

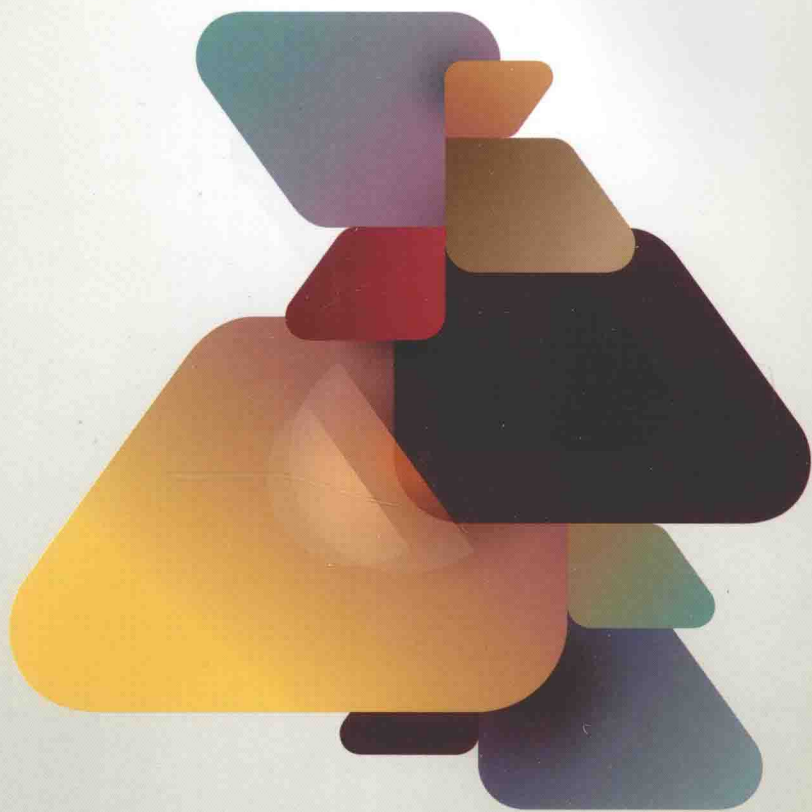
高等院校计算机应用系列教材

Access 2010

数据库应用技术

Access 2010 Shujuku Yingyong Jishu

朱广华 主编



138AC-43

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等院校计算机应用系列教材

Access 2010 数据库应用技术

朱广华 主 编

廖志斌 余 浚 李 培 张晓利 副主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书以应用为目的,以案例为引导,详细介绍了 Access 的主要功能和使用方法。全书共 11 章,内容包括 Access 基础、数据库的创建和操作、表的建立和管理、查询的创建和使用、窗体的设计 and 应用、报表的创建和使用、宏的建立和使用、Access 的编程工具 VBA 以及 Access 综合应用案例等内容。通过引例,循序渐进地介绍数据库的设计、建立和使用方法,能让读者在较短的时间内掌握 Access 的基本功能和操作以及 Access 的编程功能和技巧,能够完成小型数据库应用系统的开发。

本书内容充实、结构严谨,突出操作性和实践性,语言通俗易懂、深入浅出,案例丰富。在内容选材上,注重理论与实践并重,突出实践应用;在内容编排上,注重各知识点的合理安排,层次清楚;在写作方法上,由浅入深,循序渐进。

本书适合作为高等学校“大学计算机基础”课程的教材,也可作为全国计算机等级考试二级 Access 的培训或自学教材。

图书在版编目(CIP)数据

Access 2010 数据库应用技术/朱广华主编. —北京:
中国铁道出版社, 2014. 7
高等院校计算机应用系列教材
ISBN 978-7-113-18638-8

I. ①A… II. ①朱… 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(2014)第 108320 号

书 名: Access 2010 数据库应用技术
作 者: 朱广华 主编

策 划: 巨 凤

读者热线: 400-668-0820

责任编辑: 周 欣 冯彩茹

封面设计: 一克米工作室

责任校对: 汤淑梅

责任印制: 李 佳

出版发行: 中国铁道出版社(100054, 北京市西城区右安门西街 8 号)

网 址: <http://www.51eds.com>

印 刷: 航远印刷有限公司

版 次: 2014 年 7 月第 1 版 2014 年 7 月第 1 次印刷

开 本: 787mm×1 092mm 1/16 印张: 15.5 字数: 383 千

印 数: 1~3 000 册

书 号: ISBN 978-7-113-18638-8

定 价: 32.00 元

版权所有 侵权必究

凡购买铁道版图书,如有印制质量问题,请与本社教材图书营销部联系调换。电话:(010) 63550836

打击盗版举报电话:(010) 51873659

数据库技术是构成信息系统的基础,建设以数据库为核心的各类信息系统和应用系统对提高企业效益、改善部门管理具有重要的意义。因此学习和掌握数据库技术已成为大学生的必修课程。

