



全国高等职业教育示范专业规划教材

计算机专业

SQL Server 2008 数据库程序设计

SQL Server 2008 SHUJUKU CHENGXU SHEJI

高冬梅 主编

 机械工业出版社
CHINA MACHINE PRESS



赠 电子 课件

全国高等职业教育示范专业规划教材
计算机专业

SQL Server 2008 数据库程序设计

主 编 高冬梅

副主编 陈 娜 王亚军

参 编 李 云 杨天化 于志承



机械工业出版社

本书以 SQL Server 2008 为平台,系统地讲述数据库原理与 SQL Server 2008 的功能和应用。全书共分 14 章,主要包括:数据库基础知识,SQL Server 2008 概述,Transact-SQL 语言,数据库的创建和管理,表的创建和管理,数据查询,索引,视图,存储过程,用户定义函数,触发器,游标和事务,数据库的安全性管理以及 SQL Server 2008 与 C#.Net 综合开发实例。

本书适合作为高等院校、高职高专院校计算机及相关专业的“数据库”课程教材,同时也适合作为数据库开发人员的参考书、自学教材和社会相关领域的培训教材。

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材服务网 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 2008 数据库程序设计/高冬梅主编. —北京:机械工业出版社, 2009. 5

全国高等职业教育示范专业规划教材—计算机专业

ISBN 978-7-111-27018-8

I. S… II. 高… III. 关系数据库—数据库管理系统, SQL Server 2008—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(2009)第 066715 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:王玉鑫 刘子峰 责任编辑:刘子峰

版式设计:张世琴 责任校对:纪敬

封面设计:鞠杨 责任印制:李妍

北京铭成印刷有限公司印刷

2009 年 6 月第 1 版第 1 次印刷

184mm × 260mm · 16.5 印张 · 426 千字

0001—4000 册

标准书号:ISBN 978-7-111-27018-8

定价:28.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

销售服务热线电话:(010)68326294

购书热线电话:(010)88379639 88379641 88379643

编辑热线电话:(010)88379543

封面防伪标均为盗版

前言

随着数据库应用技术的发展,越来越多的软件开发人员和爱好者希望了解并掌握数据库的开发和管理方法。数据库的应用为高效、精确地处理各种数据创造了条件。数据库的建设规模、数据库信息量的大小和使用频率,已成为衡量一个国家信息化程度的重要标志。

Microsoft SQL Server 系列数据库软件以其优秀的性能和强大的功能一直以来得到广大使用者的喜爱,并广泛应用于各行各业。2008 年,Microsoft 公司推出了 SQL Server 2008,使它成为至今为止最强大、最全面的 SQL Server 版本数据库软件。在继承了以往版本的优秀特性的同时,SQL Server 2008 在多个方面进行了改进和优化,为用户提供了更加高效、智能的平台,同时与其他工具(如 Office 2007)进行集成,将诸多实用功能紧密结合,使其功能更强大、使用更方便、界面更友好。

本书根据高等职业教育的特点和要求,遵循“理论够用、强化动手”的原则。书中既有配合教师讲解的示例内容,也有学生课堂练习的操作训练,并且每章都配有习题和实训,以加深学生对基础知识的理解。学生通过教材所提供的实训任务和实训项目,能够顺利地进行数据库应用技术的实践与训练,尽快掌握数据库的基本概念与基本应用。

本书的第 1、2、11、13 章由高冬梅编写,第 3、4、9 章由王亚军编写,第 5、6、10 章由陈娜编写,第 7、8 章由李云编写,第 12 章由杨天化编写,第 14 章由于志承编写。全书由高冬梅统稿。

在本书的编写过程中,参考了大量的相关书籍和资料,在此对这些参考文献的作者表示感谢。同时,衷心感谢陈晓敏、米昶、田更、葛艳玲、王勇、乔吉连、江永春等在本书的编写过程中给予的支持与帮助。

