽”“肖明”“王明”等都是实体。

(2) 实体集 (Entity Set)

同类型实体集合称为实体集，如全体学生就是一个实体集。

(3) 属性 (Attribute)

属性是对实体特性的描述，如描述学生实体的属性有学号、姓名、年龄、性别等；属性值是属性的取值，如“100231，李丽，20，女”为李丽这个学生的属性值。

(4) 域 (Domain)

域是属性取值的范围，如学生学号域为“1000”~“9999”，性别域为“男”和“女”。

(5) 键 (Key)

能唯一标识实体集中各个实体的一个属性或一组属性值称为实体的键。例如，学号属性能唯一标识学生实体集中不同的学生，因此学号是学生实体的键。

(6) 联系 (Relationship)

在现实世界中，事物内部及事物之间是普遍存在联系的。这些联系在信息世界中表现为实体型内部各属性之间的联系以及实体型之间的联系。两个实体型之间的联系可以分为以下3种：

1) 一对一联系。若实体集A中每一个实体只能与实体集B中的一个实体有联系，则称A与B存在一对一的联系，记为1:1。例如，一个班只有一个班长，一个班长只能管理一个班，所以班级实体集A与班长实体集B为一对一的联系。

2) 一对多联系。若实体集 A 中每个实体与实体集 B 中多个实体有联系, 而 B 中每个实体至多与实体集 A 中的一个实体有联系, 则称实体集 A 与 B 存在一对多的联系, 记为 $1:n$ 。例如, 系对班级、班级对学生都是一对多的联系。

3) 多对多联系。若实体集 A 中的每个实体与实体集 B 中的任意多个实体有联系, 而实体集 B 中的每个实体又与实体集 A 中的任意多个实体有联系, 则称实体集 A 与 B 存在多对多的联系, 记为 $m:n$ 。例如, 教师与学生、学生与课程都是多对多的联系。

联系可以像实体一样拥有属性。例如, 学生与课程之间有一个“选修”的联系, “选修”联系本身又可有属性“成绩”。

2. 概念数据模型的表示方法

概念数据模型的表示方法最常用的是实体—联系方法 (Entity-Relationship)。该方法是指在数据库设计之前, 先用实体—联系示意图表示概念世界中的实体联系, 再将该图转换为数据库所支持的数据模型, 这种数据库设计方法与程序设计中的画框图相类似。采用这种方法建立的模型作为不依托 DBMS 的一种概念数据模型, 以比较自然的方式模拟现实世界。

在实体—联系模型中, 基本建模元素是实体、联系和属性。在数据库逻辑设计中, 用于表示概念世界中实体及其联系的结构图称为实体—联系模型图, 简称 E-R 图。E-R 图是直观表示概念模型的有效工具, 在软件工程和数据库设计过程中使用很普遍, 是描述数据模型的有效方法。在 E-R 图中, 使用的符号如下:

- 1) 用矩形表示实体。
- 2) 用椭圆形表示属性, 并用无向线标出实体与属性的关系。
- 3) 用菱形表示实体间的联系, 并用无向线标出实体间的联系。

【例 1-1】用 E-R 图表示学生选修课程成绩管理的概念模型 (学生成绩管理中涉及的实体及联系较多, 本例为方便起见, 只考虑学生和课程实体及其联系)。

学生实体的属性有学号、姓名、年龄、性别等; 课程实体的属性有课程号、课程名、课时等。两者之间具有选修的联系, 一名学生可以选修多门课程, 一门课程可以由多名学生选修, 所以学生与课程是多对多的联系。“选修成绩”联系也具有属性, 分别为学号、课程号、成绩等。据此可画出学生选修课程成绩管理的 E-R 图, 如图 1-2 所示。

