

计算机基础与实训教材系列

Access 2010 数据库应用 基础教程

施兴家 王秉宏 编著

清华大学出版社
北 京

内 容 简 介

本书由浅入深、循序渐进地介绍了 Access 2010 数据库应用开发工具的详细内容。全书共分 12 章,包括数据库系统概论、Access 2010 工作环境及其各种数据库对象、数据库的创建、表的创建与使用、创建查询与窗体、报表的创建与打印、宏与模块等 VBA 编程技巧。最后一章综合运用前述内容,创建了一个简易的客户订单管理系统,使读者能够掌握数据库系统开发和设计的基本流程,以及数据库维护与安全管理的知识。

本书内容丰富,结构清晰,语言简练,图文并茂,具有很强的实用性和可操作性,是一本适合于大中专院校、职业院校及相关各类社会培训机构的优秀教材,也是广大初、中级电脑用户的自学参考书。

本书对应的电子教案、实例源代码和习题答案可以到 <http://www.tupwk.com.cn/edu> 网站下载。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Access 2010 数据库应用基础教程/施兴家,王秉宏 编著. —北京:清华大学出版社,2013.5
(计算机基础与实训教材系列)

ISBN 978-7-302-32138-5

I. ①A… II. ①施… ②王… III. ①关系数据库系统—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2013)第 083146 号

责任编辑:胡辰浩 袁建华

装帧设计:牛艳敏

责任校对:成凤进

责任印制:

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62794504

印 刷 者:

装 订 者:

经 销:全国新华书店

开 本:190mm×260mm 印 张:20 字 数:525 千字

版 次:2013 年 5 月第 1 版 印 次:2013 年 5 月第 1 次印刷

印 数:1~5000

定 价:35.00 元

产品编号:

编审委员会

计算机基础与实训教材系列

主任：闪四清 北京航空航天大学

委员：(以下编委顺序不分先后，按照姓氏笔画排列)

王永生 青海师范大学
王相林 杭州电子科技大学
卢 锋 南京邮电学院
申浩如 昆明学院计算机系
白中英 北京邮电大学计算机学院
石 磊 郑州大学信息工程学院
伍俊良 重庆大学
刘 悦 济南大学信息科学与工程学院
刘晓华 武汉工程大学
刘晓悦 河北理工大学计控学院
孙一林 北京师范大学信息科学与技术学院计算机系
朱居正 河南财经学院成功学院
何宗键 同济大学软件学院
吴裕功 天津大学
吴 磊 北方工业大学信息工程学院
宋海声 西北师范大学
张凤琴 空军工程大学
罗怡桂 同济大学
范训礼 西北大学信息科学与技术学院
胡景凡 北京信息工程学院
赵文静 西安建筑科技大学信息与控制工程学院
赵素华 辽宁大学
郝 平 浙江工业大学信息工程学院
崔洪斌 河北科技大学
崔晓利 湖南工学院
韩良智 北京科技大学管理学院
薛向阳 复旦大学计算机科学与工程系
瞿有甜 浙江师范大学

执行委员：陈 笑 胡辰浩 袁建华

执行编辑：胡辰浩 袁建华

丛书序

计算机基础与实训教材系列

计算机已经广泛应用于现代社会的各个领域,熟练使用计算机已经成为人们必备的技能之一。因此,如何快速地掌握计算机知识和使用技术,并应用于现实生活和实际工作中,已成为新世纪人才迫切需要解决的问题。

为适应这种需求,各类高等院校、高职高专、中职中专、培训学校都开设了计算机专业的课程,同时也将非计算机专业学生的计算机知识和技能教育纳入教学计划,并陆续出台了相应的教学大纲。基于以上因素,清华大学出版社组织一线教学精英编写了这套“计算机基础与实训教材系列”丛书,以满足大中专院校、职业院校及各类社会培训学校的教学需要。

一、丛书书目

本套教材涵盖了计算机各个应用领域,包括计算机硬件知识、操作系统、数据库、编程语言、文字录入和排版、办公软件、计算机网络、图形图像、三维动画、网页制作以及多媒体制作等。众多的图书品种可以满足各类院校相关课程设置的需要。

◎ 已出版的图书书目

《计算机基础实用教程(第二版)》	《中文版 Photoshop CS4 图像处理实用教程》
《电脑入门实用教程(第二版)》	《中文版 Flash CS4 动画制作实用教程》
《电脑办公自动化实用教程(第二版)》	《中文版 Dreamweaver CS4 网页制作实用教程》
《计算机组装与维护实用教程(第二版)》	《中文版 Illustrator CS4 平面设计实用教程》
《计算机基础实用教程(Windows 7+Office 2010 版)》	《中文版 InDesign CS4 实用教程》
《Windows 7 实用教程》	《中文版 CorelDRAW X4 平面设计实用教程》
《中文版 Word 2003 文档处理实用教程》	《中文版 3ds Max 2012 三维动画创作实用教程》
《中文版 PowerPoint 2003 幻灯片制作实用教程》	《中文版 Office 2007 实用教程》
《中文版 Excel 2003 电子表格实用教程》	《中文版 Word 2007 文档处理实用教程》
《中文版 Access 2003 数据库应用实用教程》	《中文版 Excel 2007 电子表格实用教程》
《中文版 Project 2003 实用教程》	《Excel 财务会计实战应用(第二版)》
《中文版 Office 2003 实用教程》	《中文版 PowerPoint 2007 幻灯片制作实用教程》
《Access 2010 数据库应用基础教程》	《中文版 Access 2007 数据库应用实例教程》
《多媒体技术及应用》	《中文版 Project 2007 实用教程》
《中文版 Premiere Pro CS4 多媒体制作实用教程》	《Office 2010 基础与实战》
《中文版 Premiere Pro CS5 多媒体制作实用教程》	《Director 11 多媒体开发实用教程》

(续表)

《ASP.NET 3.5 动态网站开发实用教程》	《中文版 AutoCAD 2010 实用教程》
《ASP.NET 4.0 动态网站开发实用教程》	《中文版 AutoCAD 2012 实用教程》
《ASP.NET 4.0(C#)实用教程》	《AutoCAD 建筑制图实用教程(2010版)》
《Java 程序设计实用教程》	《AutoCAD 机械制图实用教程(2012版)》
《JSP 动态网站开发实用教程》	《Mastercam X4 实用教程》
《C#程序设计实用教程》	《Mastercam X5 实用教程》
《Visual C# 2010 程序设计实用教程》	《中文版 Photoshop CS5 图像处理实用教程》
《Access 2010 数据库应用基础教程》	《中文版 Dreamweaver CS5 网页制作实用教程》
《SQL Server 2008 数据库应用实用教程》	《中文版 Flash CS5 动画制作实用教程》
《网络组建与管理实用教程》	《中文版 Illustrator CS5 平面设计实用教程》
《计算机网络技术实用教程》	《中文版 InDesign CS5 实用教程》
《局域网组建与管理实训教程》	《中文版 CorelDRAW X5 平面设计实用教程》

二、丛书特色

1、选题新颖，策划周全——为计算机教学量身打造

本套丛书注重理论知识与实践操作的紧密结合，同时突出上机操作环节。丛书作者均为各大院校的教学专家和业界精英，他们熟悉教学内容的编排，深谙学生的需求和接受能力，并将这种教学理念充分融入本套教材的编写中。

本套丛书全面贯彻“理论→实例→上机→习题”4阶段教学模式，在内容选择、结构安排上更加符合读者的认知习惯，从而达到老师易教、学生易学的目的。

2、教学结构科学合理，循序渐进——完全掌握“教学”与“自学”两种模式

本套丛书完全以大中专院校、职业院校及各类社会培训学校的教学需要为出发点，紧密结合学科的教学特点，由浅入深地安排章节内容，循序渐进地完成各种复杂知识的讲解，使学生能够一学就会、即学即用。

对教师而言，本套丛书根据实际教学情况安排好课时，提前组织好课前备课内容，使课堂教学过程更加条理化，同时方便学生学习，让学生在学完后有例可学、有题可练；对自学者而言，可以按照本书的章节安排逐步学习。

3、内容丰富、学习目标明确——全面提升“知识”与“能力”

本套丛书内容丰富，信息量大，章节结构完全按照教学大纲的要求来安排，并细化了每一章内容，符合教学需要和计算机用户的学习习惯。在每章的开始，列出了学习目标和本章重点，便于教师和学生提纲挈领地掌握本章知识点，每章的最后还附带有上机练习和习题两部分内容，教师可以参照上机练习，实时指导学生进行上机操作，使学生及时巩固所学的知识。自学者也可以按照上机练习内容进行自我训练，快速掌握相关知识。

4、实例精彩实用，讲解细致透彻——全方位解决实际遇到的问题

本套丛书精心安排了大量实例讲解，每个实例解决一个问题或是介绍一项技巧，以便读者在最短的时间内掌握计算机应用的操作方法，从而能够顺利解决实践工作中的问题。

范例讲解语言通俗易懂，通过添加大量的“提示”和“知识点”的方式突出重要知识点，以便加深读者对关键技术和理论知识的印象，使读者轻松领悟每一个范例的精髓所在，提高读者的思考能力和分析能力，同时也加强了读者的综合应用能力。

5、版式简洁大方，排版紧凑，标注清晰明确——打造一个轻松阅读的环境

本套丛书的版式简洁、大方，合理安排图与文字的占用空间，对于标题、正文、提示和知识点等都设计了醒目的字体符号，读者阅读起来会感到轻松愉快。

三、读者定位

本丛书为所有从事计算机教学的老师和自学人员而编写，是一套适合于大中专院校、职业院校及各类社会培训学校的优秀教材，也可作为计算机初、中级用户和计算机爱好者学习计算机知识的自学参考书。

四、周到体贴的售后服务

为了方便教学，本套丛书提供精心制作的 PowerPoint 教学课件(即电子教案)、素材、源文件、习题答案等相关内容，可在网站上免费下载，也可发送电子邮件至 wkservice@vip.163.com 索取。

此外，如果读者在使用本系列图书的过程中遇到疑惑或困难，可以在丛书支持网站 (<http://www.tupwk.com.cn/edu>) 的互动论坛上留言，本丛书的作者或技术编辑会及时提供相应的技术支持。咨询电话：010-62796045。

Access 是 Microsoft 公司的 Office 办公自动化软件的一个重要组成部分。作为一个小型的关系型数据库管理系统，它可以有效地组织、管理和共享数据库的信息，具有界面友好、易学易用、开发简单、接口灵活等特点。为了使广大数据库初学者能够快速掌握这款优秀的数据库管理系统，我们选择其最新版本 Access 2010，精心策划并编写了本书。

本书从教学实际需求出发，合理安排知识结构，从零开始、由浅入深、全面介绍了 Access 2010 关系型数据库的各项功能、各种数据库对象的创建以及数据库安全管理的相关知识，本书共分为 12 章，主要内容如下：

第 1 章是数据库系统概述，从零开始介绍了数据库的基本概念，数据库系统结构，关系数据库理论的相关知识，以及创建数据库应用系统的基本步骤。

第 2 章介绍了 Access 2010 的工作环境，包括各功能区的布局与作用，各种数据库对象的基本概念等，本章是后面章节的基础，学好本章可以为后面的学习打下良好的基础。

第 3 章介绍了数据库的创建与使用，包括创建数据库的方法，Access 2010 数据库与早期版本的区别以及数据库的基本操作等内容。

第 4 章介绍了数据表的创建与使用，包括使用向导、使用设计视图、使用模板等多种创建表的方法，以及表间关系的建立等。

第 5 章继续介绍表的相关知识，主要包括数据记录的增删改查等操作、数据的排序与筛选，数据导出以及行汇总统计等内容。

第 6 章介绍了查询对象的创建与使用，包括查询的类型、SQL 语言的基本语法、各种查询的创建方法和技巧等内容。

第 7 章介绍了窗体的创建与设计，包括窗体的功能与分类、窗体的各种创建方法、控件的使用，主/子窗体以及切换面板窗体的创建与使用等。

第 8 章介绍了报表的创建与打印，包括报表的分类、报表的创建、编辑和打印报表，以及主/子报表的创建等。

第 9 章介绍了宏的用法，包括宏的类型、宏的创建方法以及调试和运行宏等内容。

第 10 章介绍了模块与 VBA 编程相关的知识，包括 VBA 编程环境、面向对象编程的基本概念、VBA 的基本语法和流程控制语句、过程与函数等内容。

第 11 章介绍了数据库的安全与管理，包括数据库的压缩与备份、数据库的加密与解密、数据库的打包与签署等内容。

第 12 章通过创建一个完整的数据库应用系统，综合应用前述内容，使用 Access 2010 开发一个客户订单管理系统。

本书图文并茂，条理清晰，通俗易懂，内容丰富，在讲解每个知识点时都配有相应的实例，方便读者上机实践。同时在难于理解和掌握的部分内容上给出相关提示，让读者能够快速提



高操作技能。此外，本书配有大量综合实例和练习，让读者在不断的实际操作中更加牢固地掌握书中讲解的内容。

本书是集体智慧的结晶，除封面署名的作者外，参加本书编写和制作的人员还有丁琳、张程、王有材、石志明、王玲、罗美云、吴萍、石耀东、林运涛、赵鹏、丁乐、赵传同、毛传明、吴勇、秦吉、张秀云、杜百贵、贺桂仙和王娟等。由于作者水平有限，加之创作时间仓促，本书不足之处在所难免，欢迎广大读者批评指正。我们的邮箱是 huchenhao@263.net，电话是 010-62796045。

作者
2013年2月



推荐课时安排

计算机基础与实训教材系列

章 名	重点掌握内容	教学课时
第1章 数据库系统概述	<ol style="list-style-type: none">1. 数据库相关的概念2. 数据库系统结构3. 数据模型4. 关系数据库5. 数据库设计	3 学时
第2章 Access 2010 基础	<ol style="list-style-type: none">1. Access 2010 工作界面2. Access 2010 的功能区3. Access 2010 数据库对象	2 学时
第3章 数据库的创建与使用	<ol style="list-style-type: none">1. Access 数据库结构2. 从模块创建数据库3. 复制数据库对象4. 数据库的基本操作	3 学时
第4章 表的创建与使用	<ol style="list-style-type: none">1. 设置字段属性2. 使用数据表视图创建表3. 使用向导创建表4. 创建查阅字段列5. 表的相关操作6. 表之间的关系	4 学时
第5章 表中数据记录的操作	<ol style="list-style-type: none">1. 数据的增删改查2. 隐藏列与取消隐藏列3. 设置数据表格式4. 数据排序与筛选5. 统计汇总行6. 导出数据表	4 学时
第6章 查询	<ol style="list-style-type: none">1. 查询与表的关系2. 使用 SELECT 语句3. 使用向导创建查询4. 创建参数查询5. 创建操作查询6. 创建 SQL 查询	5 学时