Access 是一种小型关系数据库管理系统,其完整的数据库概念、友好的用户操作界面、可靠的数据管理方式、面向对象的操作理念以及强大的网络支持功能,受到了众多小型数据库应用系统开发者的青睐,成为当今最受欢迎的数据库软件之一。

本书以系统性、科学性和实用性为原则,以结构严谨、布局合理、概念清晰、内容适宜、重视实践能力为培养目标,具有如下几个方面的特点:

第一,在内容安排上,从基本概念、使用方法和实际应用 3 个方面进行介绍。在编写过程中,坚持从实际应用出发,以典型案例为引导,介绍 Access 的基本功能和编程功能。

第二,理论联系实际,突出操作性和实践性。

第三,内容注意适应教学要求,重视学生的认知规律。

第四,适应范围较广。

第五,配套的习题和实验。本书每章后面均附有习题,同时还编写了与本书配套的实训指导书,指导学生通过实验和练习,掌握各章的知识要点。

全书共 11 章,第 1 章简单介绍了数据库基础知识、数据库设计等数据库基本理论;第 2 章简单介绍了 Access 2010 的基本框架;第 3 章和第 4 章介绍了数据库和表的建立,以及对表的操作和管理;第 5 章详细介绍了 Access 重要对象查询的创建和使用;第 6 章和第 7 章介绍了如何创建窗体和报表,如何更好地应用窗体和报表;第 8 章和第 9 章详细介绍了如何使用 Access 提供的宏和 VBA 模块实现数据库的自动处理,提高使用 Access 的效率;第 10 和 11 章提供了 2 个实际应用 Access 创建数据库管理系统的案例,方便学生通过实践巩固前面掌握的基本知识,进而具备一定的数据库系统开发的经验和能力。

本书由多年从事计算机教学的一线教师编写,由朱广华任主编,廖志斌、余浚、李培、张晓利任副主编。具体分工如下:第 1 章和第 9 章由朱广华编写,第 5 章、第 6 章和第 12 章由余浚编写,第 2 章、第 7 章和第 8 章由李培编写,第 10 章由廖志斌编写,第 3 章、第 4 章和第 11 章由张晓利编写,全书由朱广华策划和统稿。

本书的编写得到很多老师、同仁和亲友的帮助与支持，特别是乔平安老师、秦承德老师对本书的编写提供了极大的帮助。邵凯、李莹莹、段立军参与了本书的校对工作。在此对以上人士和单位表示衷心的感谢。

由于本书涉及内容广泛，编写时间仓促，加之编者水平有限，书中难免存在疏漏和不足之处，恳请读者提出宝贵意见。

编 者

2014 年 4 月

第 1 章 数据库系统基本知识	1	2.4.1 数据表	23
1.1 数据库系统概述	1	2.4.2 查询	24
1.1.1 数据库系统的构成	1	2.4.3 窗体	25
1.1.2 数据库管理系统的功能	2	2.4.4 报表	25
1.2 数据模型	3	2.4.5 宏	26
1.2.1 数据模型的概念	3	2.4.6 模块	26
1.2.2 关系模型中常使用的术语	5	2.4.7 SharePoint 网站	27
1.2.3 关系模型的完整性规则	6	本章小结	28
1.2.4 关系型数据库的 基本关系操作	7	习题 2	28
1.2.5 数据的组成层次	10	第 3 章 数据库的基本操作	29
1.3 数据库设计方法和步骤	10	3.1 数据库的概念	29
1.3.1 程序设计方法与风格	11	3.1.1 数据库概述	29
1.3.2 程序设计方法的发展	11	3.1.2 关系数据库	29
1.3.3 数据库设计步骤	13	3.1.3 Access 数据库的 基本概念	30
1.4 数据库的运行与维护	14	3.2 数据库窗口	30
本章小结	15	3.2.1 Access 2010 数据库窗口 ...	30
习题 1	15	3.2.2 表窗口	31
第 2 章 Access 2010 系统简介	16	3.2.3 查询窗口	31
2.1 Access 2010 系统的基本特点	16	3.2.4 窗体窗口	32
2.2 Access 2010 的窗口界面	17	3.2.5 报表窗口	32
2.2.1 Backstage 视图	17	3.2.6 宏窗口	33
2.2.2 标题栏和快速访问 工具栏	18	3.2.7 模块窗口	33
2.2.3 功能区	19	3.3 数据库的创建	33
2.2.4 “导航”窗格	21	3.3.1 数据库文件	33
2.2.5 工作区	22	3.3.2 创建 Access 数据库的 方法	34
2.2.6 状态栏	22	3.4 数据库的打开	37
2.2.7 浮动工具栏	23	3.4.1 打开最近使用的 Access 数据库	37
2.3 Access 2010 的启动与退出	23	3.4.2 打开很久没有使用的	
2.4 Access 2010 系统的基本对象	23		