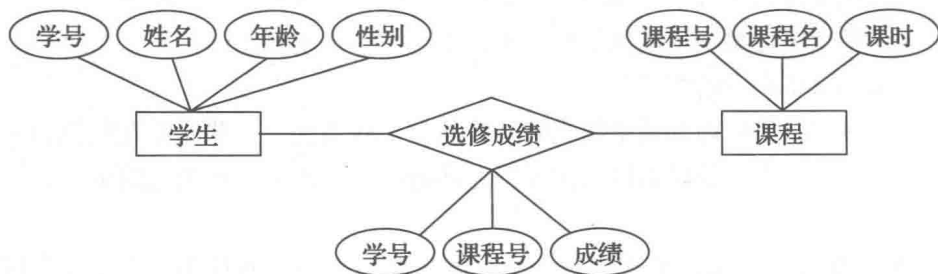


图 1-2 学生选修课程成绩管理的 E-R 图

1.2.2 结构化数据模型

1. 数据模型定义的三要素

结构化数据模型定义的三要素为数据结构、数据操作及完整性数据约束。

1) 数据结构是对实体类型和实体间联系的表达和实现，如层次数据结构、网状数据结构和关系数据结构。

2) 数据操作是对数据库中各种对象进行检索和更新（包括插入、删除、修改）等操作，数据模型必须定义这些操作的确切含义、操作符号、操作规则（如优先级）以及实现操作的语言。

3) 完整性数据约束是对数据对象及其联系的约束规定，以保证数据库中数据的正确性、有效性和安全性。

2. 数据模型的分类

(1) 层次模型

层次模型是用树形层次结构表示实体之间联系，如图 1-3 所示。现实世界中许多实体之间的联系本来就呈现为一种自然的层次关系，如产品组成、企业行政机构、家族关系等。因此，层次模型可以自然地表达自然界数据间具有层次规律的分类关系、概括关系、部分关系等，但在结构上有一定的局限性，且缺乏操作代数性质。

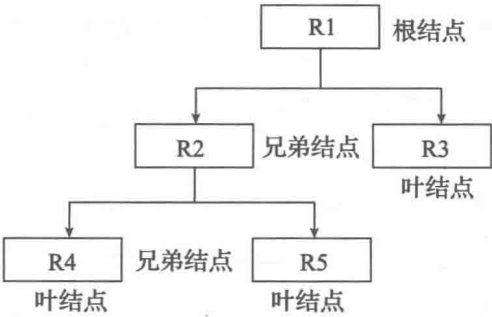


图 1-3 层次模型结构示意图

(2) 网状模型

用有向图结构表示实体间联系的数据模型称为网状模型，如图 1-4 所示。有向图结构中的结点代表实体记录类型，连线表示结点间的关系，这一关系必须是一对多的关系。网状模型必须满足以下两个条件：其一是允许一个以上的结点没有双亲结点；其二是至少有一个结点可以有多个的双亲结点。

网状数据模型能更为直接地描述现实世界，存取效率较高，但结构比较复杂，多对多的关系必须分解为多个一对多的关系才能进行处理，不利于用户掌握。在网状数据模型中，由于记录之间的联系是通过存取路径实现的，因

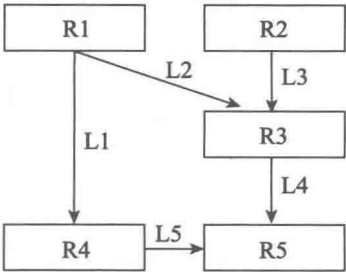


图 1-4 网状模型结构示意图

此用户必须了解系统存储结构的细节。

(3) 关系模型

正是由于层次模型和网状模型都无法直接表示概念世界中多对多的联系,因此在概念世界向数据世界转换的过程中必须要改进数据模型的结构。1970 年 IBM 公司 San Jose 研究室的研究员 E. F. Codd 首次提出了数据库系统的关系模型。

关系模型用二维数据表表示实体,用外键表示实体之间的联系。在关系模型中,每张二维表均称为一个关系,表中每一行称为记录,每一列称为数据项。关系模型是建立在严格的数学概念基础上的,无论是实体还是实体之间的联系都是用关系表示。关系模型的存取路径对用户透明,简化了程序员的工作和数据库的开发工作,因此已经成为目前主流的数据模型。

1.2.3 关系数据库管理系统

建立在关系模型基础上的数据库管理系统称为关系数据库管理系统 (Relational Database Management System, RDBMS)。Oracle、SQL Server、MySQL、Access 等数据库管理系统都是 RDBMS,其具有如下特点:

1) 容易理解。二维表结构是非常贴近逻辑世界的一个概念,关系模型相对于网状、层次等其他模型来说更容易理解。

2) 使用方便。通用的 SQL (结构化查询语言) 使得操作关系型数据库非常方便,程序员和数据管理员可以方便地在逻辑层面操作数据库,而完全不必理解其底层实现。

3) 易于维护。丰富的完整性 (实体完整性、参照完整性和用户定义的完整性) 大大降低了数据冗余和数据不一致的概率。

下面从关系模型的数据结构、关系操作和完整性数据约束 3 部分来介绍。

1. 数据结构

关系模型的数据结构是一张二维数据表,用二维表来表示实体,用外键表示实体之间的联系。每张二维表均称为一个关系,表中每一行称为元组,每一列称为属性 (数据项)。在数据库中,通常关系被称为数据表,元组被称为记录,属性则被称为字段。关系模型的数据结构如图 1-5 所示。