由于编者水平有限,书中难免存在错误及不当之处,敬请广大专家和读者的批评指正,以便及时修订和补充。

编者

目 录

前言

第1章 数据库基础知识	1
1.1 数据库概述	1
1.1.1 信息、数据与数据处理	1
1.1.2 数据库、数据库管理系统与数据库系统	2
1.1.3 数据模型	2
1.2 关系数据库	4
1.2.1 关系术语	5
1.2.2 关系数据库的性质	5
1.2.3 关系规范化	6
1.2.4 专门的关系运算	7
1.2.5 数据完整性	8
1.3 关系数据库的设计	8
1.3.1 关系数据库的设计原则	8
1.3.2 关系数据库的设计简介	9
本章小结	9
习题	9
第2章 SQL Server 2008 概述	11
2.1 SQL Server 2008 简介	11
2.1.1 SQL Server 2008 基本概况	11
2.1.2 SQL Server 2008 新增功能	12
2.2 SQL Server 2008 的安装	13
2.2.1 SQL Server 2008 安装要求	14
2.2.2 SQL Server 2008 安装过程	15
2.3 SQL Server 2008 配置和管理	22
2.3.1 配置管理器	22
2.3.2 SQL Server Management Studio	23
2.3.3 注册服务器	24
本章小结	26
习题	26
本章实训	26
第3章 Transact-SQL 语言	27
3.1 SQL 语言概述	27
3.1.1 Transact-SQL 语言的概念	27

12	3.1.2 命名规则和注释	27
12	3.1.3 批处理	28
20	3.1.4 脚本	28
12	3.2 常量与变量	28
00	3.2.1 全局变量	29
00	3.2.2 局部变量	29
12	3.3 运算符	30
12	3.4 函数	33
50	3.4.1 系统函数	33
50	3.4.2 转换函数	34
50	3.4.3 字符串函数	35
80	3.4.4 数学函数	36
80	3.4.5 日期和时间函数	36
80	3.4.6 聚合函数	37
05	3.4.7 排名函数	37
05	3.4.8 行集函数	38
05	3.5 表达式	38
15	3.6 流程控制语句	38
05	3.6.1 BEGIN...END 语句	39
05	3.6.2 IF...ELSE 语句	39
05	3.6.3 CASE 语句	39
18	3.6.4 WHILE...CONTINUE...BREAK 语句	41
18	3.6.5 GOTO 语句	42
58	3.6.6 WAITFOR 语句	42
48	3.6.7 RETURN 语句	42
28	3.6.8 PRINT 语句	42
48	3.7 Transact-SQL 中的错误处理	43
88	3.7.1 使用系统变量 @@ERROR 处理错误	43
00	3.7.2 使用错误函数处理错误	44
10	本章小结	45
10	习题	45
50	本章实训	46
	第 4 章 数据库的创建和管理	47
80	4.1 数据库的存储结构	47
50	4.1.1 数据库文件	47
00	4.1.2 数据库文件组	48
00	4.2 系统数据库	48
00	4.3 创建数据库	49
00	4.3.1 在图形界面下创建数据库	49
50	4.3.2 使用 CREATE DATABASE 语句创建数据库	49

4.4 管理数据库	54
4.4.1 查看数据库	54
4.4.2 修改数据库	55
4.4.3 收缩数据库	57
4.4.4 删除数据库	60
4.4.5 分离和附加数据库	60
4.5 备份数据库	61
4.5.1 创建备份设备	61
4.5.2 备份操作	62
4.6 还原数据库	65
4.7 数据库快照	66
本章小结	68
习题	68
本章实训	68
第5章 表的创建和管理	70
5.1 表的基础知识	70
5.1.1 表的概念	70
5.1.2 数据类型	71
5.2 表结构的创建、修改和删除	76
5.2.1 表结构的创建	76
5.2.2 表结构的修改	79
5.2.3 表结构的删除	81
5.3 约束	81
5.3.1 PRIMARY KEY 约束	82
5.3.2 UNIQUE 约束	84
5.3.3 CHECK 约束	85
5.3.4 DEFAULT 约束	86
5.3.5 FOREIGN KEY 约束	88
5.3.6 禁用约束	90
5.4 添加、修改和删除表中的数据	91
5.4.1 使用图形界面添加、修改和删除表中的数据	91
5.4.2 使用 INSERT 语句添加数据	92
5.4.3 使用 UPDATE 语句更新数据	94
5.4.4 使用 DELETE 语句删除数据	96
5.4.5 使用 MERGE 语句插入、更新和删除数据	97
本章小结	99
习题	100
本章实训	100
第6章 数据查询	103
6.1 SELECT 语句	103