Access 数据库	38	5.1.1 查询的概念	75
3.4.3 数据库的打开方式	39	5.1.2 查询的创建方法和分类	75
3.5 数据库的关闭	40	5.1.3 查询的基本操作	76
本章小结	40	5.2 查询向导创建查询	78
习题 3	41	5.2.1 简单查询向导	79
第 4 章 表的基本操作	43	5.2.2 交叉表查询向导	81
4.1 表的基本知识	42	5.2.3 查找重复项查询向导	83
4.1.1 表与主题	42	5.2.4 查询不匹配项查询向导	85
4.1.2 表的结构	42	5.3 查询设计创建查询	88
4.1.3 字段的数据类型	43	5.3.1 查询设计视图基本结构	88
4.2 数据表的创建及保存方法	44	5.3.2 选择查询设计	89
4.2.1 使用设计器创建表	44	5.3.3 参数查询设计	90
4.2.2 通过输入数据创建表	54	5.3.4 交叉表查询设计	92
4.2.3 使用表模板创建表	55	5.3.5 操作查询设计	93
4.2.4 通过“导入表”获取外部 数据	56	5.4 SQL 命令创建查询	98
4.2.5 通过链接创建表	59	5.4.1 SQL 简介	98
4.2.6 几种创建表方法的比较	59	5.4.2 创建联合查询	100
4.2.7 表的保存	60	5.4.3 创建数据定义查询	100
4.3 表的编辑与维护	60	5.4.4 创建传递查询	102
4.3.1 修改表结构	60	5.4.5 创建子查询	107
4.3.2 编辑数据表记录	61	本章小结	108
4.3.3 表的复制	63	习题 5	108
4.3.4 表的删除	63	第 6 章 窗体	109
4.3.5 表的重命名	64	6.1 窗体的基本知识	109
4.3.6 表的导出	64	6.1.1 窗体的定义与功能	109
4.4 表间关系的建立与修改	67	6.1.2 窗体的数据源与 视图方式	110
4.4.1 表间关系的概念	67	6.1.3 窗体视图方式的选择与 窗体的类型	112
4.4.2 主键和外键	67	6.1.4 窗体的结构	113
4.4.3 表间关系的分类	67	6.2 窗体的创建	114
4.4.4 编辑关系中的约束	68	6.2.1 “窗体”创建窗体	114
4.4.5 建立表间关系的操作	70	6.2.2 “窗体设计”创建窗体	114
本章小结	72	6.2.3 “空白窗体”创建窗体	116
习题 4	73	6.2.4 “窗体向导”创建窗体	116
第 5 章 查询	75	6.2.5 “其他窗体”创建窗体	120
5.1 有关查询的基本知识	75		

6.3 窗体控件.....	123	8.3.1 序列宏的创建	147
6.3.1 窗体控件概述	123	8.3.2 宏组的创建	149
6.3.2 窗体控件的使用.....	123	8.3.3 将宏转换为模块.....	152
6.3.3 常用窗体控件	125	8.4 常用的宏操作	153
本章小结	127	8.5 宏的运行	156
习题 6	127	8.5.1 直接运行宏	156
第 7 章 报表制作	128	8.5.2 运行宏组中的宏	157
7.1 报表的基础知识	128	8.5.3 运行宏或事件过程以响应窗 体、报表或控件的事件.....	158
7.1.1 报表的类型	128	8.6 宏的创建实例	159
7.1.2 报表的结构	129	本章小结.....	164
7.1.3 报表的视图方式.....	130	习题 8	164
7.2 创建与维护报表	131	第 9 章 模块与 VBA.....	165
7.2.1 使用“自动创建报表” 创建报表	131	9.1 模块与 VBA 概述.....	165
7.2.2 使用各类向导创建报表....	132	9.1.1 关于模块	165
7.2.3 在设计视图中自行创建 报表.....	137	9.1.2 关于 VBA	166
7.2.4 几种创建报表方法的 比较	138	9.2 VBA 编程基础	167
7.3 报表设计.....	139	9.2.1 语句的语法规则.....	167
7.3.1 报表工具箱的使用.....	139	9.2.2 编程中用到的常量 与变量	170
7.3.2 为报表添加标题、字段名、 字段内容	140	9.2.3 函数.....	171
7.3.3 排序和分组	140	9.2.4 数组	174
7.3.4 美化报表	141	9.2.5 运算符与表达式.....	175
7.4 报表的打印输出	142	9.2.6 常用的事件	177
本章小结	144	9.2.7 数据类型	178
习题 7	144	9.3 创建 VBA 模块与编程环境	178
第 8 章 宏的应用	145	9.3.1 进入 VBE 编程环境	178
8.1 宏与宏组的概述	145	9.3.2 VBE 编辑窗口与编辑器..	179
8.2 运行宏的条件	146	9.3.3 创建新过程	180
8.2.1 数据处理事件	146	9.4 程序的组成与基本结构形式	181
8.2.2 焦点处理事件	146	9.4.1 程序设计中语句 书写规则	181
8.2.3 键盘输入事件	147	9.4.2 程序设计中的常用语句....	182
8.2.4 鼠标操作事件	147	9.4.3 结构程序设计	184
8.3 宏与宏组的创建	147	9.4.4 数据库引擎及接口	188
		9.4.5 VBA 访问数据库的类型....	188