(续表)

章 名	重点掌握内容	教学课时
第7章 窗体	<ol style="list-style-type: none">1. 窗体的分类2. 窗体的视图3. 快速创建窗体4. 设计窗体5. 弹出式窗体6. 子窗体的创建	5 学时
第8章 报表	<ol style="list-style-type: none">1. 报表的视图2. 创建报表3. 编辑报表4. 打印报表5. 创建子报表6. 交叉报表	3 学时
第9章 宏	<ol style="list-style-type: none">1. 宏的类型2. 创建简单宏3. 创建条件宏4. 运行宏5. 调试宏	4 学时
第10章 模块与VBA	<ol style="list-style-type: none">1. 什么是VBA2. 创建和运行模块3. VBA 编程环境4. 流程控制语句5. 调试VBA 程序6. 过程与函数7. VBA 代码的保护	6 学时
第11章 数据库管理与安全	<ol style="list-style-type: none">1. 压缩和修复数据库2. 备份和恢复数据库3. 数据库加密与解密4. 数据库的打包与签署	2 学时
第12章 客户订单管理系统	<ol style="list-style-type: none">1. 需求分析2. 系统设计3. 创建数据库对象4. 创建 AutoExec 宏	4 学时



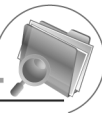
目 录

计算机基础与实训教材系列

第 1 章 数据库系统概述	1	第 2 章 Access 2010 基础	35
1.1 数据库相关的概念	1	2.1 初识 Access 2010	35
1.1.1 数据与数据处理	2	2.1.1 Access 简介	35
1.1.2 数据库	3	2.1.2 启动 Access 2010	36
1.1.3 数据库系统	4	2.1.3 关闭 Access 2010	37
1.1.4 数据库管理系统(DBMS)	8	2.2 Access 2010 的工作界面	38
1.1.5 数据库应用系统(DBAS)	10	2.2.1 起始页	38
1.2 数据库系统的体系结构	10	2.2.2 标题栏	39
1.2.1 内部体系结构	10	2.2.3 功能区	40
1.2.2 外部体系结构	11	2.2.4 导航窗格	40
1.3 数据模型	12	2.2.5 状态栏	41
1.3.1 概念模型	13	2.3 Access 2010 的功能区	41
1.3.2 用 E-R 方法表示概念模型	14	2.3.1 显示或隐藏功能区	42
1.3.3 逻辑数据模型	15	2.3.2 常规命令选项卡	42
1.3.4 数据模型的三要素	17	2.3.3 上下文命令选项卡	46
1.4 关系数据库	18	2.3.4 自定义功能区	47
1.4.1 关系模型中的基本术语	18	2.4 Access 2010 数据库对象	49
1.4.2 关系数据库中表之间的关系	19	2.4.1 表	49
1.4.3 关系模型的完整性约束	20	2.4.2 查询	49
1.5 关系代数	21	2.4.3 窗体	50
1.5.1 传统的集合运算	22	2.4.4 报表	50
1.5.2 专门的关系运算	24	2.4.5 宏	51
1.6 规范化理论	25	2.4.6 模块	51
1.6.1 模式规范化的必要性	25	2.5 上机练习	52
1.6.2 规范化理论的基本概念	26	2.6 习题	53
1.7 数据库语言	28	第 3 章 数据库的创建与使用	54
1.7.1 数据定义语言 DDL	28	3.1 Access 数据库概述	54
1.7.2 数据操纵语言 DML	29	3.1.1 Access 数据库结构	54
1.8 数据库设计	29	3.1.2 Access 数据库文件	55
1.8.1 数据库设计的目标	29	3.2 创建数据库	55
1.8.2 数据库设计的特点	29	3.2.1 创建空白数据库	56
1.8.3 数据库设计的方法	30	3.2.2 使用模板创建数据库	57
1.8.4 数据库设计的步骤	31	3.2.3 转换数据库	58
1.9 上机练习	33	3.3 操作数据库和数据库对象	59
1.10 习题	34		



3.3.1	打开数据库	59	5.1.1	增加新记录	105
3.3.2	数据库对象的组织	60	5.1.2	输入数据	106
3.3.3	操作数据库对象	62	5.1.3	修改记录	107
3.3.4	保存数据库	64	5.1.4	查找与替换	108
3.3.5	关闭数据库	65	5.1.5	复制数据	109
3.4	上机练习	65	5.1.6	删除记录	110
3.5	习题	66	5.2	改变数据记录的显示方式	111
第 4 章	表的创建与使用	67	5.2.1	隐藏列/取消隐藏列	111
4.1	数据表的相关知识	67	5.2.2	设置行高与列宽	112
4.1.1	数据表相关概念	67	5.2.3	冻结/解冻列	112
4.1.2	表之间的关系	68	5.2.4	改变列的显示顺序	113
4.1.3	表的结构	69	5.2.5	设置数据表格式	114
4.1.4	数据类型	69	5.3	数据排序与筛选	115
4.1.5	字段属性	72	5.3.1	排序规则	115
4.2	创建 Access 数据表	76	5.3.2	数据排序	116
4.2.1	使用数据表视图创建表	77	5.3.3	数据筛选	119
4.2.2	使用设计视图创建表	78	5.4	对数据表中的行汇总统计	123
4.2.3	使用模板创建表	79	5.4.1	添加汇总行	123
4.2.4	通过导入或链接创建表	81	5.4.2	隐藏汇总行	124
4.2.5	创建查阅字段列	85	5.5	导出数据表	124
4.3	表的相关操作与修改	90	5.5.1	导出到文本文件	124
4.3.1	打开和关闭表	90	5.5.2	导出到 Excel 工作表	126
4.3.2	复制表	92	5.5.3	导出到 XML 文件	126
4.3.3	重命名表	93	5.6	上机练习	128
4.3.4	删除表	93	5.7	习题	129
4.3.5	修改表结构	94	第 6 章	查询	130
4.3.6	设置表的主键	96	6.1	查询概述	130
4.4	表之间的关系	97	6.1.1	查询与表的区别	131
4.4.1	建立表间的关系	97	6.1.2	查询的类型	131
4.4.2	关系选项	99	6.2	SQL 语言简介	132
4.4.3	编辑表间关系	100	6.2.1	SQL 概述	132
4.4.4	删除表间关系	100	6.2.2	使用 SELECT 语句	133
4.5	上机练习	101	6.2.3	高级查询语句	136
4.6	习题	103	6.3	使用查询向导创建查询	140
第 5 章	表中数据记录的操作	105	6.3.1	简单查询	140
5.1	数据的增删改查	105	6.3.2	交叉表查询	142

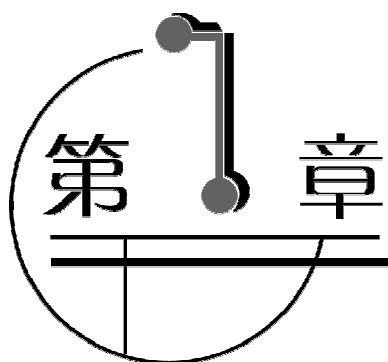


6.3.3	查找重复项查询	145	8.1.2	报表的视图	196
6.3.4	查找不匹配项查询	146	8.1.3	报表的分类	197
6.4	使用查询设计创建查询	148	8.2	创建报表	198
6.4.1	查询设计视图	148	8.2.1	创建基本报表	198
6.4.2	编辑查询	150	8.2.2	使用设计视图创建报表	199
6.4.3	查询条件的设置	152	8.2.3	使用空报表创建报表	200
6.4.4	参数查询	156	8.2.4	使用报表向导创建报表	201
6.4.5	操作查询	158	8.2.5	使用标签创建报表	203
6.4.6	SQL 查询	163	8.3	编辑与打印报表	205
6.5	上机练习	164	8.3.1	在报表中进行排序和分组	205
6.6	习题	165	8.3.2	在报表中添加分页符	207
第 7 章	窗体	166	8.3.3	在报表中添加页码	207
7.1	窗体概述	166	8.3.4	在报表中添加背景图片	208
7.1.1	窗体的功能	167	8.3.5	报表的预览与打印	208
7.1.2	窗体的分类	167	8.4	其他报表	211
7.2	创建窗体	168	8.4.1	创建子报表	211
7.2.1	快速创建窗体	169	8.4.2	交叉报表	213
7.2.2	窗体的视图	172	8.4.3	将窗体另存为报表	214
7.2.3	创建数据透视图窗体	173	8.5	上机练习	215
7.2.4	使用窗体向导创建窗体	175	8.6	习题	216
7.2.5	创建【空白】窗体	176	第 9 章	宏	217
7.3	设计窗体	177	9.1	认识宏	217
7.3.1	窗体设计视图的结构	177	9.1.1	什么是宏	217
7.3.2	【窗体设计工具】功能区 选项卡	179	9.1.2	宏的类型	218
7.3.3	使用窗体控件	180	9.1.3	宏的设计视图	219
7.3.4	编辑控件	183	9.2	创建和使用宏	220
7.3.5	弹出式窗体	185	9.2.1	创建简单宏	220
7.4	创建其他窗体	186	9.2.2	创建条件宏	223
7.4.1	创建主/子窗体	186	9.2.3	编辑宏	226
7.4.2	创建切换面板窗体	190	9.2.4	运行宏	227
7.5	上机练习	192	9.2.5	调试宏	229
7.6	习题	194	9.3	上机练习	230
第 8 章	报表	195	9.4	习题	233
8.1	报表概述	195	第 10 章	模块与 VBA	234
8.1.1	报表的功能	195	10.1	模块概述	234
			10.1.1	什么是 VBA	234





10.1.2	模块的分类	235	11.2.1	用户级安全机制	278
10.1.3	创建和运行模块	237	11.2.2	数据库加密与解密	278
10.2	VBA 程序设计基础	239	11.3	数据库的打包与签署	279
10.2.1	VBA 编程环境	239	11.3.1	获取数字证书	280
10.2.2	数据类型	244	11.3.2	应用数字签名	281
10.2.3	常量、变量和数组	245	11.4	上机练习	284
10.2.4	运算符与表达式	249	11.5	习题	284
10.2.5	VBA 常用语句	252	第 12 章	客户订单管理系统	285
10.2.6	面向对象程序设计	253	12.1	系统分析与设计	285
10.3	VBA 高级程序设计	255	12.1.1	需求分析	285
10.3.1	选择结构	256	12.1.2	功能设计	286
10.3.2	循环结构	259	12.1.3	数据库设计	286
10.3.3	跳转语句	262	12.2	系统实现	288
10.3.4	过程与函数	263	12.2.1	创建数据库	288
10.3.5	程序调试	267	12.2.2	创建数据表	288
10.3.6	VBA 代码的保护	270	12.2.3	创建查询	291
10.4	上机练习	271	12.2.4	创建窗体	293
10.5	习题	273	12.2.5	创建报表	297
第 11 章	数据库管理与安全	274	12.2.6	创建模块	299
11.1	数据库的压缩与备份	274	12.2.7	创建 AutoExec 宏	301
11.1.1	压缩和修复数据库	274	12.3	系统运行效果图	302
11.1.2	备份和恢复数据库	276	参考文献		304
11.2	Access 中的安全机制	277			



数据库系统概述

学习目标

数据库作为数据管理技术，是计算机科学的重要分支。在当今信息社会中，信息已经成为各行各业的重要财富和资源，对数据库的应用无处不在。因此，掌握数据库的基本知识及使用方法不仅是计算机科学与技术专业、信息管理专业学生的基本技能，也是非计算机专业学生应该具备的技能。本章主要介绍数据库系统的基本概念，包括数据库系统常用术语、数据库管理系统、安全性和完整性、关系数据模型、关系运算、关系模式的规范化理论，以及数据库设计的方法与步骤等。

本章重点

- 数据与数据处理
- 数据库与数据库系统
- 关系数据模型
- 关系代数
- 规范化理论
- 数据库设计的方法和步骤

1.1 数据库相关的概念

数据库技术是信息系统的核心和基础，它提供了最基本、最准确、最全面的信息资源，对这些资源的管理和应用，已成为人们科学决策的依据。数据库应用已遍及人们生活中的各个角落，如铁路及航空公司的售票系统、图书馆的图书借阅系统、学校的教学管理系统、超市售货系统和银行的业务系统等。数据库与人们的生活已经是密不可分，几乎每个人的生活都离不开数据库。对于一个国家来说，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这个国家信息化发达程度的重要标志之一，而信息化对于加快国家产业结构调整、促进经济增



长和提高人们生活质量具有明显的倍增效应和带动作用。

1.1.1 数据与数据处理

人们在现实中进行的各种活动都会产生相应的信息，例如，生产塑料餐具的工厂，其用于生产的原材料的名称、库存量、单价、产地；生产出来的产品的名称、数量、单价；该工厂中职工的职称、编号、薪水、奖金等，所有这些都是信息，这些信息代表了所属实体的特定属性或状态，当把这些信息以文字记录下来时便形成数据，因此可以说，数据就是信息的载体。本节主要介绍信息、数据和数据处理的概念。

1. 信息与数据

信息与数据是两个密切相关的概念，信息是各种数据所包含的意义，数据则是负载信息的物理符号。如某个人的年龄，某个考生的考试成绩，某年度的国民总收入等，这些都是信息。如果将这些信息用文字或其他符号记录下来，那么，这些文字或符号就是数据。同一数据在不同的场合具有不同的意义，例如，56这个数字，既可以表示一个人的年龄，也可以表示水的温度，或者表示某个考生某科目的考试成绩。在许多场合下，对信息和数据的概念并不做严格的区分，可互换使用，例如，通常所说的“信息处理”和“数据处理”，这两个概念的意义是相同的。