6.2	数据基本查询	103
6.2.1	选择列	104
6.2.2	选择行	105
6.2.3	数据的排序	106
6.2.4	使用 DISTINCT 和 TOP 关键字	107
6.2.5	在查询的基础上创建新表	108
6.3	数据的分组和汇总	108
6.4	子查询	114
6.5	联接查询	115
6.5.1	内部联接	116
6.5.2	外部联接	117
6.5.3	交叉联接	119
6.6	集合运算	120
6.6.1	使用 UNION 运算符组合结果集	121
6.6.2	使用 EXCEPT 和 INTERSECT 执行半联接操作	122
	本章小结	123
	习题	123
	本章实训	123
第7章	索引	125
7.1	索引概述	125
7.1.1	索引分类	125
7.1.2	设计索引	127
7.2	创建索引	128
7.2.1	在图形界面下创建索引	128
7.2.2	用 CREATE INDEX 语句创建索引	129
7.3	管理索引	131
7.3.1	查看索引	131
7.3.2	修改索引	132
7.3.3	删除索引	133
	本章小结	133
	习题	133
	本章实训	133
第8章	视图	134
8.1	视图概述	134
8.2	创建视图	135
8.2.1	在图形界面下创建视图	135
8.2.2	用 CREATE VIEW 语句创建视图	136
8.3	修改视图	139
8.4	通过视图查询数据	140
8.5	删除视图	140

8.6 通过视图修改数据	141
本章小结	142
习题	142
本章实训	142
第9章 存储过程	143
9.1 存储过程概述	143
9.2 存储过程的创建与执行	144
9.2.1 存储过程的创建	144
9.2.2 存储过程的执行	148
9.3 管理存储过程	152
9.3.1 查看存储过程	152
9.3.2 修改存储过程	152
9.3.3 重命名存储过程	153
9.3.4 删除存储过程	154
*9.3.5 创建和使用 CLR 存储过程	156
本章小结	156
习题	157
本章实训	157
第10章 用户定义函数	159
10.1 用户定义函数概述	159
10.2 创建用户定义函数	160
10.2.1 创建用户定义标量函数	160
10.2.2 创建用户定义表值函数	162
10.3 管理用户定义函数	165
10.3.1 查看用户定义函数	165
10.3.2 执行用户定义函数	165
10.3.3 修改用户定义函数	166
10.3.4 删除用户定义函数	167
本章小结	167
习题	167
本章实训	167
第11章 触发器	168
11.1 触发器的基本概念	168
11.2 创建 DML 触发器	169
11.2.1 在图形界面下创建 DML 触发器	170
11.2.2 使用 CREATE TRIGGER 命令创建 DML 触发器	170
11.3 创建 DDL 触发器	177
11.4 管理触发器	179
11.4.1 查看触发器	179
11.4.2 修改触发器	180