9.4.6 数据访问对象	188	10.12 将主窗体设置为启动窗体	214
9.4.7 ActiveX 数据库对象	189	10.13 为系统设置和撤销密码	215
9.5 程序的调试方法	189	10.14 写出系统任务说明书	217
9.5.1 调试工具的使用	189	本章小结	217
9.5.2 调试工具示例	191	习题 10	218
9.6 数据的导入与导出	192	第 11 章 “基于 Web 的图书管理系统”	
9.6.1 外部数据	193	示例	219
9.6.2 数据的导入与导出	193	11.1 Web 应用环境的建立	219
9.6.3 Office 系列软件之间的数据共享	194	11.1.1 Web 服务器的配置	219
本章小结	194	11.1.2 Web 应用系统的发布	227
习题 9	195	11.2 总体功能分析	230
第 10 章 使用 Access 2010 实现“学生信息管理系统”示例	196	11.2.1 系统需求分析	230
10.1 需求分析	196	11.2.2 逻辑模型设计	231
10.2 概念设计	197	11.3 数据库创建	231
10.3 逻辑结构设计	198	11.3.1 创建数据库	232
10.4 物理设计	198	11.3.2 创建数据库表	233
10.5 设计步骤	199	11.3.3 查询的创建	233
10.6 建立表间关系	202	11.3.4 窗体的创建	234
10.7 将各个功能模块细化	204	11.3.5 报表的创建	236
10.8 设计编辑各个表的窗体	204	11.3.6 “控制面板”窗体的创建	237
10.9 设计各个查询及查询窗体	207	11.4 网页创建	238
10.10 设计各个报表	211	本章小结	239
10.11 设计“学生信息管理系统”主窗体	214	习题 11	239
		参考文献	240

第 1 章 | 数据库系统基本知识

1.1 数据库系统概述

1.1.1 数据库系统的构成

数据库系统 (DataBase System, DBS) 是指计算机系统中引入数据库后的系统, 一般由数据库、数据库管理系统 (及其开发工具)、应用系统、数据库管理员和用户构成。应当指出的是, 数据库的建立、使用和维护等工作只靠一个数据库管理系统 (DataBase Management System, DBMS) 是远远不够的, 还要由专门的人员来完成, 这些人称为数据库管理员 (DataBase Administrator, DBA)。

1. 硬件平台及数据库

由于数据库系统数据量很大, 加之 DBMS 丰富的功能使得自身的规模也很大, 因此整个数据库系统对硬件资源提出了较高的要求:

- (1) 要有足够大的内存, 存放操作系统、DBMS 的核心模块、缓冲区和应用程序。
- (2) 有足够大的磁盘等直接存取设备存放数据库, 有足够空间的数据备份设备。
- (3) 系统有较高的通道能力, 以提高数据传输率。

2. 软件

数据库系统的软件主要包括:

- (1) DBMS。DBMS 是为数据库的建立、使用和维护配置的软件。
- (2) 支持 DBMS 运行的操作系统。
- (3) 具有与数据库接口的高级语言及其编译系统, 便于开发应用程序。
- (4) 以 DBMS 为核心的应用开发工具。
- (5) 为特定应用环境开发的数据库应用系统。

3. 人员

开发、管理和使用数据库系统的人员主要是: 数据库管理员、系统分析员、数据库设计人员、应用程序员和最终用户。不同的人员涉及不同的数据抽象级别, 具有不同的数据视图, 如图 1-1 所示, 其各自的职责分别是:

1) 数据库管理员

在数据库系统环境下, 有两类共享资源: 一类是数据库, 另一类是数据库管理系统软件。因此需要有专门的管理机构来监督和管理数据库系统。DBA 则是这个机构的一个 (组) 人员, 负责全面管理和控制数据库系统。