信息是对现实世界事物存在方式或运动状态的反映。它已成为人类社会活动的一种重要资源，与能源、物质并称为人类社会活动的三大要素。一般来说，信息是一种被加工成为特定形式的数据，这种数据形式对接收者来说是有意义的，而且对当前和将来的决策具有明显的或实际的价值。它具有如下特征：

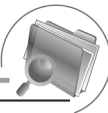
- 信息可以被感知，不同的信息源有不同的感知方式。
- 信息的获取和传递不仅需要载体，而且还消耗能量。
- 信息可以通过载体进行存储、压缩、加工、传递、共享、扩散、再生和增值等。

数据是将现实世界中的各种信息记录下来的、可以识别的符号。它是信息的载体，是信息的具体表现形式。在计算机内部，所有的数据均采用0和1进行编码。在数据库技术中，数据的含义很广泛，除了数字之外，文字、图形、图像、声音和视频等也视为数据，它们分别表示不同类型的信息。

另外，同一种信息可以用多种不同的数据形式进行表达，而信息的意义不随数据的表现形式的改变而改变。例如，要表示某支股票每天的收盘价格，既可以通过绘制曲线图表示，也可以通过绘制柱状图表示，还可以通过表格数据进行表示，而无论使用何种方式来表示，丝毫不会改变信息的含义。

例如，对数据可以作如下定义：描述事物的符号记录称为数据。在学校的学生档案中，可以记录学生的姓名、性别、出生日期、所在系、电话号码和入学时间等。按这个次序排列组合成如下所示的一条记录：





(赵智轩, 男, 1981-04-11, 心理系, 15910806516, 1998)

这条记录中的信息就是数据。当然数据可能会因为记录介质被破坏而丢失,如记录在纸上的数据,可能因为纸介质丢失、火灾而造成数据丢失;记录在计算机磁盘上的数据,可能因为病毒、误操作、火灾等造成数据丢失。

2. 数据与信息的关系

数据与信息有着不可分割的联系。信息是处理加工过的数据,数据和信息的关系是一种原料和成品之间的关系,如图 1-1 所示。

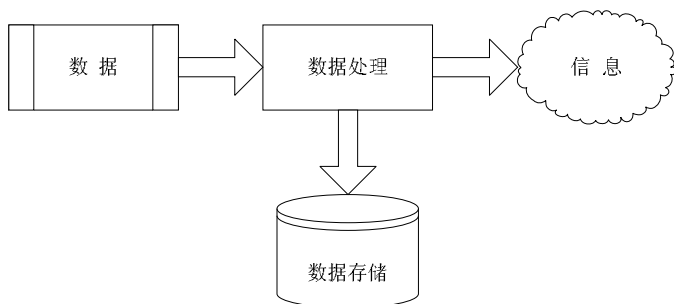


图 1-1 数据与信息的关系

- (1) 数据是信息的符号表示,或称载体。
- (2) 信息是数据的内涵,是数据的语义解释。
- (3) 数据是符号化的信息。
- (4) 信息是语义化的数据。

3. 数据处理

数据处理是指对各种形式的数据进行收集、存储、加工和传播的一系列活动的总和。

进行数据处理的目的是有两个,一是从大量的、原始的数据中抽取、整理出对人们有价值的信息,以作为行动和决策的依据;二是为了借助计算机科学地保存和管理复杂的、大量的数据,以便人们能够方便而充分地利用这些宝贵的资源。

1.1.2 数据库

数据库(Database, DB)就是数据的集合。例如,日常生活中,我们用笔记本记录亲朋好友的联系方式,将他们的姓名、地址、电话等信息都记录下来。这个“通讯录”就是一个最简单的“数据库”,每个人的姓名、地址、电话等信息就是这个数据库中的“数据”。我们可以在笔记本这个“数据库”中添加新朋友的个人信息,由于某个朋友的电话变动也可以修改他的电话号码这个“数据”。使用笔记本这个“数据库”可以方便地查到某位亲朋好友的地址、邮编或电话号码这些“数据”。





数据库是组织数据的一种手段，按照字面的理解，数据库就是存放数据的仓库。它是为了实现一定的目的按某种规则组织起来的“数据”的“集合”。在信息社会中，数据库的应用非常广泛，如银行业用数据库存储客户的信息、账户、贷款以及银行的交易记录；学校里用数据库存储学生的个人信息、课程成绩等。

在计算机领域，数据库是指长期存储在计算机内的、有组织的、可共享的、统一管理的相关数据的集合。

数据库中的数据不仅需要合理地存放，还要便于查找；数据库不仅可以供创建者本人使用，还可以供多个用户从不同的角度共享，即多个不同的用户可以根据不同的需求，使用不同的语言，同时存取数据库，甚至同时存取同一数据。

1.1.3 数据库系统

数据库系统是计算机化的记录保持系统，它的目的是存储和产生所需要的有用信息。这些有用的信息可以是使用该系统的个人或组织的有意义的任何事情，是对某个人或组织辅助决策过程中不可少的事情。

1. 数据库系统的组成

通常，一个数据库系统要包括以下 4 个主要部分：数据、用户、硬件和软件。

(1) 数据

数据是数据库系统的工作对象。为了区别输入、输出或中间数据，常把数据库数据称为存储数据、工作数据或操作数据。它们是某特定应用环境中进行管理和决策所必需的信息。特定的应用环境，可以指一个公司、一个银行、一所医院和一个学校等。在这些应用环境中，各种不同的应用可通过访问其数据库获得必要的信息以辅助进行决策，决策完成后，再将决策结果存储在数据库中。

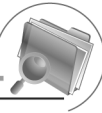
数据库中的存储数据是“集成的”和“共享的”。“集成”是指把某特定应用环境中的各种应用关联的数据及其数据间的联系全部集中地按照一定的结构形式进行存储，也就是把数据库看成若干个性质的数据文件的联合和统一的数据整体，并且在文件之间局部或全部消除了冗余，这使得数据库系统具有整体数据结构化和数据冗余小的特点；“共享”是指数据库中的一块块数据可为多个不同的用户所共享，即多个不同的用户，使用多种不同的语言，为了不同的应用目的，而同时存取数据库，甚至同时存取同一数据块。共享实际上是基于数据库集成的。

(2) 用户

用户是指存储、维护和检索数据库中数据的人员。数据库系统中主要有 3 类用户：终端用户、应用程序员和数据库管理员。

- 终端用户：也称为最终用户，是指从计算机联机终端存储数据库的人员，也可以成为联机用户。这类用户使用数据库系统提供的终端命令语言、表格语言或菜单驱动等交





交互式方式来存取数据库中的数据。终端用户一般是不精通计算机和程序设计的各级管理人员、工程技术人员和各类科研人员。

- 应用程序员：也称为系统开发员，是指负责设计和编制应用程序的人员。这类用户通过设计和编写“使用及维护”数据库的应用程序来存取和维护数据库。这类用户通常使用 Access、SQL Server 或 Oracle 等数据库语言来设计和编写应用程序，以对数据库进行存取操作。
- 数据库管理员(DBA)：是指全面负责数据库系统的“管理、维护和正常使用的”人员，可以是一个人或一组人。而对于大型数据库系统来说，DBA 极为重要，通常设置有 DBA 办公室，应用程序员是 DBA 手下的工作人员。DBA 不仅要具有较高的技术专长，而且还要具备较深的资历，并具有了解和阐明管理要求的能力。DBA 的主要职责包括参与数据库设计的全过程；与用户、应用程序员、系统分析员紧密结合，设计数据库的结构和内容；决定数据库的存储和存取策略，使数据的存储空间利用率和存取效率均较优；定义数据的安全性和完整性；监督控制数据库的使用和运行，及时处理运行程序中出现的各种问题；改进和重新构建数据库系统等。

(3) 硬件

硬件是指存储数据库和运行数据库管理系统 DBMS 的硬件资源，包括物理存储数据库的磁盘、磁鼓、磁带或其他外存储器及其附属设备、控制器、I/O 通道、内存、CPU 以及外部设备等。数据库服务器的处理能力、存储能力、可靠性直接关系到整个系统的性能优劣，因此对服务器端硬件资源也有着较高的要求，应选用高可靠性、高可用性、高性价比的服务器。通常要求考虑以下问题。

- 具有足够大的内存，用于存放操作系统、DBMS 的核心模块、数据缓冲区和应用程序。
- 具有高速大容量的直接存取设备。一般数据库系统的数据量和数据的访问量都很大，因此需要容量大、速度快的存储系统存放数据，如采用高速大缓存硬盘，或者应用光纤通道外接到外置的专用磁盘系统。
- 具有高速度 CPU，以拥有较短的系统响应时间。数据库服务器必须应对大量的查询并作出适当且及时的应答，因此要求处理能力强的 CPU 以满足较高的服务器处理速度和对客户的响应速率的要求。
- 有较高的数据传输能力，以提高数据传输率，保证足够的系统吞吐能力，否则，系统性能将形成瓶颈。
- 有足够的外存来进行数据备份。常配备磁盘阵列、磁带机或光盘机等存储设备。
- 高稳定性的系统。即数据库系统能够稳定持续运行，能提供长时间可靠稳定的服务。

(4) 软件

软件是指负责数据库存取、维护和管理的软件系统，通常叫做数据库管理系统(Database Management System, DBMS)。数据库系统的各类用户对数据库的各种操作请求，都是由 DBMS 来完成的，它是数据库系统的核心软件。DBMS 提供一种超出硬件层之上的对数据库管理的功能，使数据库用户不受硬件层细节的影响。DBMS 是在操作系统支持下工作的。





2. 数据库系统的特点

数据库系统具有如下特点。

(1) 数据低冗余、共享性高

数据不再是面向某个应用程序，而是面向整个系统。当前所有用户可同时存取库中的数据，从而减少了数据冗余，节约存储空间，同时也避免了数据之间的不相容性和不一致性。

(2) 数据独立性提高

数据的独立性包括逻辑独立性和物理独立性。

- 数据的逻辑独立性是指当数据的总体逻辑结构改变时，数据的局部逻辑结构不变，由于应用程序是依据数据的局部逻辑结构编写的，所以，应用程序可不必修改，从而保证了数据与程序间的逻辑独立性。例如，在原有的记录类型之间增加新的联系，或在某些记录类型中增加新的数据项时，均可确保数据的逻辑独立性。
- 数据的物理独立性是指当数据的存储结构改变时，数据的逻辑结构不变，从而应用程序也不必改变。例如，改变存储设备和增加新的存储设备，或改变数据的存储组织方式，均可确保数据的物理独立性。

(3) 有统一的数据控制功能

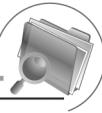
数据库可以被多个用户所共享，当多个用户同时存取数据库中的数据时，为保证数据库中数据的正确性和有效性，数据库系统提供了以下 4 个方面的数据控制功能。

- 数据的安全性(security)控制：可防止不合法使用数据造成数据的泄漏和破坏，保证数据的安全和机密。例如，系统提供口令检查或其他手段来验证用户身份，以防止非法用户使用系统；也可以对数据的存取权限进行限制，只有通过检查后才能执行相应的操作。
- 数据完整性(integrity)控制：系统通过设置一些完整性规则以确保数据的正确性、有效性和相容性。正确性是指数据的合法性，如代表年龄的整型数据，只能包含 0~9，不能包含字母或特殊符号；有效性是指数据是否在其定义的有效范围内，如月份只能用 1~12 之间的数字来表示；相容性是指表示同一事实的两个数据应相同，否则就不相容，例如，一个人的性别不能既是男又是女。
- 并发(concurrency)控制：多用户同时存取或修改数据库时，防止相互干扰而提供给用户不正确的数据，并使数据库受到破坏。
- 数据恢复(recovery)：当数据库被破坏或数据不可靠时，系统有能力将数据库从错误状态恢复到最近某一时刻的正确状态。

3. 数据库系统的发展史

从最早的商用计算机起，数据处理就一直推动着计算机的发展。事实上，数据处理自动化早于计算机出现。Hollerith 发明的穿孔卡片，早在 20 世纪初就用来记录美国的人口普查数据，用机械系统来处理这些卡片并列结果。穿孔卡片后来被广泛应用作为将数据输入计算机的一种手段。





按照年代来划分, 数据库系统的发展可划分为以下几个阶段。

(1) 20 世纪 50 年代至 60 年代早期

20 世纪 50 年代至 60 年代早期, 磁带被用于数据存储。诸如工资单这样的数据处理已经自动化了, 并且把数据存储存储在磁带上。数据处理包括从一个或多个磁盘上读取数据, 并将数据写回到新的磁带上。数据也可以由一叠穿孔卡片输入, 而输出到打印机上。例如, 工资增长的处理是通过将增长表示到穿孔卡片上, 在读入一叠穿孔卡片的同时要配有保存主要工资细节的磁带。工资的增加额将被加入到从主磁带读出的工资中, 并被写到新的磁带上, 新磁带将成为新的主磁带。

磁带(和卡片)都只能顺序读取, 并且数据可以比内存大得多, 因此, 数据处理程序被迫用一种特定的顺序对来自磁带和卡片的数据进行读取和合并处理。

(2) 20 世纪 60 年代末至 20 世纪 70 年代

20 世纪 60 年代末硬盘的广泛使用极大地改变了数据处理的情况, 因为硬盘可以直接对数据进行访问。磁盘上数据的位置是无意义的, 因为磁盘上的任何位置都可在几十毫秒内访问到, 数据由此摆脱了顺序读取的限制。有了磁盘, 就可以创建网状数据库和层次数据库, 它们可以具有保存在磁盘上的如表和树等数据结构。程序员也可以创建和操作这些数据结构。

由 Codd 写的一篇具有里程碑意义的论文, 定义了关系模型和在关系模型中用非过程化的方法来查询数据, 关系数据库由此诞生了。关系模型的简单性和能够对程序员隐藏所有细节的能力具有真正的诱惑力。