11.4.3	删除触发器	181
11.4.4	禁用和启用触发器	182
	本章小结	183
	习题	183
	本章实训	183
第 12 章 游标和事务		184
12.1	游标概述	184
12.1.1	游标的概念及特点	184
12.1.2	游标分类	184
12.2	游标的声明和应用	186
12.2.1	游标的声明	186
12.2.2	游标的操作	187
12.2.3	游标的应用	189
12.3	事务	191
12.3.1	事务的概念	191
12.3.2	事务的操作举例	193
	本章小结	196
	习题	197
	本章实训	197
第 13 章 数据库的安全性管理		198
13.1	数据库的安全性管理概述	198
13.2	登录管理	198
13.2.1	身份验证模式	199
13.2.2	创建 SQL Server 登录名	200
13.3	用户账户管理	202
13.3.1	系统用户	203
13.3.2	创建数据库用户	203
13.4	角色管理	204
13.4.1	服务器级别角色	205
13.4.2	数据库级别角色	208
13.4.3	应用程序角色	209
13.5	权限管理	210
13.5.1	权限的类型	210
13.5.2	权限的管理	210
13.6	架构	212
	本章小结	213
	习题	213
	本章实训	213
第 14 章 SQL Server 2008 与 C#.Net 综合开发实例		214
14.1	大型企业物流管理系统概述	214

14.1.1	系统开发背景	214
14.1.2	系统环境介绍	215
14.1.3	系统功能分析	215
14.1.4	系统模块设计	215
14.2	数据库设计	218
14.3	客户端设计	224
14.3.1	系统登录	224
14.3.2	人员管理	226
14.3.3	基础数据	228
14.3.4	业务流程	232
14.3.5	数据查询	236
14.3.6	日志管理	240
14.4	服务器设计	241
14.4.1	数据库连接	241
14.4.2	系统初始化	245
14.4.3	数据库保护	247
14.4.4	数据库备份与恢复	248
14.4.5	数据库发布	248
参考文献		252

第1章 数据库基础知识

计算机因其所具有的极快的运算速度、巨大的存储容量以及精确的算术和逻辑运算性能等特点,已被广泛应用于各行各业。当今社会的信息极大丰富,信息的采集和加工更是离不开不断更新、发展的计算机技术。数据库理论及其应用技术是计算机技术的一个重要分支,是信息处理的核心技术。它是在人们运用计算机进行数据处理过程中,产生的一门有关数据采集、整理、存储、分类、排序、检索、维护、加工、统计和传输等一系列操作过程的知识和技术。

数据库的应用,为高效、精确地处理数据创造了条件。数据库的建设规模、数据库信息量的大小和使用频率已成为衡量一个国家信息化程度的重要标志。

本章学习目标:

- 1) 了解数据库、数据库管理系统、数据库系统、关系数据库的概念。
- 2) 掌握关系规范化的使用。
- 3) 掌握数据模型的概念。
- 4) 熟悉数据完整性的概念与实现方法。
- 5) 熟悉关系数据库的设计原则和设计步骤。

1.1 数据库概述

1.1.1 信息、数据与数据处理

1. 信息

信息是现实世界中各种事物的存在方式、运动形态以及它们之间的相互联系等诸要素在人脑中的反映,通过人脑抽象后形成的概念。当今社会,信息已经覆盖了人们生产、生活的每一个角落。我们不妨以企业产供销管理来引入信息的概念。对于企业来说,要想生产的产品有市场,就必须了解客户的需求,掌握客户的资料;要保持源材料及时、充足的供给,就必须掌握有关供应商的资料。由此,对企业来说,客户、供应商资料和进货的采购订单、销售的销售订单及企业内部的库存、产品资料等,所有这些资料都是信息。

2. 数据

数据一般是指信息的一种符号化表示方法,就是说用一定的符号表示信息,而采用何种符号,完全是人为规定。所以数据的概念包括两个方面的含义:一是数据的内容是信息;二是数据的表现形式是符号。例如客户资料中的客户名称、地址、联系电话等就是数据。

3. 信息与数据的关系

数据是承载信息的物理符号,或称之为载体,而信息是数据的内涵。比如“169cm”,这是一个数据,可以说是一个毫无意义的数字。但如果这个数据用来表示一个人的身高,就有了实际的意义。