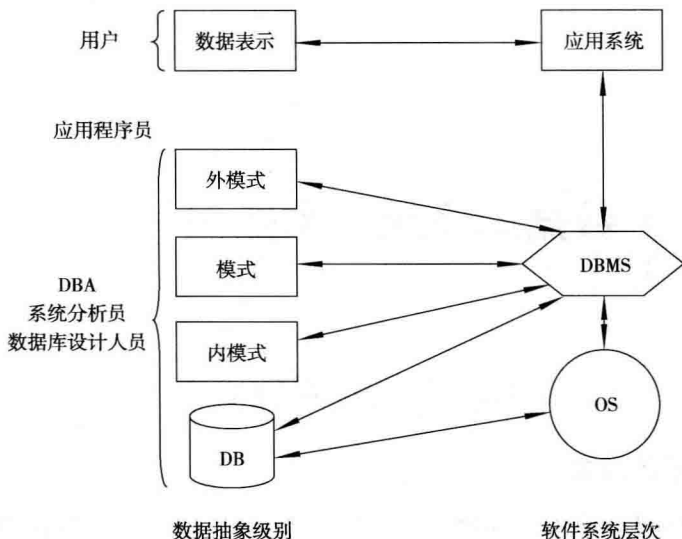


图 1-1 各种人员的数据视图

2) 系统分析员

系统分析员负责应用系统的需求分析和规范说明,要和用户及 DBA 相结合,确定系统的硬件软件配置,并参与数据库系统的概要设计。

3) 数据库设计人员

数据库设计人员负责数据库中数据的确定、数据库各级模式的设计。数据库设计人员必须参加用户需求调查和系统分析,然后进行数据库设计。在很多情况下,数据库设计人员就是由数据库管理员担任的。

4) 应用程序员

应用程序员负责设计和编写应用系统的程序模块,并进行调试和安装。

5) 用户

用户是指最终用户。最终用户通过应用系统的用户接口使用数据库。常用的接口方式有浏览器、菜单驱动、表格操作、图形显示、报表书写等,给用户提供了简明直观的数据表示。

1.1.2 数据库管理系统的功能

数据库管理系统 (DBMS) 是位于用户与操作系统 (OS) 之间的一层数据管理软件,是数据库系统的核心。它为用户或应用程序提供访问数据库的方法,包括数据库的建立、查询、更新以及各种数据控制。DBMS 总是基于某种数据模型,主要有层次型、网状型、关系型和面向对象型等。DBMS 具有以下基本功能:

1. 对象定义功能

DBMS 通过提供数据定义语言 (Data Definition Language, DDL) 来实现对数据库中数据对象的定义,例如:对外模式、模式和内模式的描述和定义,数据库完整性的定义,安全保密的定义 (如用户口令、级别、存取权限),存取路径 (如索引) 的定义。

2. 数据操纵功能

DBMS 通过提供数据操纵语言 (Data Manipulation Language, DML) 来实现对数据库中数据的

一些基本操作,如检索、插入、修改、删除和排序等。DML一般包括两类:一类是嵌入主语言的DML,如嵌入到C++等高级语言(称为宿主语言)中;另一类是非嵌入式语言,它的语法简单,可以单独使用,由单独的解释或编译系统来执行,所以一般称为自主型或自含型DML。

3. 数据库控制功能

DBMS提供的数据库控制语言(Data Control Language, DCL)保证数据库操作都在统一的管理下协调工作,以确保事务处理的正常工作,保证数据库的正确性、安全性、有效性和多用户对数据的并发使用,以及发生故障后的系统恢复功能等。如安全性检查、完整性约束条件的检查、数据库内部的索引和数据字典自动维护、缓存区大小的设置等。

4. 数据组织、存储和管理功能

DBMS要分类组织、存储和管理各种数据,包括数据字典、用户数据、存取路径等;要确定以何种文件结构和存取方式在存储级上组织这些数据,以及如何实现数据之间的联系。数据组织和存储的基本目标是提高存储空间利用率和方便存取,提供多种存取方法(如索引查找、Hash查找、顺序查找等)以提高存取效率。

5. 其他功能

DBMS的其他功能包括网络通信功能,一个DBMS与另一个DBMS或文件系统的数据库转换功能,异构数据库之间的互访和互操作功能等。

1.2 数据模型

1.2.1 数据模型的概念

数据模型是数据库系统的基础,是数据库的形式框架,用于描述一组数据的概念和定义。现有的数据库系统均是基于某种数据模型的,因此,了解数据模型的基本概念是学习数据库的基础。

1. 数据模型的构成

不同的数据模型实际上是提供给我们模型化数据和信息的不同工具。数据模型包含3个层面:

1) 概念数据模型

概念数据模型是面向数据库用户现实世界的数据库模型,主要用来描述世界的概念化结构,它是数据库的设计人员在设计的初始阶段,摆脱计算机系统及DBMS的具体技术问题,集中精力分析数据及数据之间的联系等。概念数据模型必须换成逻辑数据模型,才能在DBMS中实现。

2) 逻辑数据模型

逻辑数据模型是用户从数据库中看到的数据库模型,是具体的DBMS所支持的数据库模型,如网状数据模型、层次数据模型、关系数据模型等。此模型既要面向用户,又要面向系统。通常在不引起概念混乱的情况下将逻辑数据模型简称为数据库模型。