(3) 20 世纪 80 年代

尽管关系模型在学术上很受重视, 但是最初并没有实际的应用, 因为它在性能上的不足, 关系型数据库在性能上还不能和当时已有的网状和层次数据库相提并论。这种情况直到 System R 的出现才得以改变, IBM 研究院的一个突破性项目开发了一种能够构造高效的关系型数据库系统的技术。Astrahan 和 Chamberlin 等人提供了很好的关于 System R 的综述。具有完全功能的 System R 原型诞生了 IBM 的第一个关系数据库产品 SQL/DS。最初的商用关系数据库系统, 例如 IBM 的 DB2、Oracle、Ingres 和 DEC 的 Rdb, 在推动有效的处理陈述式查询技术上起到了主要作用。到了 20 世纪 80 年代早期, 关系数据库已经可以在性能上和网状、层次数据库进行竞争了。关系数据库是如此简单易用, 以至于最后它完全取代了网状和层次数据库。因为程序员在使用后者时, 必须处理许多底层的实现问题, 并且不得不将要做的查询任务编码成过程化的形式。更重要的是, 在设计应用程序时还要时时考虑效率问题, 而这需要付出很大的努力。相反, 在关系数据库中, 几乎所有的底层工作都由数据库自动来完成, 使得程序员可以只考虑逻辑层的工作。因为关系模型在 20 世纪 80 年代已经取得了优势, 所以它在数据模型中具有最高的统治地位。

另外, 在 20 世纪 80 年代人们还对并行和分布式数据库进行了很多研究, 同样在面向对象数据库方面也展开了初步的工作。

(4) 20 世纪 90 年代初

SQL 语言主要是为了决策支持应用设计的, 重在查询; 而 20 世纪 80 年代主要的数据库是





处理事务的应用，重在更新。决策支持和查询再度成为数据库的一个主要应用领域。分析大量数据的工具有了很大的发展。

在这个时期许多数据库厂商推出了并行数据库产品。数据库厂商还开始在其数据库中加入对象-关系的支持。

(5) 20 世纪 90 年代末至今

随着互联网的兴起和发展，数据库比以前有了更加广泛的应用。现在数据库系统必须支持很高的事务处理速度，而且还要有很高的可靠性和 24×7 的可用性(一天 24 小时，一周 7 天都可用，也就是没有进行维护的停机时间)。数据库系统还必须支持网络接口。

1.1.4 数据库管理系统(DBMS)

数据库管理系统是位于用户和数据库之间的一个数据管理软件，它的主要任务是对数据库的建立、运用和维护进行统一管理、统一控制，即用户不能直接接触数据库，而只能通过 DBMS 来操纵数据库。

1. DBMS 概述

数据库管理系统负责对数据库的存储进行管理、维护和使用，因此，DBMS 是一种非常复杂的、综合性的、在数据库系统中对数据进行管理的大型系统软件，它是数据库系统的核心组成部分，在操作系统(OS)支持下工作。在确保数据安全可靠的同时，DBMS 大大提高了用户使用“数据”的简明性和方便性，用户在数据库系统中的一切操作，包括数据定义、查询、更新及各种操作，都是通过 DBMS 完成的。

DBMS 是数据库系统的核心部分，它把所有应用程序中使用的数据汇集在一起，并以记录为单位存储起来，便于应用程序查询和使用，如图 1-2 所示。

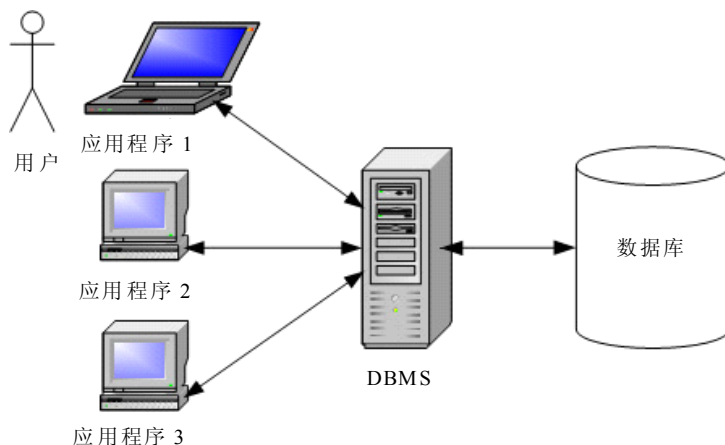
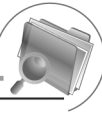


图 1-2 DBMS、数据库以及与用户之间的关系

常见的 DBMS 有 Access、Oracle、SQL Server、DB2、Sybase 和 FoxPro 等。不同的数据库





管理系统有不同的特点。Access 相对于其他的一些数据库管理软件，如 SQL Server、Oracle 等来说，操作相对简单，不需要用户具有高深的数据库知识，就能完成数据库所有的构造、检索、维护等功能，并且 Access 拥有简捷、美观的操作界面。

Access 属于小型桌面数据库管理系统，通常用于办公管理。它允许用户通过构建应用程序来收集数据，并可以通过多种方式对数据进行分类、筛选，将符合要求的数据供用户查看，用户可以通过显示在屏幕上的窗体来查看数据库中的数据，也可以通过报表将相关的数据打印出来，以便更详细地进行研究。

2. DBMS 的功能

DBMS 由于缺乏统一的标准，其性能、功能等许多方面随系统而异。一般情况下，大型系统功能比较强，小型系统功能较弱，同一类系统，性能也是有差异的。通常情况下，DBMS 提供了以下几个方面的功能。

- **数据库定义功能：**DBMS 提供相应数据定义语言定义数据库结构，刻画数据库的框架，并被保存在数据字典中。数据字典是 DBMS 存取和管理数据的基本依据。
- **数据存取功能：**DBMS 提供数据操纵语言实现对数据库数据的检索、插入、修改和删除等基本存取操作。
- **数据库运行管理功能：**DBMS 提供数据控制功能，即数据的安全性、完整性和并发控制等，对数据库运行进行有效的控制和管理，以确保数据库数据正确有效和数据库系统的有效运行。
- **数据库的建立和维护功能：**包括数据库初始数据的装入，数据库的转储、恢复、重组、系统性能监视、分析等功能。这些功能大都由 DBMS 的实用程序来完成。
- **数据通信功能：**DBMS 提供处理数据的传输功能，实现用户程序与 DBMS 之间的通信，这通常与操作系统协调完成。

3. DBMS 的组成

DBMS 大多是由许多系统程序所组成的一个集合。每个程序都有各自的功能，一个或几个程序一起协调完成 DBMS 的一件或几件工作任务。各种 DBMS 的组成因系统而异，一般来说，它由以下几个部分组成。

- **语言编译处理程序：**语言编译处理程序主要包括数据描述语言翻译程序、数据操作语言处理程序、终端命令解释程序、数据库控制命令解释程序等。
- **系统运行控制程序：**主要包括系统总控程序、存取控制程序、并发控制程序、完整性控制程序、保密性控制程序、数据存取与更新程序和通信控制程序等。
- **系统建立、维护程序：**主要包括数据装入程序、数据库重组程序、数据库系统恢复程序和性能监督程序等。
- **数据字典：**数据字典通常是一系列表，它存储着数据库中有关信息的当前描述。它能帮助用户、数据库管理员和数据库管理系统本身使用和管理数据库。





1.1.5 数据库应用系统(DBAS)

数据库应用系统(Database Application System, 简称 DBAS)是指在 DBMS 的基础上, 针对一个实际问题开发出来的面向用户的系统。如网上银行就是一个数据库应用系统, 用户通过登录网上银行, 可以查询自己的账户余额, 还可以进行转账汇款等操作。

1.2 数据库系统的体系结构

从数据库管理系统的角度看, 数据库系统通常采用三级模式结构, 这是数据库系统内部的体系结构; 从数据库最终用户的角度看, 数据库系统的结构分为集中式结构、文件服务器结构和客户/服务器结构, 这是数据库系统外部的体系结构。数据库系统的体系结构可分为内部体系结构和外部体系结构。



1.2.1 内部体系结构

根据美国国家标准协会和标准规划与需求委员会(ANSI/SPARC)提出的建议, 数据库系统的内部体系结构是三级模式结构, 分别为模式、外模式和内模式, 如图 1-3 所示。

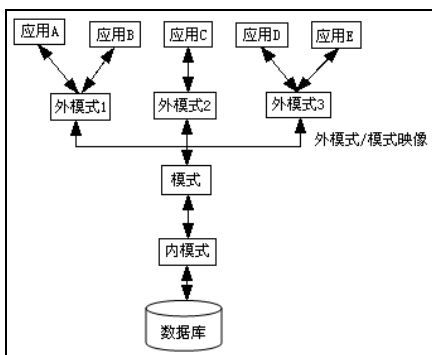
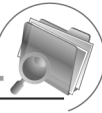


图 1-3 数据库系统的三级模式结构

1. 模式

模式也称为概念模式, 是数据库中全体数据的逻辑结构和特征的描述, 是所有用户的公共数据视图。它是数据库系统模式结构的中间层, 既不涉及数据的物理存储细节和硬件环境, 也与具体的应用程序、所使用的应用开发工具及高级程序设计语言无关。

模式实际上是数据库数据在逻辑级上的视图。一个数据库只有一个模式。数据库模式以某种数据模型为基础, 统一综合地考虑了所有用户的需求, 并将这些需求有机地结合成一个逻辑整体。定义模式时不仅要定义数据的逻辑结构, 例如, 数据记录由哪些数据项构成, 数据项的



名字、类型和取值范围等，而且要定义数据之间的联系，定义与数据有关的安全性、完整性要求。DBMS 提供模式描述语言来严格地定义模式。

2. 外模式

外模式也称子模式或用户模式，它是数据库用户(包括应用程序员和最终用户)看见和使用的局部数据的逻辑结构和特征的描述，是数据库用户的数据视图，是与某一应用有关的数据的逻辑表示。

外模式通常是模式的子集。一个数据库可以有多个外模式。由于它是各个用户的数据视图，如果不同的用户在应用需求、看待数据的方式、对数据保密的要求等方面存在差异，则其外模式描述就是不同的。即使是模式中同一数据，在外模式中的结构、类型、长度、保密级别等都可以不同。另一方面，同一外模式也可以为某一用户的多个应用系统所使用，但一个应用程序只能使用一个外模式。

外模式是保证数据库安全性的一个有力措施。每个用户只能看见和访问所对应的外模式中的数据，数据库中的其余数据是不可见的。

3. 内模式

内模式也称存储模式，一个数据库只有一个内模式。它是数据物理结构和存储方式的描述，是数据在数据库内部的表示方式。例如，记录的存储方式是堆存储，还是按照某个属性值的升(降)序存储，还是按照属性值聚簇存储；索引按照什么方式组织，是 B+树索引还是 hash 索引；数据是否压缩存储，是否加密；数据的存储记录结构有何规定，如定长结构或变长结构，一个记录不能跨物理页存储等。DBMS 提供内模式描述语言来严格地定义内模式。

数据库系统的三级模式是对数据的 3 个抽象级别，它把数据的具体组织留给 DBMS 管理，使用户能逻辑地、抽象地处理数据，而不必关心数据在计算机中的具体表示方式与存储方式。为了能够在系统内部实现这 3 个抽象层次的联系和转换，数据库管理系统在这三级模式之间提供了以下两层映射，正是这两层映射保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

- 外模式/模式映射：它定义了外模式和模式之间的对应关系。当模式改变时，由数据库管理员对各个外模式/模式的映像做相应改变，可以使外模式保持不变，从而应用程序不必修改，以保证数据的逻辑独立性。
- 模式/内模式映射：模式/内模式映像是唯一的，它定义了数据全局逻辑结构和存储结构之间的对应关系。当数据库的存储结构改变时，由数据库管理员对模式/内模式映像作相应的改变，可以使模式保持不变，从而保证了数据的物理独立性。

1 2.2 外部体系结构

外部体系结构主要有集中式结构、文件服务器结构和客户/服务器结构。





1. 集中式结构

集中式数据库结构由两个关键硬件组成：主机和客户终端。

数据库和应用程序存放在主机中，数据的处理和主要的运算操作也在主机上进行。它的主要特点是数据和应用集中，维护方便，安全性好；但对主机性能要求较高，价格昂贵。

2. 文件服务器结构

在文件服务器结构中，数据库存放在文件服务器中，应用程序分散安排在各个客户工作站上。文件服务器只负责文件的集中管理，所有的应用处理安排在客户端完成。文件服务器结构的特点是费用低，配置灵活，但是缺乏足够的计算和处理能力，对客户端的计算机性能要求高。Access 和 Visual FoxPro 支持文件服务器方案。

3. 客户/服务器结构

在客户/服务器结构中，数据库存放在服务器中，应用程序可以根据需要安排在服务器或客户工作站上，实现了客户端程序和服务器端程序的协同工作。这种结构解决了集中式结构和文件服务器结构的费用和性能问题。SQL Server 和 Oracle 都支持客户/服务器结构。



1.3 数据模型

计算机信息管理的对象是现实生活中的客观事物，但这些事物是无法直接送入计算机的，必须通过进一步整理和归类，进行信息的规范化，然后才能将规范信息数据化并送入计算机的数据库中保存起来。这一过程经历了 3 个领域——现实世界、信息世界和数据(机器)世界。

- 现实世界：存在于人脑之外的客观世界，包括事物及事物之间的联系。
- 信息世界：是现实世界在人们头脑中的反映。
- 数据(机器)世界：将信息世界中的实体进行数据化，事物及事物之间的联系用数据模型来描述。

在现实世界中，常常用模型来对某个对象进行抽象或描述，如飞机模型，它反应了该飞机的大小、外貌特征及其型号等；并可用文字语言来对该对象进行抽象或描述。

为了用计算机来处理现实世界的事物，首先需要将它们反映到人的大脑中，即首先需要把这些事务抽象为一种既不依赖于某一具体的计算机又不受某一具体 DBMS 所左右的信息世界的概念模型，然后再将该概念模型转换为某一具体 DBMS 所支持的计算机世界的数据库模型。

信息的 3 个世界及其关系如图 1-4 所示。

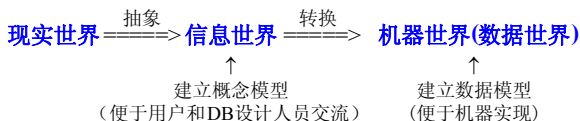
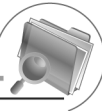


图 1-4 信息的 3 个世界及其关系



这个过程是通过研究“过程 and 对象”，然后建立相应的数据模型来实现的。在这两个转换过程中，需要建立两个模型即概念模型和逻辑数据模型。

1 3.1 概念模型

概念模型是对客观事物及其联系的抽象，用于信息世界的建模，它强调其语义表达能力，以及能够较方便、直接地表达应用中的各种语义知识。这类模型简单、清晰、易于被用户理解，是用户和数据库设计人员之间进行交流的语言。这种信息结构并不依赖于具体的计算机系统，不是某一个 DBMS 支持的数据模型，而是概念级的模型。