学生关系 (学生表)			
学号	姓名	年龄	性别
1001	Wang	20	M
1002	Wu	19	M
1003	Liu	21	F

图 1-5 关系模型的数据结构

通过关系名和属性名列表对关系进行描述,相当于二维表的表头部分。关系模型的表示形式如下:

关系名(属性名1,属性名2,...,属性名n)

关系满足以下6条性质:

- 1) 元组的次序是无关的。
- 2) 属性的次序是无关的。
- 3) 不能有完全相同的元组。
- 4) 不能有完全相同的属性名称。
- 5) 同一属性的值必须来自同一个域。
- 6) 属性是不可分的原子项。

在图1-2所示的E-R模型中,“学生”和“课程”两个实体集以及带属性的联系“选修成绩”的关系模型的表示形式如下:

学生(学号,姓名,年龄,性别)

课程(课程号,课程名,课时)

选修成绩(学年,学期,学号,课程号,成绩)

在关系数据库中还有一个重要的概念是“键”。键由一个或几个属性组成,在实际使用中,有以下几种键。

- 1) 候选键:若一个属性集能唯一标识元组,则该属性集称为候选键。
- 2) 主键:用户选作元组标识的一个候选键称为主键。
- 3) 外键:若关系中的属性或属性组不是本关系的主键,其值是引用另一个关系的主键,则称该属性或属性组为外键。

【例1-2】分析学生成绩管理系统中各个关系的候选键、主键和外键。

1) 学生关系的候选键是学号,如果学生不存在同名,则姓名也可以作为候选键。选择“学号”作为学生关系的主键。

2) 课程关系的候选键是课程号,如果课程不存在同名,则课程名也可以作为候选键。选择“课程号”作为课程关系的主键。

3) 选修成绩关系的候选键是“学号+课程号”的属性组合,这个候选键同时也是选修成绩关系的主键。选修成绩关系中的学号不是主键,但是它的取值必须引用学生关系的主键学号,所以选修成绩关系中的学号是外键。同理,选修成绩关系中的课程号也是外键,它的取值必须引用课程关系的主键课程号。