3) 物理数据模型

物理数据模型是描述数据在存储介质上的组织机构的数据库模型,它不但与具体的DBMS有关,而且还与操作系统及硬件有关。每一种逻辑数据模型在实现时都有其对应的物理数据模型。DBMS为了保证其独立性与可移植性,大部分物理数据模型的实现工作由系统自动完成,而设计者只设计索引、聚集等特殊结构。

2. 常用数据模型类型

目前,数据库领域中最常用的数据模型有 4 种:层次模型 (Hierarchical Model)、网状模型 (Network Model)、关系模型 (Relational Model)、面向对象模型 (Object Oriented Model)。其中层次模型和网状模型统称为非关系模型。

下面简单介绍一下层次模型、网状模型和关系模型:

1) 层次模型

层次数据库系统采用层次模型作为数据的组织方式。用树形 (层次) 结构表示实体类型以及实体间的联系是层次模型的主要特征。层次结构是一棵树,树的结点是记录类型,根结点只有一个,根结点以外的结点只有一个双亲结点,每一个结点可以有多个孩子结点。层次模型如图 1-2 所示。

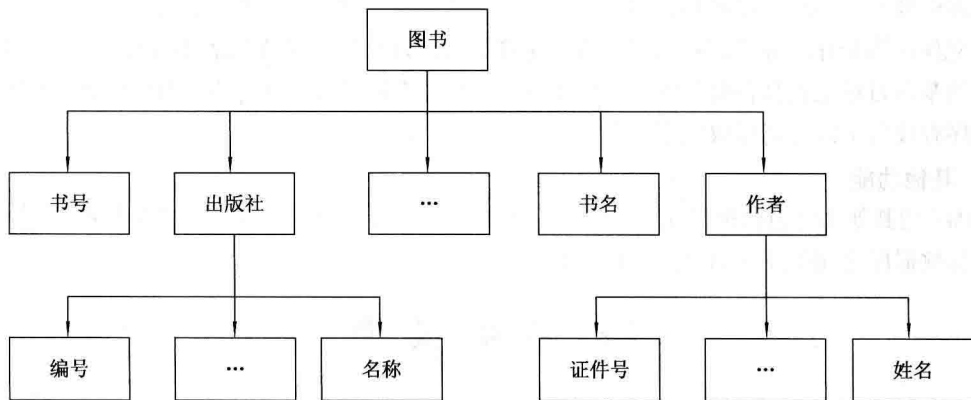


图 1-2 层次模型

层次模型的一个基本特点是,任何一个给定的记录值只有按照其路径查看时,才能显示它的全部意义。没有哪一个记录值能够脱离双亲记录而独立存在。

2) 网状模型

在数据库中,把满足以下两个条件的基本层次联系集合称为网状模型:

- (1) 允许一个以上的结点无双亲。
- (2) 一个结点可以有多于一个的双亲。

网状模型是一种比层次模型更具普遍性的结构,它去掉了层次模型的两个限制:允许多个结点没有双亲结点,允许结点有多个双亲结点。此外,它还允许两个结点之间有多种联系。因此网状模型可以更直接地描述现实世界,而层次模型实际上是网状模型的一个特例。

在网状模型中,一个子结点可以有多个父结点,在两个结点之间可以有一种或多种联系。记录之间的联系是通过指针实现的,因此时间的联系十分密切。网状模型的数据结构在物理上易于实现、效率较高,但是编写应用程序较复杂,程序员必须熟悉数据库的逻辑结构。

网状模型如图 1-3 所示。

3) 关系模型

网状数据库和层次数据库已经很好地解决了数据的集中

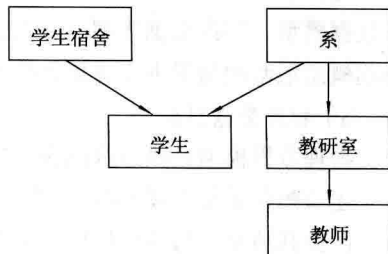


图 1-3 网状模型

和共享问题,但在数据独立性和抽象级别上仍有很大的欠缺:用户在对这两种数据库进行存取时,仍需明确数据的存储结构,指出存取路径。关系数据库较好地解决了这些问题,关系数据库采用关系模型进行数据的组织。关系模型比较简单,容易为初学者所接受。

关系模型是若干个关系模式的集合。关系模式的实例称为关系,每个关系实际上是一张二维表格。记录是表中的行,属性是表中的列。

图 1-4 “学生实体集和课程实体集的 E-R 图”对应的关系模式如表 1-1 所示。

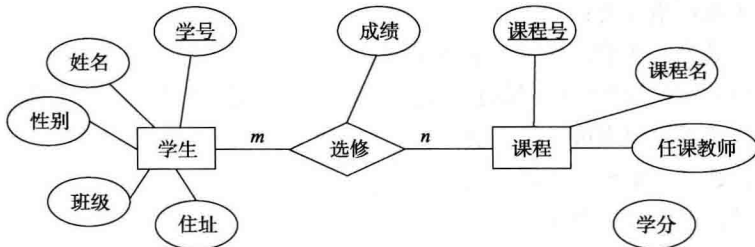


图 1-4 学生实体集和课程实体集的 E-R 图

表 1-1 关系模式

模式名	构成
学生	(<u>学号</u> 、姓名、性别、班级、住址)
选修	(<u>学号</u> 、课程号、成绩)
课程	(课程号、课程名、任课教师、学分)