概念模型主要用来描述世界的概念化结构，它使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，集中精力分析数据以及数据之间的联系等，与具体的数据管理系统无关。概念数据模型必须换成逻辑数据模型，才能在 DBMS 中实现。

在概念模型中主要有以下几个基本术语。

1. 实体与实体集

实体是现实世界中可区别于其他对象的“事件”或物体。实体可以是人，也可以是物；可以指实际的对象，也可以指某些概念；还可以指事物与事物间的联系。例如，学生就是一个实体。

实体集是具有相同类型及共享相同性质(属性)的实体集合。如全班学生就是一个实体集。实体集不必互不相交，例如，可以定义学校所有学生的实体集 student 和所有教师的实体集 teacher，而一个 person 实体可以是 student 实体，也可以是 teacher 实体，甚至可能既是 student 实体又是 teacher 实体，也可以都不是。

2. 属性

实体通过一组属性来描述。属性是实体集中每个成员所具有的描述性性质。将一个属性赋予某实体集表明数据库为实体集中每个实体存储相似信息，但每个实体在自己的每个属性上都有各自的值。一个实体可以由若干个属性来刻画。如学生实体有学号、姓名、年龄、性别和班级等属性。

每个实体的每个属性都有一个值，例如，某个特定的 student 实体，其学号是 2009228649，姓名是葛萌萌，年龄是 27，性别是女。

3. 关键字和域

实体的某一属性或属性组合，其值能唯一标识出某一实体，称为关键字，也称码。如学号是学生实体集的关键字，由于姓名有相同的可能，故不应作为关键字。

每个属性都有一个可取值的集合，称为该属性的域，或者该属性的值集。如姓名的域为字符串集合，性别的域为“男”和“女”。





4. 联系

现实世界的事物之间总是存在某种联系,这种联系必然要在信息世界中加以反映。一般存在两种类型的联系:一是实体内部的联系,如组成实体的属性之间的联系;二是实体与实体之间的联系。

两个实体之间的联系又可以分为如下 3 类。

- 一对一联系(1:1): 例如,一个班级有一个班主任,而每个班主任只能在一个班任职。这样班级和班主任之间就具有一对一的联系。
- 一对多联系(1:n): 例如,一个班有多个学生,而每个学生只可以属于一个班,因此,在班级和学生之间就形成了一对多的联系。
- 多对多的联系(m:n): 例如,学校中的课程与学生之间就存在着多对多的联系。每个课程可以供很多学生选修,而每个学生也可以选修多门课程。这种关系可以有很多种处理方法。



1.3.2 用 E-R 方法表示概念模型

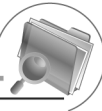
概念模型的代表方法很多,其中最著名的是 E-R 方法(Entity-Relations, 实体-联系方法),它用 E-R 图来描述现实世界的概念模型。E-R 图的主要成分是实体、联系和属性。E-R 图通用的表现规则如下。

- 矩形: 表示实体集。
- 椭圆: 表示属性。
- 菱形: 用菱形表示实体间的联系,菱形框内写上联系名。用无向边分别把菱形与有关实体相连接,在无向边旁标上联系的类型。如果实体之间的联系也具有属性,则把属性和菱形也用无向边连上。
- 线段: 将属性连接到实体集或将实体集连接到联系集。
- 双椭圆: 表示多值属性。
- 虚椭圆: 表示派生属性。
- 双线: 表示一个实体全部参与到联系集中。
- 双矩形: 表示弱实体集。

E-R 方法是抽象和描述现实世界的有力工具。用 E-R 图表示的概念模型与具体的 DBMS 所支持的数据模型独立,是各种数据模型的共同基础,因而比数据模型更一般、更抽象、更接近现实世界。

例如,要画出某个学校学生选课系统的 E-R 图,学校每学期开设若干课程供学生选择,每门课程可接受多个学生选修,每个学生可以选修多门课程,每门课程有一个教师讲授,每个教师可以讲授多门课程。

首先,确定实体集和联系。在本例中,可以将课程、学生和教师定义为实体,学生和课程



之间是“选修”关系，课程和教师之间是“教授”关系。

接着，确定每个实体集的属性：“学生”实体的属性有学号、姓名、班级和性别；“课程”实体的属性有课程号、课程名和教科书；“教师”实体的属性有职工号、姓名和性别。在联系中反映出教师讲授的课程信息、每门课程上课的学生数以及学生选修的所有课程。最终得到的E-R图如图1-5所示。

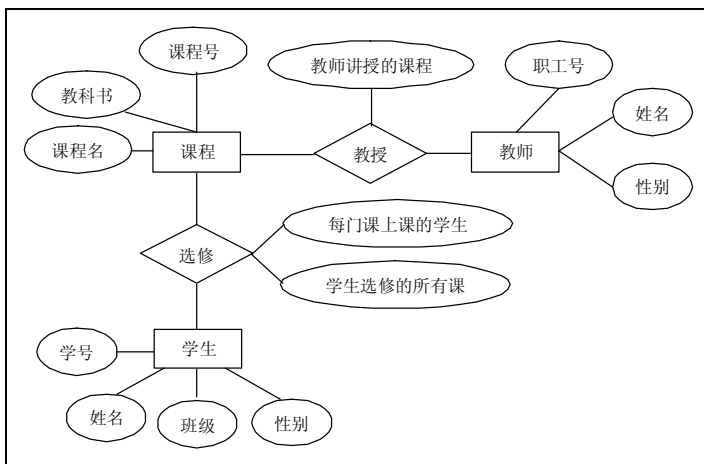


图 1-5 学生选课系统的 E-R 图

1.3.3 逻辑数据模型

数据库中的数据是结构化的，这是按某种数据模型来组织的。当前流行的基本数据模型有 3 类：关系模型、层次模型和网状模型。它们之间的根本区别在于数据之间联系的表示方式不同。关系模型用二维表来表示数据之间的联系；层次模型用树结构来表示数据之间的联系；网状模型用图结构来表示数据之间的联系。

层次模型和网状模型是早期的数据模型。通常把它们统称为格式化数据模型，因为它们是属于以“图论”为基础的表达方法。

按照这 3 类数据模型设计和实现的 DBMS 分别称为关系 DBMS、层次 DBMS 和网状 DBMS。相应地存在有关系(数据库)系统、层次(数据库)系统和网状(数据库)系统等简称。下面分别对这 3 种数据模型做一个简单的介绍。

1. 层次模型

层次数据模型是数据库系统最早使用的一种模型，它的数据结构是一颗有向树。层次结构模型具有如下特征：

- 有且仅有一个结点没有双亲，该结点是根结点。
- 其他结点有且仅有一个双亲。

在层次模型中，每个结点描述一个实体型，称为记录类型。一个记录类型可有許多记录值，





简称记录。结点间的有向边表示记录间的联系。如果要存取某一记录类型的记录，可以从根结点起，按照有向树层次逐层向下查找。查找路径就是存取路径。

层次模型结构清晰，各结点之间联系简单，只要知道每个结点的(除根结点以外)双亲结点，就可以得到整个模型结构，因此，画层次模型时可用无向边代替有向边。用层次模型模拟现实世界的层次结构的事物及其之间的联系是很自然的选择方式。比如表示“行政层次结构”、“家族关系”等是很方便的。

层次模型的缺点是结构呆板，缺乏灵活性；同一属性数据要存储多次，数据冗余大；不适合于拓扑空间数据的组织；不能表示两个以上实体型之间的复杂联系和实体型之间的多对多联系。

美国 IBM 公司 1968 年研制成功的 IMS 数据库管理系统就是这种模型的典型代表。

2. 网状模型

如果取消层次模型的两个限制，即两个或两个以上的结点都可以有多个双亲，则“有向树”就变成了“有向图”。“有向图”结构描述了网状模型。网状模型具有如下特征：

- 可有一个以上的结点没有双亲。
- 至少有一个结点可以有多个双亲。

网状模型和层次模型在本质上是一样的。从逻辑上看，它们都是基本层次联系的集合，用结点表示实体，用有向边(箭头)表示实体间的联系；从物理上看，它们每一个节点都是一个存储记录，用链接指针来实现记录间的联系。当存储数据时这些指针就固定下来了，数据检索时必须考虑存取路径问题；数据更新时，涉及链接指针的调整，缺乏灵活性；系统扩充相当麻烦。网状模型中的指针更多，纵横交错，从而使数据结构更加复杂。

3. 关系模型

关系模型(Relational Model)是用二维表格结构来表示实体及实体之间联系的数据模型。关系模型的数据结构是一个“二维表框架”组成的集合，每个二维表又可称为关系，因此可以说，关系模型是“关系框架”组成的集合。

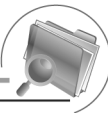
关系模型是使用最广泛的数据模型，目前大多数数据库管理系统都是关系型的，本书要介绍的 Access 就是一种关系数据库管理系统。

例如：对于某校学生、课程和成绩的管理，要用到如表 1-1 至表 1-3 所示的几个表格。如果要找到学生“潘涌纸”的“高等数学”成绩，首先需在学生信息表中找到“姓名”为“潘涌纸”的记录，记下他的学号 944114044，如表 1-1 所示。

表 1-1 学生信息表

学号	姓名	性别	年龄	院系ID	联系电话
982111056	葛冰	女	26	9001	13831705678
981000021	洪袖伽	女	33	9002	13910022348
944114044	潘涌纸	男	38	7012	13901040948





(续表)

学号	姓名	性别	年龄	院系ID	联系电话
100020202	李二勇	女	21	801	
991000022	赵智轩	男	21	9002	13722719080
201231008	栾鹏	男	30	7012	13681187162

再到课程表中找到“课程名称”为“高等数学”的课程号“81002”，如表 1-2 所示。

表 1-2 课程表

课程号	课程名称	学分	教师ID
1001	经济学原理	3	91001
1002	变态心理学	4	61001
1003	高等数学	6	81002

接着到成绩表中查找“课程号”为“81002”，学号为“944114044”的对应成绩值，如表 1-3 所示。

表 1-3 学生成绩表

课程号	学号	成绩
1001	982111056	91
1002	981000021	61
1003	944114044	52
1001	981000021	82

通过上面的例子可以看出，关系模型中数据的逻辑结构就是一张二维表，它由行和列组成。一张二维表对应了一个关系，表中的一行即为一条记录，表中的一列即为记录的一个属性。

关系模型的优点是：结构特别灵活，满足所有布尔逻辑运算和数学运算规则形成的查询要求；能搜索、组合和比较不同类型的数据；增加和删除数据非常方便。

其缺点是：数据库大时，查找满足特定关系的数据较费时；对空间关系无法满足。

1.3.4 数据模型的三要素

数据模型所描述的内容包括 3 部分：数据结构、数据操作和数据约束。

- 数据结构：数据模型中的数据结构主要描述数据的类型、内容、性质以及数据间的联系等。数据结构是数据模型的基础，数据操作和数据约束都建立在数据结构上。不同的数据结构具有不同的操作和约束。
- 数据操作：数据操作主要描述在相应的数据结构上的操作类型和操作方式。
- 数据约束：数据模型中的数据约束主要描述数据结构内数据间的语法、词义联系以及它们之间的制约和依存关系、数据动态变化的规则，以保证数据的正确、有效和相容。





1.4 关系数据库

关系数据库是当今世界的主流数据库。本节主要介绍关系模型中的一些基本术语，以及关系模型的完整性约束。

在关系模型中基本数据结构是二维表，不用像层次或网状那样的链接指针。记录之间的联系是通过不同关系中的同名属性来体现的。例如，要查找某个教师讲授的课程，首先要在“教师”关系中根据“姓名”查找到对应的教师“编号”，然后根据“编号”的值在“课程”关系中找到对应的“课程名”即可。在查询过程中，同名属性教师“编号”起到了连接两个关系的纽带作用。由此可见，关系模型中的各个关系模式不应当孤立起来，不是随意拼凑的一堆二维表，它必须满足相应的要求。

1.4.1 关系模型中的基本术语

关系模型中经常用到的术语如下。

1. 关系

一个关系就是一张二维表。

2. 元组

二维表中的每一条记录就是一个元组，它是构成关系的一个个实体，可以说，“关系”是“元组”的集合，“元组”是属性值的集合，一个关系模型中的数据就是这样逐行逐列组织起来的。

3. 属性

二维表中的一列就是一个属性，又称为字段，第一行列出的是属性名(字段名)。

4. 域

属性的取值范围。例如，“性别”属性只能取值为“男”或“女”。

5. 分量

元组中的一个属性值。关系模型要求关系必须是规范化的，最基本的条件就是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。

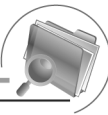
6. 关系模式

对关系的描述，一般表示如下。

关系名(属性 1, 属性 2, ……, 属性 n)

例如，可以将学生关系描述为：





学生(学号, 姓名, 性别, 出生年月, 籍贯, 院系编号)

7. 候选关键字

关系中的一个或几个属性的集合, 该属性集唯一标识一个元组, 这个属性集合称为候选关键字。

8. 关系数据库

对应于一个关系模型的所有关系的集合称为关系数据库。

9. 主关键字

一个关系中有多个候选关键字, 可以选择其中一个作为主关键字, 也称为主码或主键。

10. 外关键字

如果一个属性组不是所在关系的关键字, 但它是其他关系的关键字, 则该属性组称为外关键字, 也称为外码或外键。

11. 主属性

包含在任一候选关键字中的属性称为主属性, 不包含在任何候选关键字中的属性称为非关键字属性。

例如, 描述院系的关系模式如下:

院系(院系编号, 院系名称)

其主键为【院系编号】, 所以【学生】关系中的【院系编号】字段就是外键。关系是一个二维表, 但并不是所有的二维表都是关系。关系应该具有如下性质。

- 每一列中的分量是同一类型的数据。
- 不同的列要给予不同的属性名。
- 列的次序可以任意交换。
- 一个关系中的任意两个元组不能完全相同。
- 行的次序可以任意交换。
- 每一个分量必须是不可分的数据项。