具体来说,信息和数据的关系主要表现在以下三个方面:

1) 信息是加工后的数据。信息是一种经过挑选、分析、综合的数据,它使用户可以更清楚地了解正在发生什么事。所以,数据是原材料,信息是产品,信息是数据的含义。

2) 数据和信息是相对的。一些数据对某些人来说信息,而对另外一些人而言则可能只是数据。例如,在运输管理中,运输单对司机来说是信息,这是因为司机可以从该运输单上知道什么时候要为什么客户运输什么物品。而对负责经营的管理者来说,运输单只是数据,因为从单张运输单中,他无法知道本月经营情况,并不能掌握现有可用的司机、运输工具等。

3) 信息是观念上的。因为信息是加工了的数据,所以采用什么模型(或公式)、多长的信息间隔时间来加工数据以获得信息,是受人对客观事物变化规律的认识制约,由人确定的。因此,信息是揭示数据内在的含义,是观念上的。

4. 数据处理

数据处理是指将数据转换成信息的过程。数据处理的主要内容是数据的存储、查询、修改、分类、排序等,以及支持决策功能。例如对客户资料数据进行检索、分类、排序等。

1.1.2 数据库、数据库管理系统与数据库系统

1. 数据库

数据库(DataBase,简称DB)是存储在计算机存储设备上,是结构化的相关数据的集合。更准确地说,是长期存在计算机内、有组织、可共享的数据集合。它不仅包括描述事物的数据本身,而且包括相关事物之间的联系。例如,老师每天上课时用的点名册就是一种最简单、最原始的数据库。数据库技术的主要目标是解决数据处理的非数值问题,例如仓库管理、档案管理、图书资料管理、学生成绩管理等。

2. 数据库管理系统

数据库管理系统(DataBase Management System,简称DBMS)可以对数据库的建立、使用和维护进行管理,是位于用户与操作系统之间的一层数据管理软件。DBMS能定义数据的存储结构,提供数据的操纵机制,维护数据库的安全性、完整性和可靠性。

3. 数据库系统

数据库系统是指引进数据库技术后的计算机系统,实现有组织、动态地存储大量相关数据,提供数据处理和信息资源共享的便利手段。数据库系统由五部分组成:硬件系统、数据库集合、数据库管理系统及相关软件、数据库管理员和用户。

1.1.3 数据模型

在数据库中,用数据模型这个工具来抽象、表示和处理现实世界中的数据和信息。通俗地讲,数据模型就是对现实世界的模拟。

概念数据模型与用户通常理解数据的方式很相近,而物理数据模型则描述了数据如何在计算机上存储的具体细节。物理数据模型提供的概念通常对计算机专家很有意义,但对于一般的最终用户而言则没什么用处。在这两种极端模型之间是一类表示(或实现)数据模型,这类数据模型是数据库管理系统中使用的可以为最终用户所理解的模型。它隐藏了数据存储的一些细节,但可以用一种直接的方式在计算机系统上实现。

1. 概念数据模型

这种模型使用了诸如实体、属性和关系等概念。实体表示数据库中描述的一个现实世界的对象或概念,可以是具体的人、事、物或抽象的概念,如一名员工或一个项目。属性是实体所

具有的某一特性,一个实体可以由若干个属性来刻画,表示进一步描述一个实体的重要性质,如员工的姓名或工资。实体联系或实体间联系是指组成实体的各属性之间的联系及不同实体集之间的联系;两个或多个实体间的关系表示的是两个或多个实体间的一个关联,例如,一名员工和一个项目间的工作关系(即该员工从事此项目的工作)。

两个实体间的联系可以归结为三种类型:

1) 一对一联系。如果实体集 A 与实体集 B 之间存在联系,并且对于实体集 A 中的任意一个实体,实体集 B 中至多只有一个实体与之对应;而对实体集 B 中的任意一个实体,在实体集 A 中也至多只有一个实体与之对应,则称实体集 A 到实体集 B 的联系是一对一的,记为 1:1。