1.2.2 关系模型中常使用的术语

在用户观点下,关系模型中数据的逻辑结构是一张二维表,它由行和列组成。现在以学生登记表(见图 1-5)为例,介绍关系模型中的一些术语。

学生登记表

学号	姓名	年龄	性别	系	年级
99001	王小明	19	女	社会学	99
99006	黄彭	20	男	计算机	99
99008	张斌	18	男	媒体艺术	99
⋮	⋮	⋮	⋮	⋮	⋮

图 1-5 关系模型的数据结构

(1) 关系 (Relation)。一个关系对应通常说的一张表,如图 1-5 中的学生登记表。

(2) 元组 (Tuple)。表中的一行即为一个元组。

(3) 属性 (Attribute)。表中的一列即为一个属性,给每一个属性起一个名称,即属性名。如“学生登记表”有 6 列,对应 6 个属性(学号,姓名,年龄,性别,系和年级)。

(4) 主键 (Key)。表中的某个属性组,它可以唯一确定一个元组,如图 1-5 中的学号,可以

唯一确定一个学生，也就成为本关系的主键。

(5) 域 (Domain)。属性的取值范围，如人的年龄一般在 1~150 岁之间，大学生年龄属性的域是 (14~38)，性别的域是 (男, 女)，系别的域是一个学校所有系名的集合。

(6) 分量。元组中的一个属性值。

(7) 关系模式。对关系的描述，一般表示为：

关系名 (属性名 1, 属性名 2, ..., 属性名 n)

例如上面的关系可描述为：

学生 (学号, 姓名, 年龄, 性别, 系和年级)

在关系模型中，实体以及实体间的联系都是用关系来描述。例如学生、课程、学生与课程之间的多对多联系在关系模型中可以如下表示：

学生 (学号, 姓名, 年龄, 性别, 系和年级)

课程 (课程号, 课程名, 学分)

选修 (学号, 课程号, 成绩)

关系模型要求关系必须是规范化的，即要求关系必须满足一定的规范条件。这些规范条件中最基本的一条就是，关系的每一个分量必须是一个不可分的数据项，也就是说，不允许表中嵌套。

1.2.3 关系模型的完整性规则

关系模型提供了 3 类完整性规则：实体完整性规则、参照完整性规则、用户定义的完整性规则。其中，实体完整性规则和参照完整性规则是关系模型必须满足的完整性的约束条件，称为关系完整性规则。

1. 实体完整性规则

实体完整性规则规定关系中元组的主键值不能为空。

例如，学生表 (见表 1-2) 的主键是学号，不允许为空或重复；课程表 (见表 1-3) 的主键是课程号，不允许为空或重复；成绩表 (见表 1-4) 的主键是学号和课程号的组合，不允许为空或重复。这 3 个主键的值在表中是唯一和确定的。

表 1-2 学 生 表

学 号	姓 名	性 别	班 级	住 址
2010001119	张瑜	女	网络一班	2-203
2010001120	马云飞	男	网络一班	7-306
2010001121	周婷	女	网络一班	2-503
⋮	⋮	⋮	⋮	⋮

表 1-3 课 程 表

课 程 号	课 程 名	任 课 教 师	学 分
1-01	计算机基础	马红	1
1-02	大学英语	张勇	1
1-03	高等数学	韩刚	2
⋮	⋮	⋮	⋮

表 1-4 成绩表

学 号	课 程 号	成 绩
20100001119	1-01	85
20100011119	1-03	90
20090011120	1-01	80
...	...	...

2. 参照完整性规则

参照完整性规则的形式定义如下：如果属性集 K 是关系模式 R_1 的主键，同时 K 也是关系模式 R_2 的外键，那么在 R_2 的关系中， K 的取值只有两种可能：或者为空值，或者等于 R_1 关系中某个主键值。这条规则的实质是“不允许引用不存在的实体”。

对于参照完整性规则，有 3 点需要注意：

(1) 外键和相应的主键可以不同名，但要定义在相同的值域上。

(2) 当 R_1 和 R_2 是同一个关系模式时，表示同一个关系中不同元组之间的联系。例如，表示课程之间选修联系的模式 $R(CNO, CNAME, PCNO)$ ，其属性表示课程号、课程名、选修课程的课程号， R 的主键值是 CNO ，而 $PCNO$ 就是一个外键，表示 $PCNO$ 值一定要在关系中存在（某个 CNO 值）。