1.4.2 关系数据库中表之间的关系

在关系数据库中, 可以通过外部关键字来实现表与表之间的联系, 公共字段是一个表的主键和另一个表的外键。如图 1-6 所示的【学生】表和【院系】表都包含【院系编号】属性, 通过这个字段就可以在【院系】和【学生】表之间建立联系, 这个联系是一对多的联系, 即一个院系中有多个学生。





学号	姓名	性别	出生年月	籍贯	院系编号
20060101	洪伟	女	1982-1-1	广西	01
20060102	许诺	女	1984-1-1	北京	02
20060103	丁丁	女	1983-1-1	江西	01
20060104	赵建	男	1978-1-1	河北	02

院系代码	院系名称
01	数学科学学院
02	计算机技术学院

图 1-6 【学生】表和【院系】表之间的联系

1.4.3 关系模型的完整性约束

关系模型的完整性规则是对关系的某种约束条件，也就是说，关系的值随着时间变化应该满足一些约束条件。这些约束条件实际上是现实世界的要求。任何关系在任何时刻都要满足这些语义约束。

关系模型中有 3 类完整性约束：实体完整性、参照完整性和用户定义的完整性。其中，实体完整性和参照完整性是关系模型必须满足的完整性约束条件，被称作是关系的两个不变性，应该由关系系统自动支持。用户定义的完整性是应用领域需要遵循的约束条件，体现了具体领域中的语义约束。

1. 实体完整性(Entity Integrity)

实体完整性规则为：如果属性(指一个或一组属性)A 是基本关系 R 的主属性，则 A 不能取空值。所谓空值就是“不知道”或“不存在”的值。例如，在“学生”关系中，【学号】这个属性为主键，则该字段不能取空值。

按照实体完整性规则的规定，基本关系的主键都不能取空值。如果主键由若干属性组成，则所有这些主属性都不能取空值。

对于实体完整性规则说明如下：

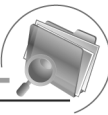
(1) 实体完整性规则是针对基本关系而言的。一个基本表通常对应现实世界的一个实体集。例如，“学生”关系对应于学生的集合。

(2) 现实世界中的实体是可区分的，即它们具有某种唯一性标识。例如，每个学生都是独立的个体，是不一样的。

(3) 关系模型中以主键作为唯一性标识。

(4) 主键中的属性即主属性不能取空值。如果主属性取空值，就说明存在某个不可标识的实体，即存在不可区分的实体，这与(2)相矛盾，因此这个规则称为实体完整性。





2. 参照完整性(Referential Integrity)

参照完整性规则为：如果属性(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(基本关系 R 和 S 不一定是不同的关系)，则对于 R 中每个元组在 F 上的值必须为空或是等于 S 中某个元组的主键值。

现实世界中的实体之间往往存在某种联系，在关系模型中，实体和实体之间的联系都是用关系来描述的，这样就自然存在着关系和关系间的引用。例如，图 1-6 中的【学生】表和【院系】表，【学生】表中每条学生记录的【院系编号】在【院系】表中必须存在，即学生所属的院系必须是该学校中已存在的院系。



提示

除了不同关系之间存在参照完整性之外，同一个关系的内部也可能存在参照完整性。

3. 用户定义的完整性(User-defined Integrity)

任何关系数据库系统都应该支持实体完整性和参照完整性。这是关系模型所要求的。除此之外，不同的关系数据库系统根据其应用环境的不同，往往还需要一些特殊的约束条件。用户定义的完整性就是针对某一具体关系数据库的约束条件。它反映某一具体应用所涉及的数据必须满足的语义要求。例如，某个属性必须取唯一值、某个非主属性也不能取空值、某个属性的取值范围在 0~100 之间(如学生的成绩)等。

关系模型应提供定义和检验这类完整性的机制，以使用统一的、系统的方法处理它们，而不要由应用程序承担这一功能。

满足完整性约束的关系数据库中的数据表之间不但具有独立性，而且若干个表之间又有一定的相关性，这一特点使其具有极大的优越性，主要表现在以下几点。

- 使数据具有最小的冗余度，支持复杂的数据结构。
- 数据具有共享性，能为多个用户服务。
- 具有高度的数据和程序的独立性，应用程序与数据的逻辑结构及数据的物理存储方式无关。

1.5 关系代数

关系代数是一种抽象的查询语言，它用关系的运算来表达查询。

任何一种运算都是将一定的运算符作用于一定的运算对象之上，从而得到预期的结果，所以运算对象、运算符和运算结果是运算的三大要素。

关系代数的运算对象是关系，运算结果也是关系。关系代数用到的运算符包括 4 类：集合运算符、专门的关系运算符、比较运算符和逻辑运算符，如表 1-4 所示。





表 1-4 关系代数用到的运算符

运算符		含义	运算符		含义
集合运算符	U	并	比较运算符	>	大于
	-	差		<	小于
	∩	交		≠	不等于
	×	笛卡尔积		≥	大于等于
		≤		小于等于	
专门的关系运算符	σ	选择	逻辑运算符	¬	非
	Π	投影		∧	与
	÷	除		∨	或
	⋈	连接			

按照运算符的不同，可以将关系代数的运算分为传统的集合运算和专门的关系运算两大类。其中，传统的集合运算将关系看成是元组的集合，其运算是从关系的“水平”方向即行的角度来进行的；而专门的关系运算同时涉及行和列。比较运算符和逻辑运算符则是用来辅助专门的关系运算符进行操作的。

关于关系代数的理论，在这仅作简单介绍，详细信息请参考专门的数据库理论书籍。

1.5.1 传统的集合运算

传统的集合运算都是二目运算，包括并、差、交和笛卡尔积 4 种运算。

设关系 R 和关系 S 都具有 n 个属性，且相应的属性取自同一个域，t 是元组变量， $t \in R$ 表示 t 是 R 的一个元组，如图 1-7 所示。

R			S		
A	B	C	A	B	C
a1	b1	c1	a1	b2	c2
a1	b2	c2	a1	b3	c2
a2	b2	c1	a2	b2	c1

图 1-7 关系 R 和关系 S

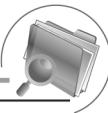
可以定义并、差、交、笛卡尔积运算如下。

1. 并

关系 R 和关系 S 的并运算记作。

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





其结果仍具有 n 个属性，由属于 R 或属于 S 的元组组成，结果如图 1-8 所示。

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1
a1	b3	c2

图 1-8 $R \cup S$

2. 差

关系 R 和关系 S 的差记作：

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

其结果关系仍具有 n 个属性，由属于 R 而不属于 S 的所有元组组成，结果如图 1-9 所示。

A	B	C
a1	b1	c1

图 1-9 $R - S$

3. 交

关系 R 和关系 S 的交记作：

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

其结果关系仍具有 n 个属性，由既属于 R 又属于 S 的元组组成。关系的交也可用差来表示，即 $R \cap S = R - (R - S)$ ，结果如图 1-10 所示。

A	B	C
a1	b2	c2
a2	b2	c1

图 1-10 $R \cap S$

4. 笛卡尔积

严格地讲，在这里的笛卡尔积应该是广义的笛卡尔积，因为这里的笛卡尔积的元素是元组。

两个分别具有 n 和 m 个属性的关系 R 和 S 的笛卡尔积是一个 $(n+m)$ 列的元组的集合。元组的前 n 列是关系 R 的一个元组，后 m 列是关系 S 的一个元组。若 R 有 k_1 个元组， S 有 k_2 个元组，则关系 R 和关系 S 的笛卡尔积有 $k_1 \times k_2$ 元组。记作：

$$R \times S = \{t_r t_s \mid t_r \in R \wedge t_s \in S\}$$

结果如图 1-11 所示。





R.A	R.B	R.C	S.A	S.B	S.C
a1	b1	c1	a1	b2	c2
a1	b1	c1	a1	b3	c2
a1	b1	c1	a2	b2	c1
a1	b2	c2	a1	b2	c2
a1	b2	c2	a1	b3	c2
a1	b2	c2	a2	b2	c1
a2	b2	c1	a1	b2	c2
a2	b2	c1	a1	b3	c2
a2	b2	c1	a2	b2	c1

图 1-11 R×S

1 5.2 专门的关系运算

专门的关系运算包括选择、投影、连接、除运算等。下面简单介绍这些运算。

1. 选择

从一个关系中选出满足给定条件的记录的操作称为选择或筛选。选择运算是从行的角度进行的运算，选出满足条件的那些记录构成原关系的一个子集，其中，条件表达式中可以使用=、<>、>=、>、<和<=等比较运算符，多个条件之间可以使用 AND(∧)、OR(∨)和 NOT(¬) 进行连接。选择操作记作：

$$\sigma_F(R) = \{t \mid t \in R \wedge F(t) = \text{'真'}\}$$

其中，F 表示选择条件。

假设对于表 1-1 所示的学生信息表 Student，如果要查询年龄小于 25 岁的学生可以表示为：

$$\sigma_{\text{年龄} < 25}(\text{Student}) \text{ 或 } \sigma_{4 < 25}(\text{Student})$$

运算的结果为学生信息表 Student 中所有年龄小于 25 的记录。这里的 4 表示 Student 表的第 4 列。

2. 投影

从一个关系中选出若干指定字段的值的操作称为投影。投影是从列的角度进行的运算，所得到的字段个数通常比原关系少，或者字段的排列顺序不同。

投影操作记作：

$$\pi_A(R) = \{t[A] \mid t \in R\}$$

其中，A 为 R 中的属性列。

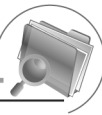
例如，查询学生的姓名和联系电话的操作如下：

$$\pi_{\text{姓名, 联系电话}}(\text{Student}) \text{ 或 } \pi_{2,6}(\text{Student})$$

运算的结果为姓名和联系电话两列，以及这两列对应的所有数据组成的关系。

投影之后得到的关系不仅取消了原关系中的某些列，而且还可能取消原关系中的某些元组，因为取消了某些列之后，就可能出现重复行，应取消这些完全相同的行。





3. 连接

连接是把两个关系中的记录按一定条件横向结合,生成一个新的关系。最常用的连接运算是自然连接,它是利用两个关系中公用字段,把该字段值相等的记录连接起来。

需要明确的是,选择和投影都属于单目运算,它们的操作对象只是一个关系,而连接则是双目运算,其操作对象是两个关系。

连接操作记作:

$$R \underset{A \cap B}{\bowtie} S = \{t_r t_s \mid t_r \in R \wedge t_s \in S \wedge t_r[A] = t_s[B]\}$$

系统在执行连接运算时,要进行大量的比较操作。不同关系中的公共字段或具有相同语义的字段是实现连接运算的“纽带”,例如“学生信息表”和“学生成绩表”可以通过“Student.学号”和“Score.学号”作为连接的“纽带”。

4. 除

给定关系 $R(X,Y)$ 和 $S(Y,Z)$, 其中 X, Y, Z 为属性组。 R 中的 Y 和 S 中的 Y 可以有不同的属性名,但必须出自相同的域集。

那么 R 和 S 的除运算得到一个新的关系 $P(X)$, P 是 R 中满足下列条件的元组在 X 属性列上的投影: 元组在 X 上分量值 x 的象集 Y_x 包含 S 在 Y 上投影的集合。

除运算记作:

$$R \div S = \{t_r[X] \mid t_r \in R \wedge \pi_y(S) \subseteq Y_x\}$$

其中 Y_x 为 x 在 R 中的象集, $x = \pi_x[t_r]$ 。



知识点

除操作是同时从行和列角度进行运算的。

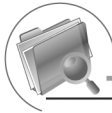


1.6 规范化理论

为了使数据库设计的方法趋于完善,人们研究了规范化理论。目前规范化理论的研究已经有了很大的发展。本节将主要介绍模式规范化在数据库设计过程中的必要性及其规范化原理。

1.6.1 模式规范化的必要性

一般而言,关系数据库设计的目标是生成一组关系模式,使用户既不必存储不必要的重复信息,又可以方便地获取信息。方法之一就是设计满足适当范式的模式。要确定一个关系模式是否属于期望的范式,还需要有关作为数据库建模对象的现实企业的额外信息。



1.6.2 规范化理论的基本概念

满足一定条件的关系模式称为范式(Normal Form, 简称NF)。在1971年至1972年,关系数据库模型的创始人E.F.Codd系统地提出了第一范式(1NF)、第二范式(2NF)和第三范式(3NF)的概念。1974年Codd和Boyce共同提出了BCNF范式,为第三范式的改进。一个低级范式的关系模式,通过投影分解的方法可转换成多个高级范式的关系模式的集合,这个过程称为规范化。

1. 第一范式(1NF)

第一范式是最低的规范化要求,它要求关系满足一种最基本的条件,它与其他范式不同,不需要诸如函数依赖之类的额外信息。

如果某个域的元素被认为是不可分的单元,那么这个域就是原子的(atomic);如果一个关系模式的所有熟悉的域都是原子的,则称此关系模式属于第一范式(1NF)。

第一范式要求数据表不能存在重复的记录,即存在一个关键字,第二个要求是每个字段都已经分到最小不再可分,关系数据库的定义就决定了数据库满足这一条。主关键字应满足下面几个条件。

- 主关键字在表中是唯一的。
- 主关键字段不存在空值。
- 每条记录都必须有一个主关键字。
- 主关键字是关键字的最小子集。

满足第一范式的关系模式有许多不必要的重复值,并且增加了修改数据时疏漏的可能性,为了避免这种数据冗余和更新数据的疏漏,就引出了第二范式。

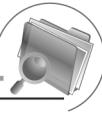
2. 第二范式(2NF)

如果一个关系属于第一范式(1NF),且所有的非主关键字段都完全依赖于主关键字,则称之为第二范式。

举个例子来说,有一个存储物品的关系有5个字段(物品ID、仓库号、物品名称、物品数量、仓库地址),这个库符合1NF,其中“物品ID”和“仓库号”构成主关键字,但因为“仓库地址”只完全依赖于“仓库号”,即只依赖于主关键字的一部分,所以它不符合第二范式(2NF)。这样首先存在数据冗余,因为仓库数量可能不多,其次,在更改仓库地址时,如果漏改了某一条记录,存在数据不一致性。再次,如果某个仓库的物品全部出库了,那么这个仓库地址就会丢失,所以这种关系不允许存在某个仓库中不放物品的情况。可以用投影分解的方法消除部分依赖的情况,从而达到2NF的标准。方法是从关系中分解出新的二维表,使得每个二维表中所有的非关键字都完全依赖于各自的主关键字。这里可以做如下分解,将原来的一个表分解成两个表:

物品(物品ID, 仓库号, 物品名称, 物品数量)
仓库(仓库号, 仓库地址)





这样就完全符合第二范式(2NF)了。

3. 第三范式(3NF)

如果一个关系属于第二范式(2NF),且每个非关键字不传递依赖于主关键字,这种关系就是第三范式(3NF)。简而言之,从2NF中消除传递依赖,就是3NF。如有一个关系(姓名,工资等级,工资额),其中姓名是关键字,此关系符合2NF,但是因为工资等级决定工资额,这就叫传递依赖,它不符合3NF。同样可以使用投影分解的方法将上表分解成两个表:(姓名,工资等级)和(工资等级,工资额)。

上面提到了投影分解的方法,关系模式的规范化过程是通过投影分解来实现的。这种把低一级关系模式分解成若干个高一级关系模式的投影分解方法不是唯一的,应该在分解中满足3个条件。

- 无损连接分解,分解后不丢失信息。
- 分解后得到的每个关系都是高一级范式,不要同级甚至低级分解。
- 分解的个数最少,这就是完美要求,应该做到尽量少。

一般情况下,规范化到3NF就满足需要了,规范化程度更高的还有BCNF、4NF、5NF。

规范化的基本思想是逐步消除数据依赖中不合适的部分,使模式中的各种关系模式达到某种程度的“分离”,即“一事一地”的模式设计原则。让一个关系描述一个概念、一个实体或者实体间的一种联系。如果多于一个概念,就把它分离出去。因此所谓规范化实质上是概念的单一化。

应该指出的是,规范化的优点是明显的,它避免了大量的数据冗余,节省了空间,保持了数据的一致性,如果完全达到3NF,用户不会在超过两个以上的地方更改同一个值,而当记录会经常发生改变时,这个优点便很容易显现出来。但是,它最大的不利是,由于用户把信息放在不同的表中,增加了操作的难度,同时把多个表连接在一起的时间花费也是巨大的,节省了时间必然付出了空间的代价,反之,节省了空间也必然要付出时间的代价,时间和空间在计算机领域中是一个矛盾统一体,它们是互相作用、对立统一的。因为表和表的连接操作的时间花费是很大的,从而降低了系统运行性能。

4. 第四范式(4NF)

第四范式的定义用到了多值依赖,多值依赖的定义如下:设 $R(U)$ 是属性集 U 上的一个关系模式。 X 、 Y 、 Z 是 U 的子集,并且 $Z=U-X-Y$ 。关系模式 $R(U)$ 中多值依赖 $X \twoheadrightarrow Y$ 成立,当且仅当对 $R(U)$ 的任一关系 r ,给定的一对 (x, z) 值有一组 Y 的值,这组值仅仅决定于 x 值而与 z 值无关。

若 $X \twoheadrightarrow Y$,而 $Z=\Phi$ 即 Z 为空,则称 $X \twoheadrightarrow Y$ 为平凡的多值依赖。

多值依赖具有以下性质。

- 多值依赖具有对称性,即若 $X \twoheadrightarrow Y$,则 $X \twoheadrightarrow Z$,其中 $Z=U-X-Y$ 。
- 多值依赖的传递性,即若 $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$,则 $X \twoheadrightarrow Z-Y$ 。





函数依赖和多值依赖集为 D 的关系模式 $R(U)$ 属于第四范式(4NF)的条件是对所有 $D+$ 中形如 $X \twoheadrightarrow Y$ 的多值依赖(X, Y 是 U 的子集), 至少有以下条件之一成立。

- $X \twoheadrightarrow Y$ 是一个平凡的多值依赖。
- X 是 R 的超码。

5. 其他范式

第四范式不是“最终”范式, 正如前面提到的, 多值依赖有助于理解并解决利用函数依赖无法理解的某些形式的信息重复。还有一些类型的概括多值依赖的约束称为连接依赖(Join Dependence), 由此引出的另外一种范式称为投影-连接范式(Project-Join Normal Form, 简称 PJNF), 有的书中也将其称为第五范式(5NF)。

使用这些通用约束的一个实际问题是, 难以用于推导, 而且还没有形成一套具有保真性和完备性的推理规则用于约束的推导, 因此很少被使用。



提示

有关范式的更多信息请读者参考相关书籍, 本书不作过多介绍。

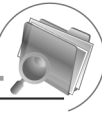
1.7 数据库语言

数据库系统提供两种不同类型的语言: 一种是数据定义语言, 用于定义数据库模式; 另一种是数据操纵语言, 用于表达数据库的查询和更新。而实际上, 数据定义和数据操纵语言并不是两种分离的语言, 相反, 它们构成了单一的数据库语言, 比如广泛使用的 SQL 语言。

1.7.1 数据定义语言 DDL

数据库模式是通过一系列定义来说明的, 这些定义由一种称为数据定义语言(Data-Definiton Language, DDL)的特殊语言来表达。例如, 下面的 SQL 语句描述了 USER 表的定义:

```
Create table USER  
(cno varchar(10),  
  Cname varchar(50),  
  csex varchar(4),  
  cage integer,  
  cdeptno integer ,  
  ctelephone varchar(20))
```



1.7.2 数据操纵语言 DML

数据操纵语言(Data-Manipulation Language, DML)使得用户可以访问或操纵那些按照某种特定数据模式组织起来的数据。数据操纵包括对存储在数据库中的信息进行检索,向数据库中插入新的信息,从数据库中删除信息和修改数据库中存储的信息。

通常有以下两种基本的数据操纵语言。

- 过程化 DML: 要求指定需要什么数据以及如何获得这些数据。
- 陈述式 DML: 也称非过程化 DML, 只要求用户指定需要什么数据, 而不指明如何获得这些数据。

通常陈述式 DML 比过程化 DML 更易学易用。但是, 由于不必指明如何获得数据, 因此数据库系统会指出一种访问数据的高效路径。SQL 语言的 DML 部分是非过程化的。

查询是要求对信息进行检索的语句。DML 中涉及信息检索的部分称为查询语句。例如, 下面的语句将从 USER 表中查询名为“潘涌纸”的用户信息:

```
SELECT * FROM USER WHERE cname='潘涌纸';
```

1.8 数据库设计

数据库设计是指对于一个给定的应用系统, 构造(设计)优化的数据库逻辑模式和物理结构, 并据此建立数据库及其应用系统, 使之能够有效地存储和管理数据, 满足各种用户的应用需求, 包括信息管理要求和数据操作要求。

- 信息管理要求是指在数据库中应该存储和管理哪些数据对象;
- 数据操作要求是指对数据对象需要进行哪些操作, 如查询、增加、删除、修改和统计等操作。

1.8.1 数据库设计的目标

数据库设计的目标是为用户和各种应用系统提供一个信息基础设施和高效率的运行环境。高效率的运行环境包括数据库数据的存取效率、数据库存储空间的利用率以及数据库系统运行管理的效率等。

数据库设计的过程是数据库应用系统从设计、实施到运行与维护的全过程。

1.8.2 数据库设计的特点

数据库设计和一般的软件系统的设计、开发和运行与维护有许多相同之处, 更有其自身的





一些特点。

1. 数据库建设的基本规律

“三分技术，七分管理，十二分基础数据”是数据库设计的特点之一。

在数据库建设中，不仅涉及技术，还涉及管理。要建设好一个数据库应用系统，开发技术固然重要，但是相比之下则管理更加重要。这里的管理不仅仅包括数据库建设作为一个大型的工程项目本身的项目管理，而且包括该企业的业务管理。

“十二分基础数据”则强调了数据的收集、整理、组织和不断更新是数据库建设中的重要环节。人们往往忽视基础数据在数据库建设中的地位和作用。基础数据的收集、入库是数据库建立初期工作量最大、最烦琐、最细致的工作。在以后数据库运行过程中更需要不断地把新的数据加入到数据库中，使数据库成为一个“活库”，否则就成为“死库”。数据库一旦成了“死库”，系统也就失去了应用价值，原来的投资也就失败了。

2. 结构(数据)设计和行为(处理)设计相结合

数据库设计应该和应用系统相结合。也就是说，整个设计过程中要把数据库结构设计和对数据的处理设计密切结合起来。这是数据库设计的特点之二。

在早期的数据库应用系统开发过程中，常常把数据库设计和应用系统的设计相分离开来。由于数据库设计有它专门的技术和理论，因此，需要专门来讲解数据库设计。这并不等于数据库设计和在数据库之上开发应用系统是相互分离的。相反，必须强调设计过程中数据库设计和应用程序的密切结合，并把它作为数据库设计的重要特点。

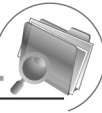
1 8.3 数据库设计的方法

大型数据库设计是涉及多学科的综合性的技术，同时又是一项庞大的工程项目。它要求从事数据库设计的专业人员具备多方面的技术和知识。主要包括：计算机的基础知识、软件工程的原理和方法、程序设计的方法和技巧、数据库的基本知识、数据库设计技术、应用领域的知识等。这样才能设计出符合具体领域要求的数据库及其应用系统。

早期数据库设计主要采用手工与经验相结合的方法。设计的质量往往与设计人员的经验与水平有直接的关系。数据库设计是一种技艺，缺乏科学理论和工程方法的支持，设计质量难以保证。因此，人们努力探索，提出了各种数据库设计方法，其中比较著名的有以下4种。

- 新奥尔良(New Orleans)方法：该方法把数据库设计分为若干阶段和步骤，并采用一些辅助手段实现每一过程。它运用软件工程的思想，按一定的设计规程用工程化方法设计数据库。新奥尔良方法属于规范化设计法。虽然从本质上看它仍然是手工设计方法，其基本思想是过程迭代和逐步求精。
- 基于 E-R 模型的数据库设计方法：该方法用 E-R 模型来设计数据库的概念模型，是数据库概念设计阶段广泛采用的方法。





- 3NF(第三范式)设计方法: 该方法用关系数据理论为指导来设计数据库的逻辑模型, 是设计关系数据库时在逻辑阶段可以采用的一种有效方法。
- ODL(Object Definition Language)方法: 这是面向对象的数据库设计方法。该方法用面向对象的概念和术语来说明数据库结构。ODL 可以描述面向对象的数据库结构设计, 可以直接转换为面向对象的数据库。

1 8.4 数据库设计的步骤

数据库设计是指对于一个给定的应用环境, 构造最优的数据库模式, 建立数据库及其应用系统, 使之能够有效地存储数据, 满足各种用户的应用需求。

数据库设计一般分为以下 6 个步骤。

1. 需求分析

进行数据库设计首先必须准确了解与分析用户需求, 包括数据和处理。需求分析是整个设计过程的基础, 是最困难、最耗时的一步。作为“地基”的需求分析是否做得充分与准确, 决定了在其上构建数据库大厦的速度与质量。需求分析做得不好, 可能会导致整个数据库设计返工重做。

2. 概念结构设计

概念结构设计是整个数据库设计的关键, 它通过对用户需求进行综合、归纳与抽象, 形成一个独立于具体 DBMS 的概念模型。

概念模型是整个组织各个用户关心的信息结构。描述概念结构的有力工具是 E-R 图。数据库设计通常基于 E-R 模型来进行, 然后转化成关系模型。

3. 逻辑结构设计

逻辑结构设计将概念结构转换为某个 DBMS 所支持的数据模型, 并对其进行优化。

4. 物理结构设计

物理设计为逻辑数据模型选取一个最适合应用环境的物理结构, 包括存储结构和存取方法等。物理结构设计通常分为以下两步。

- 确定数据库的物理结构: 可分为确定数据的存取方法和数据的存储结构。
- 对物理结构进行评估: 包括对时间效率、空间效率、维护开销和各种用户要求进行权衡, 从多种设计方案中选择一个较优的方案。

5. 数据库实施

在数据库实施阶段, 设计人员运用 DBMS 提供的数据库语言(如 SQL)及其宿主语言, 根据逻辑设计和物理设计的结果建立数据库, 编制与调试应用程序, 组织数据入库, 并进行调





试运行。

(1) 定义数据库结构

确定了数据库的逻辑结构与物理结构后,就可以用所选用的 DBMS 提供的数据库定义语言 (DDL)来严格描述数据库结构。

(2) 数据装载

数据库结构建立后,就可以向数据库中装载数据了。组织数据入库是数据库实施阶段最主要的工作。对于数据量不是很大的小型系统,可以用人工方式完成数据的入库,具体包括如下几个步骤。

- 筛选数据: 需要装入数据库中的数据通常都分散在各个部门的数据文件或原始凭证中,所以首先必须把需要入库的数据筛选出来。
- 转换数据格式: 筛选出来的需要入库的数据,其格式往往不符合数据库要求,还需要进行转换。这种转换有时可能很复杂。
- 输入数据: 将转换好的数据输入计算机中。
- 校验数据: 检查输入的数据是否有误。

对于中大型系统,由于数据量大,用人工方式组织数据入库将会耗费大量人力物力,而且很难保证数据的正确性,因此应该设计一个数据输入子系统由计算机辅助数据的入库工作。

(3) 编制与调试应用程序

数据库应用程序的设计应该与数据设计并行进行。在数据库实施阶段,当数据库结构建立好后,就可以开始编制与调试数据库的应用程序,也就是说,编制与调试应用程序是与组织数据入库同步进行的。调试应用程序时由于数据入库尚未完成,可先使用模拟数据。

(4) 数据库试运行

应用程序调试完成,并且已有少部分数据入库后,就可以开展数据库的试运行。数据库试运行也称为联合调试,其主要工作如下。

- 功能测试: 即实际运行应用程序,执行对数据库的各种操作,测试应用程序的各种功能。
- 性能测试: 即测量系统的性能指标,分析是否符合设计目标。

6. 数据库运行和维护

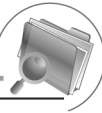
数据库应用系统经过试运行后即可投入正式运行。数据库投入运行标志着开发任务的基本完成和维护工作的开始,并不意味着设计过程的终结,由于应用环境在不断变化,数据库运行过程中物理存储也会不断变化,对数据库设计进行评价、调整、修改等维护工作是一个长期的任务,也是设计工作的继续和提高。