例如,有班级和班长两个实体,并且有定义:一个班级只能有一名班长,一个学生也只能在一个班里当班长。那么,班级和班长之间的联系是一对一的,把这种联系命名为任职。

2) 一对多联系。如果实体集 A 与实体集 B 之间存在联系,并且对于实体集 A 中的一个实体,实体集 B 中可以有多个实体与之对应;而对实体集 B 中的任意一个实体,在实体集 A 中至多只有一个实体与之对应,则称实体集 A 到实体集 B 的联系是一对多的,记为 1:n。

例如,有班级和学生两个实体,并且有定义:一个班级可以有多名学生,但是一个学生只能在归属一个班级。那么,班级和学生之间的联系是一对多的,把这种联系命名为归属。

3) 多对多联系。如果实体集 A 与实体集 B 之间存在联系,并且对于实体集 A 中的一个实体,实体集 B 中可以有多个实体与之对应;而对实体集 B 中的一个实体,在实体集 A 中也可以有多个实体与之对应,则称实体集 A 到实体集 B 的联系是多对多的,记为 m:n。

例如,有学生和课程两个实体,并且有定义:一个学生可以同时选修多门课程,一门课程同时有若干个学生选修。那么,学生和课程之间的联系就是多对多的,把这种联系命名为选修。

同样两个实体,如果赋予不同的定义则有不同的设计结果。也就是说,实体之间的联系类型直接影响着目标数据库的设计结果和设计质量。

需要说明的是,还有一种对象数据模型,作为常用的高级概念模型,在软件工程领域经常使用。

2. 网状、层次和关系模型

这类模型包括网状模型和层次模型、关系模型。网状模型是将每项记录当成一个节点,节点与节点之间可以建立关联,形成一个复杂的网状结构。层次模型采用树状结构,依据数据的不同类型,将其分门别类地存储在不同的层次之下。网状模型和层次模型都要采用指针链表来实现数据的联系。关系模型使用二维矩阵(或称二维表格)来实现数据间的联系(或称关系)。关系模型将在 1.2 节详细叙述。

3. 物理数据模型

物理数据模型通过提供诸如记录格式、记录次序和存取路径等信息,来描述数据是如何作为文件在计算机中存储的。存取路径是一个可以使特定数据库记录的查找更为高效的结构。

任何一个数据库管理系统都是基于某种数据类型的。数据库管理系统所支持的常用的数据模型有:层次模型、网状模型、关系模型、面向对象模型。因此,使用支持某种特定数据模型的数据库管理系统开发出来的应用系统相应地称为层次数据库系统、网状数据库系统、关系数据库系统、面向对象数据库系统等。

1.2 关系数据库

首先请看下面三张表：表 1-1 为学生信息表、表 1-2 为课程信息表，表 1-3 为学生课程成绩表。

表 1-1 学生信息表 (tblStudents)

StudentID	Sname	Sex	Birthday	Department	Note
S200820001	李斌	男	1989-9-8	软件技术系	
S200820002	于莉	女	1990-1-8	电气工程系	
S200820003	李云华	男	1988-6-22	软件技术系	
S200820004	宋海波	男	1989-4-16	电气工程系	该生擅长书法

表 1-2 课程信息表 (tblCourses)

CourseID	Cname	Credit	CourseID	Cname	Credit
C8030120	C 语言程序设计	4	C8030248	数据库程序设计 SQL Server	4
C8030122	数据结构	4			

表 1-3 学生课程成绩表 (tblScore)

StudentID	CourseID	Grade	StudentID	CourseID	Grade
S200820001	C8030120	78.5	S200820002	C8030120	63.0
S200820001	C8030268	66.0	S200820003	C8030256	