(3) 外键值是否允许为空，应视具体问题而定。若外键是模式主键中的成分，则外键值不允许为空，否则允许为空。

在上述形式定义中，关系模式 R_1 的关系称为“参照关系”“主表”或者“父表”；关系模式 R_2 的关系称为“依赖关系”“副表”或者“子表”。

如前所述，学生表和课程表为主表，成绩表为副表，学号是学生表的主键、成绩表的外键，课程号是课程表的主键、成绩表的外键。成绩表中的学号必须是学生表中学号的有效值，学生表与成绩表之间的联系是通过学号实现的；同样，成绩表中的课程号必须是课程表中课程号的有效值，成绩表与课程表之间的联系是通过课程号实现的。

实体完整性规则和参照完整性规则是关系模型必须满足的规则，由系统自动支持。

3. 用户定义的完整性规则

用户定义的完整性规则是针对某一具体数据的约束条件，由应用环境决定。它反映某一具体应用所涉及的数据必须满足的语义要求。系统应提供定义和检验这类完整性的机制，以便用统一的系统方法处理它们。例如，学生成绩应该大于或等于零，职工的工龄应小于年龄，人的身高不能超过 2.8 m 等。

1.2.4 关系型数据库的基本关系操作

关系代数是施加于关系上的一组集合代数运算，每个运算都以一个或多个关系作为运算对象，并生成另外一个关系作为运算的结果。关系代数包含两类运算：传统的集合运算、专门的关系运算。

1. 传统的集合运算

传统的集合运算有并、差、交、笛卡儿积 4 种运算。

从集合论的观点来定义关系，将关系看成是若干个具有 K 个属性的元组集合。通过对关系进

行集合操作来完成查询请求。传统的集合运算是从关系的水平方向进行的,包括并、交、差及广义笛卡儿积,属于二目运算。

要使并、差、交运算有意义,必须满足两个条件:一是参与运算的两个关系具有相同的属性数目;二是这两个关系对应的属性取自同一个域,即属性的域相同或相容。

1) 并 (Union)

设关系 R 和关系 S 具有相同的目 K , 即两个关系都有 K 个属性, 且相应的属性取自同一个域, 则关系 R 与 S 的并是由属于 R 或属于 S 的元组构成的集合, 并运算的结果仍是 K 目关系。其形式定义如下:

$$R \cup S = \{t \mid t \in R \vee t \in S\}$$

其中, t 为元组变量; R 和 S 的元数相同。

2) 交 (Intersection)

设关系 R 和关系 S 具有相同的目 K , 即两个关系都有 K 个属性, 且相应的属性取自同一个域, 则关系 R 与 S 的交是由既属于 R 又属于 S 的元组构成的集合, 交运算的结果仍是 K 目关系。其形式定义如下:

$$R \cap S = \{t \mid t \in R \wedge t \in S\}$$

其中, t 为元组变量; R 和 S 的元数相同。交操作不是一个独立的操作。

3) 差 (Difference)

设关系 R 和关系 S 具有相同的目 K , 即两个关系都有 K 个属性, 且相应的属性取自同一个域, 则关系 R 与 S 的差是由属于 R 但不属于 S 的元组构成的集合, 差运算的结果仍是 K 目关系。其形式定义如下:

$$R - S = \{t \mid t \in R \wedge t \notin S\}$$

其中, t 为元组变量; R 和 S 的元数相同。

4) 广义笛卡儿积 (Extended Cartesian Product)

设关系 R 的属性数目是 K_1 , 元组数目为 m ; 关系 S 的属性数目是 K_2 , 元组数目为 n ; 则 R 和 S 的广义笛卡儿积是一个 (K_1+K_2) 列的 $(m \times n)$ 个元组的集合。记作 $R \times S$ 。

广义笛卡儿积是一个有序对的集合。有序对的第一个元素是关系 R 中的任何一个元组, 有序对的第二个元素是关系 S 中的任何一个元组。

如果 R 和 S 中有相同的属性名, 可在属性名前加上所属的关系名作为限定。

2. 专门的关系运算

专门的关系运算有选择、投影和连接运算。

1) 选择 (Selection)

从关系中找出满足给定条件的所有元组称为选择, 其中的条件以逻辑表达式的形式给出, 该逻辑表达式的值为真的元组被选取。

选择运算是从关系的水平方向进行运算, 是从关系 R 中选取符合给定条件的所有元组, 生成新的关系。记作:

$$\sigma_{\text{条件表达式}}(R)$$

其中, 条件表达式的基本形式为 $X \theta Y$, θ 表示运算符, 包括比较运算符 ($<$, $<=$, $>$, $>=$, $=$, $\neq$) 和逻辑运算符 ($\wedge$, $\vee$, $\neg$)。 X 和 Y 可以是属性、常量或简单函数。属性名可以用它的序号或者