学生关系表、课程关系表、成绩关系表及其主、外键联系如图1-6所示。

2. 关系操作

关系操作是对关系型数据库中各种对象进行检索和更新(包括插入、删除、修改)等操作,关系模型使用关系数据语言实现数据操作。关系数据语言分为三类,如图1-7所示。

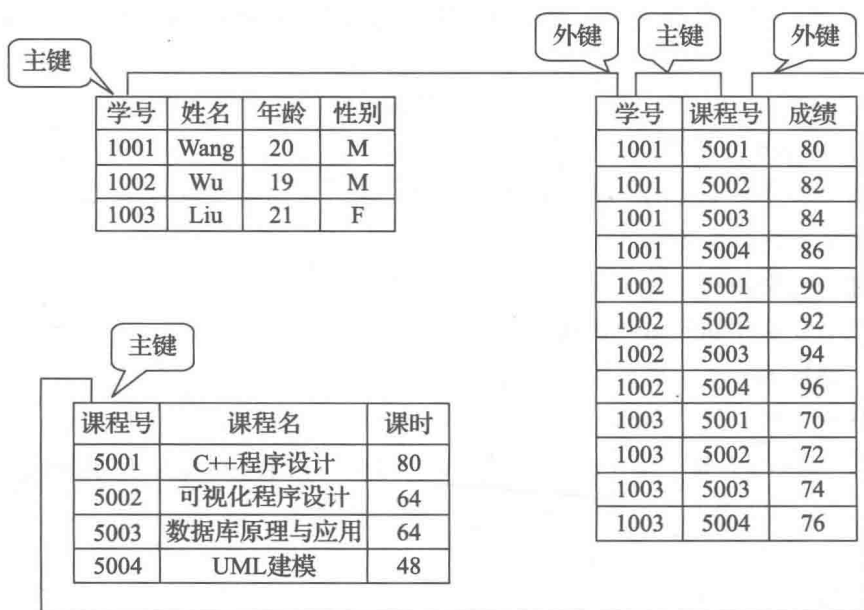


图 1-6 学生、课程、成绩关系表

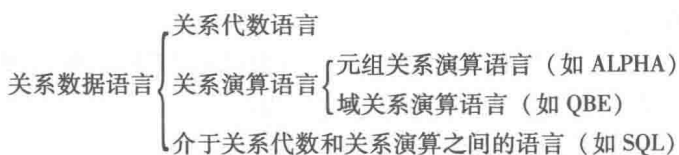


图 1-7 关系数据语言分类

关系代数、关系演算都是抽象的查询语言，它们与具体的 DBMS 中实现的实际语言并不完全一致，但它们能用作评估实际系统中查询语言能力的标准或基础。

基于关系集合运算的关系代数，能抽象而直观地表达对关系的各种操作，其中常用的关系操作有选择、投影、连接、除法、并、交、差等查询操作和添加、修改、删除等更新操作。其中，查询是最主要的组成部分。

SQL (Structured Query Language, 结构化查询语言) 最初于 20 世纪 70 年代由 IBM 公司的 Chamberlin 和 Boyce 提出，当时称为 SEQUEL (Structured English Query Language, 结构化英语查询语言)。由于 SQL 的成功，ANSI 在 1986 年将其作为标准，称为 SQL-86，随后 ISO 组织也采用它，称其为 SQL-87。SQL 经过多次修订，发展过程见表 1-1。

表 1-1 SQL 发展一览表

年份	标准	别名	说明
1986	SQL-86	SQL-87	ANSI SQL 的最初版本
1989	SQL-89	FIPS 127-1	少量修订
1992	SQL-92	SQL2	重要修订，是一个标志性的标准

(续)

年份	标准	别名	说明
1999	SQL: 1999	SQL3	增加了正则表达式、递归查询、触发器、过程控制流、非标量类型、面向对象特征
2003	SQL: 2003	SQL 2003	引入 XML 支持、标准化序列、自动生成 ID
2006	SQL: 2006	SQL 2006	提供了对 XML 更多的支持
2008	SQL: 2008	SQL 2008	增加 INSTEAD OF 触发器、增加 TRUNCATE 语句等

SQL 已经是一种非常成熟的语言，一般所说的支持 SQL 标准通常是指支持 SQL-92 或 SQL: 1999 标准。

SQL 是高级的非过程化编程语言，它允许用户在高层数据结构上工作，不要求用户指定对数据的存放方法，也不需要用户了解其具体的数据存放方式。而它的界面，能使具有与底层结构完全不同的数据库系统和不同的数据库之间，使用相同的 SQL 作为数据的输入与管理。它以记录项目〔records〕的合集 (set)〔项集, record set〕作为操纵对象，所有 SQL 语句接受项集作为输入，回送出的项集作为输出，这种项集特性允许一条 SQL 语句的输出作为另一条 SQL 语句的输入，所以 SQL 语句可以嵌套，这使它拥有极大的灵活性和强大的功能。在多数情况下，在其他编程语言中需要用一大段程序才可实现的一个单独事件，在 SQL 中只需要一个语句就可以被表达出来，这也意味着用 SQL 可以写出非常复杂的语句。

由于篇幅限制，关系代数、关系演算语言不再做详细介绍，而 SQL 则是本门课程的学习重点，将在后面的章节具体介绍。

3. 完整性数据约束

完整性数据约束是对数据对象及其联系的约束规定，以保证数据库中数据的正确性、有效性和安全性。

关系模型的完整性数据约束条件分为 3 类，即实体完整性约束、参照完整性约束和用户定义的完整性约束。

(1) 实体完整性约束

实体完整性约束要求每个关系都有一个主键，且主键对应的属性都不能取空值，通过主键来区分关系表中不同的记录 (元组)。

(2) 参照完整性约束

参照完整性规则：若属性 (或属性组) F 是基本关系 R 的外键，它与基本关系 S 的主键 K_s 相对应，则对于 R 中每个元组在 F 上的值必须为 K_s 值或空值，即关系 R 上的外键只能取关系 S 中的主键值或空值。

(3) 用户定义的完整性约束 (User-Defined Integrity)

用户定义的完整性约束是用户针对某一具体关系数据库制定的约束条件。

例如，用户对学生关系 S 中的性别可制定约束为：Sex 取 “M” 或 “F”。再如，大学生的年龄应定义为两位整数，如果认为这个范围太大，还可限制为 15 ~ 30 岁等。用户定