根据这三张表，思考一下：如果你是教师或教学管理员，会采用什么方法来管理学生和课程？

你可能会准备一个登记本，在登记本上制作一些类似上面的表格，登记学生信息和教师所授课程信息以及学生所修课程信息，比如某个同学的学号、班级、选了什么课、教师授了什么课、课程学分、学生成绩等各种信息。对这些表格的管理则需要人工查阅和统计等，这些都是对表的操作。上节已经说过，这种表格实现了数据间的联系。

通过分析可以发现，表中存放了两类数据：现实世界实体（学生及课程等）本身的数据以及实体间的联系，这里的联系是通过表内列与列之间联系及不同的表中具有的相同列名来实现的。

例如表 1-1 的学号列 (StudentID) 和姓名列 (Sname) 之间的联系以及表 1-2 和表 1-3 通过课程号列 (CourseID) 实现的联系，这些“联系”都可以称为“关系”。上述每一张表内部及表间的联系都可称为“关系”，表名可称为“关系名”，一个关系就是现实世界中的一个实体，而关系的属性则代表了该实体的某种性质。

例如，实体“学生”可以用关系 Student (No, Name, Sex, Age, Major, ...) 来描述，其中属性 No、Name、Sex、Age、Major 分别代表了学号、姓名、性别、年龄、专业等“学生”固有的性质。

事实上，关系除了描述实体之外，还可以描述实体之间的联系。例如，男女之间“婚姻”

关系,就可以用这样一个关系来抽象: $Marry(Male_Name, Female_Name, \dots)$, 多个这种关系的组合就成为“关系数据库”, 称这种管理方法为基于关系模型的管理方法, 用术语来讲就是关系数据库管理方法。基于这种方法开发出来的应用系统称为关系数据库管理系统。

常见的关系数据库管理系统有 Sybase、Oracle、DB2、MS SQL Server、FoxPro、Access 等。

1.2.1 关系术语

在使用基于关系数据模型的关系数据库时, 需要理解以下几个主要概念。

- 1) 关系: 由行和列组成的二维表。例如, 上例的三张表就是二维表。
- 2) 元组: 表中的一行记录。例如, 表 1-1 中的一行: S200820004, 宋海波, 男, 1989-4-16, 电气工程系, 该生擅长书法。
- 3) 属性: 表中的一列字段, 例如, 表 1-1 中的 StudentID(学号)列。
- 4) 域(Domain): 属性(字段)的取值范围。例如, 表 1-1 中的 Sex(性别)列的取值范围为只能是“男”或“女”, 并且不能为空值。
- 5) 关系模式: 关系名(属性 1, 属性 2, …… , 属性 n)表名(字段 1, 字段 2, …… , 字段 n)。例如, 表 1-1 中的关系模式是: 关系名为学生信息表(学号, 姓名, 性别, 出生日期, 所在系, 备注); 表名为 tblStudents(StudentID, Sname, Sex, Birthday, Department, Note)。
- 6) 关键字(Key): 某个或某几个字段, 可以唯一确定一条记录, 就称为本关系的关键字。或者说, 对于某个关系, 能够唯一地确定其他属性(字段)值的属性(字段)组称为关键字。例如, 表 1-1 中的 StudentID(学号)字段(属性)为关键字, 它能唯一确定其他字段(如 Sname(姓名)字段等)的属性值。
- 7) 候选关键字: 对于某个关系, 若存在多个属性组都是关键字, 则称它们中的每一个都为该关系的候选关键字。例如, 表 1-2 课程信息表 tblCourses 中的课程号字段 CourseID 和课程名字段 Cname 都是关键字, 就称为该表的候选关键字。
- 8) 主关键字: 从候选关键字中任选一个作为现行关键字, 则该关键字称为主关键字, 例如, 表 1-2 中的课程号 CourseID 就是主关键字。
- 9) 主属性和非主属性: 凡能作候选关键字的属性称为主属性, 而不能作候选关键字的属性称为非主属性。例如, 表 1-2 中的课程号 CourseID 以及课程名 Cname 为主属性, 而其他字段则为非主属性。
- 10) 外部关键字: 不是本关系的关键字, 而是另一个关系的关键字。关系之间的连接常常依靠外部关键字, 例如表 1-2 学生课程成绩表 tblScore 中学号字段 StudentID 为外部关键字, 取值来自表 1-1 学生信息表; 选修课程号字段 CourseID 为外部关键字, 取值来自表 1-2 课程信息表。

1.2.2 关系数据库的性质

关系数据库具有以下几个性质:

- 1) 关系中不允许有完全相同的元组。
- 2) 在一个关系中元组的次序无关紧要。
- 3) 在一个关系中列的次序无关紧要。
- 4) 同一属性名下的各个属性值必须来自同一个域, 是同一类型的数据。
- 5) 在同一个关系中不能出现相同的属性名, 不同的属性可来自同一个域。

6) 关系必须规范化。

1.2.3 关系规范化

关系规范化的基本思想是逐步消除数据依赖关系中不合适的部分,使反映现实世界中同一实体的数据达到有效的分离。

1. 函数依赖

在关系中,属性(字段)间的相互依赖关系,称为函数依赖。例如,有学生表(学号,姓名,出生日期,性别,所在系,系主任,课程号,课程名,学分,成绩),其中,带下划线的属性为关键字。

1) 单值函数依赖:若属性组 X 的一个具体值对应 Y 的唯一值,则称 Y 单值函数依赖于 X (关键字)。如:

X : 学号—— Y : 学生的其他属性

X : 学号, 课程号—— Y : 学生的成绩

2) 完全函数依赖:若 Y 单值函数依赖于 X ,而 Y 不能单值函数依赖于 X 的任一真子集 X_1 ,则称 Y 完全函数依赖于 X 。如:

X : 学号—— Y : 学生的其他属性

3) 部分函数依赖:若 Y 单值函数依赖于 X ,而 Y 又能单值函数依赖于 X 的某一真子集 X_1 ,则称 Y 不完全函数依赖于 X ,或 Y 部分函数依赖于 X 。如:

X : 学号, 课程号—— Y : 学生的成绩, 学分

X 真子集: 课程号—— Y : 学生的学分

4) 传递函数依赖:若 Y 单值函数依赖于 X ,而 Z 又单值函数依赖于 Y ,则称 Z 传递函数依赖于 X 。如:

Y : 所在系—— X : 学号

Z : 系主任—— Y : 所在系

2. 关系规范化

创建关系型数据库必须遵循一定的原则规范。按照这些原则规范创建数据库,就称为关系规范化。这些规范按照条件的宽严程度不同分为五个等级,每个等级用一个范式表示。满足最低要求的关系规范化称为第一范式,再高一级的称为第二范式,依此类推,最高为第五范式。高级范式总是包含了低级范式的全部要求。

(1) 范式 建立关系时需要满足的约束条件划分成若干标准,这些标准称为范式(Normal Form,简称为NF)。范式的级别越高,发生操作异常的可能性越小,数据冗余越小,但由于关联多,读取数据时花费时间也会相应增加。

达到第三范式的关系仍有可能存在冗余等问题,所以关系数据库理论还有BCNF、4NF、5NF等范式。在实际应用中,一般达到了3NF的关系就可以认为是较为优化的关系。

1) 若一个关系中无重复组,且各属性都是不可再分割的基本数据项,则称该关系属于第一范式。在关系型数据库管理系统中,涉及到的研究对象都是满足1NF的规范化关系,不是1NF的关系称为非规范化的关系。

2) 若一个关系属于1NF,且每一个非主属性都完全函数依赖于主关键字,则称该关系属于第二范式。若某关系中主关键字由一个字段构成,则1NF自然满足2NF。

3) 若一个关系属于2NF,且每一个非主属性都不传递函数依赖于主关键字,则称该关系