在数据库运行阶段,对数据库经常性的维护工作主要是由 DBA 完成的,主要包括以下内容。

(1) 数据库的转储和恢复

定期对数据库和日志文件进行备份,以保证一旦发生故障,能利用数据库备份及日志文件备份,尽快将数据库恢复到某种一致性状态,并尽可能减少对数据库的破坏。





(2) 数据库的安全性、完整性控制

DBA 必须对数据库安全性和完整性控制负起责任。根据用户的实际需要授予不同的操作权限。另外,由于应用环境的变化,数据库的完整性约束条件也会变化,也需要 DBA 不断修正,以满足用户要求。

(3) 数据库性能的监督、分析和改进

目前许多 DBMS 产品都提供了监测系统性能参数的工具, DBA 可以利用这些工具方便地得到系统运行过程中一系列性能参数的值。DBA 应该仔细分析这些数据,通过调整某些参数来进一步改进数据库性能。

(4) 数据库的重组和重构造

数据库运行一段时间后,由于记录的不断被增、删、改,会使数据库的物理存储变坏,从而降低数据库存储空间的利用率和数据的存取效率,使数据库的性能下降。这时, DBA 就要对数据库进行重组,或部分重组(只对频繁增、删的表进行重组)。数据库的重组不会改变原设计的数据逻辑结构和物理结构,只是按原设计要求重新安排存储位置,回收垃圾,减少指针链,提高系统性能。DBMS 一般都提供了供重组数据库使用的实用程序,帮助 DBA 重新组织数据库。

数据库应用环境发生变化,会导致实体及实体间的联系也发生相应的变化,使原有的数据库设计不能很好地满足新的需求,从而不得不适当调整数据库的模式和内模式,这就是数据库的重构造。DBMS 都提供了修改数据库结构的功能。

重构造数据库的程度是有限的。如果应用变化太大,已无法通过重构数据库来满足新的需求,或重构数据库的代价太大时,则表明现有数据库应用系统的生命周期已经结束,应该重新设计新的数据库系统,开始新数据库应用系统的生命周期。



提示

设计一个完善的数据库应用系统是不可能一蹴而就的,往往是上述 6 个阶段的不断反复。

1.9 上机练习

本章的上机练习将设计一个网上购物系统的概念模型,并用 E-R 图表示各实体之间的关系。

(1) 在网上购物系统中包括如下实体:商品、购物车、用户等。

(2) 实体之间的关系如下:购物车中可以有多件商品,每类商品也可以被添加到不同的购物车中,购物车属于某个用户,每个用户可以购物多次,即产生多个购物车。

(3) 确定每个实体的属性:【商品】实体的属性有【商品编号】、【商品名称】、【类别】、【单价】、【库存数量】;【购物车】实体的属性有【购物车编号】、【用户编号】和【购物车状态】;【用户】实体的属性有【用户编号】、【姓名】、【地址】和【联系电话】。

(4) 最终得到的 E-R 图如图 1-12 所示。



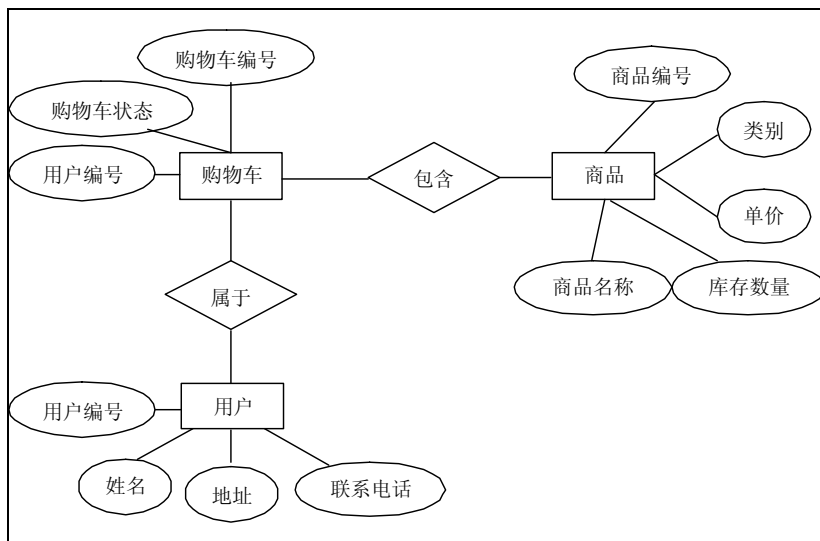


图 1-12 网上购物系统的 E-R 图



1.10 习题

1. 什么是数据库？什么是数据库系统？
2. 什么是数据库管理系统？它有哪些主要功能？
3. 常用的数据模型有哪几种？
4. 什么是关系模型？它是如何表示实体和实体之间的联系的？
5. 常用的关系运算有哪些？如何区分一元运算和二元运算？
6. 为什么要进行关系模式规范化？
7. 第三范式与第二范式相比有哪些改进？
8. 什么是数据操纵语言？它有什么作用？
9. 简述数据库设计的步骤。

第2章

Access 2010 基础

学习目标

Access 2010 是一个面向对象的、采用事件驱动的新型关系数据库，它提供了强大的数据处理功能，可以帮助用户组织和共享数据库信息，以便根据数据库信息做出有效的决策。它具有界面友好、易学易用、开发简单、接口灵活等特点，因此，目前许多中小型网站都使用 Access 作为后台数据库系统。本章将主要介绍 Access 2010 的基本工作环境及其所具有的对象。

本章重点

- Access 2010 的启动与关闭
- Access 2010 的功能区
- 自定义功能区
- Access 2010 数据库对象
- 自定义快速访问工具栏

2.1 初识 Access 2010

Access 是美国 Microsoft 公司推出的关系型数据库管理系统(RDBMS)，它是 Microsoft Office 的组成部分之一。具有与 Word、Excel 和 PowerPoint 等相同的操作界面和使用环境，深受广大用户的喜爱。Access 2010 是其最新版本。Access 2010 还可以通过 ODBC 与 Oracle、Sybase、FoxPro 等其他数据库相连，实现数据的交换和共享。

2.1.1 Access 简介

Access 2010 是一个面向对象的、采用事件驱动的新型关系数据库。它提供了表生成器、查



询生成器、宏生成器和报表设计器等许多可视化的操作工具，以及数据库向导、表向导、查询向导、窗体向导、报表向导等多种向导，可以使用户能够很方便地构建一个功能完善的数据库系统。

1. 概述

Access 能操作其他来源的资料，包括许多流行的 PC 数据库程序(如 DBASE、Paradox、FoxPro) 和服务、小型机及大型机上的许多 SQL 数据库。此外，Access 还提供了 Windows 操作系统的高级应用程序开发系统。与其他数据库开发系统相比，Access 有一个明显的区别，就是用户不需要编写一行代码，就可以在很短的时间里开发出一个功能强大且相当专业的数据库应用程序，并且这一过程是完全可视的，如果能给它加上一些简短的 VBA 代码，那么开发出的程序就与专业程序员潜心开发的程序一样了。

Access 的最主要优点是它不用携带向上兼容的软件。无论是对于有经验的数据库设计人员还是那些刚刚接触数据库管理系统的新手，都会发现 Access 所提供的各种工具既实用又方便，同时还能够获得高效的数据处理能力。

2. Access 发展历程

Access 数据库系统是一个关系数据库系统，它经历了一个长期的发展过程。

Microsoft 公司在 1990 年 5 月推出 Windows 3.0，该程序立刻受到了用户的欢迎和喜爱。1992 年 11 月 Microsoft 公司发行了 Windows 数据库关系系统 Access 1.0 版本。从此，Access 不断改进和再设计，自 1995 年起，Access 成为了办公软件 Office 95 的一部分。多年来，Microsoft 先后推出的 Access 版本有 2.0、7.0/95、8.0/97、9.0/2000、10.0/2002，直到今天的 Access 2003、2007、2010 版。本教程将介绍其最新版本 Access 2010。

2.1.2 启动 Access 2010

在安装好 Microsoft Office 2010 软件包之后，就可以从 Windows 界面启动 Access 2010 了。选择【开始】|【程序】| Microsoft Office | Microsoft Office Access 2010 命令即可启动 Access 2010。启动后的界面如图 2-1 所示。



提示

为了启动方便，可以在桌面上创建 Access 2010 的快捷启动方式，其操作方法是：将鼠标指针移到【开始】|【程序】| Microsoft Office | Microsoft Office Access 2010 命令上，按住鼠标左键不放，将其拖放到桌面上，以后只要双击桌面上此 Access 图标即可打开 Access。



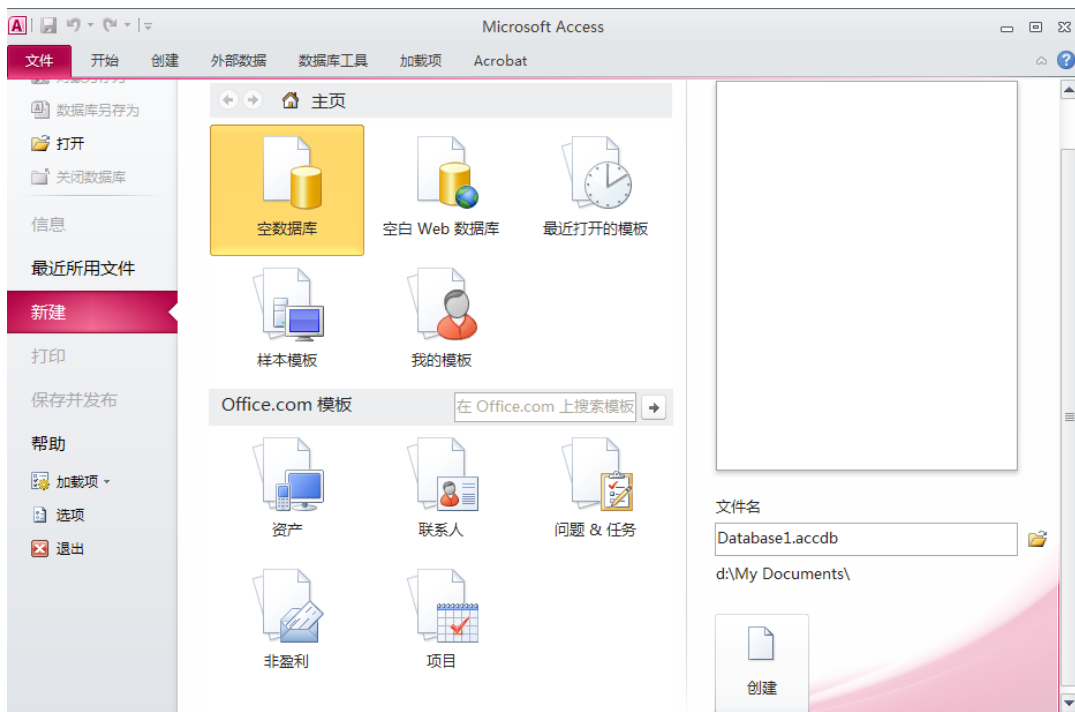
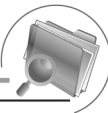





图 2-1 Access 2010 初始界面

2.1.3 关闭 Access 2010

与以前版本的 Office 软件的退出操作完全不同。执行下列任意一种操作都可以退出：

- (1) 在菜单栏中选择【文件】|【退出】命令。
- (2) 单击标题栏右端的 Access 窗口的【关闭】按钮 。
- (3) 单击标题栏左端的 Access 窗口的【控制菜单】图标 ，在打开的下拉菜单中单击【关闭】命令。
- (4) 双击标题栏左端的 Access 窗口的【控制菜单】图标 。
- (5) 右击标题栏，在打开的快捷菜单中，选择【关闭】命令。
- (6) 按组合键 Alt + F4 键。

无论何时退出 Access，Access 2010 都将自动保存对数据所作的更改。但是，如果上一次保存之后又更改了数据库对象的设计，Microsoft Access 将在关闭之前询问是否保存这些更改。



提示

如果由于断电等原因意外地退出 Access 2010 系统，可能会损坏数据库。





2.2 Access 2010 的工作界面

Access 2010 采用了一种全新的用户界面，相对于旧版本的 Access 2003，其界面发生了相当大的变化，但是与 Access 2007 却非常类似。这种用户界面是微软公司重新设计的，可以帮助用户提高工作效率。

Access 2010 启动后，屏幕上就会出现 Access 2010 的首界面，如图 2-1 所示。除了【文件】相关的操作以菜单形式展示以外，其他功能都集中在几个功能区中，包括【开始】、【创建】、【外部数据】和【数据库工具】等。

2.2.1 起始页

从图 2-1 可以看到，在首界面中默认显示的是【文件】菜单下的【新建】页面，该页面的中央区域显示的是【可用模板】，这就是用户打开 Access 2010 以后所看到的第一项变化。

在【可用模板】中选择【样本模板】选项，可以查看当前 Access 2010 系统中所有的样本模板，如图 2-2 所示。

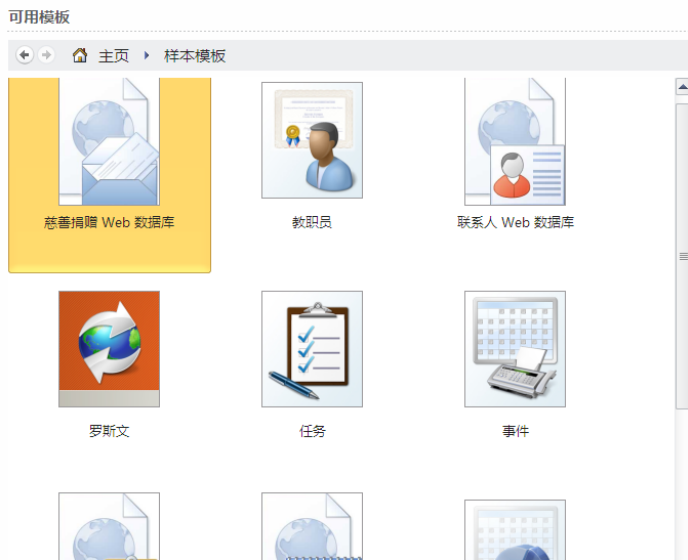


图 2-2 Access 2010 中的样本模板

Access 2010 提供的每个模板都是一个完整的应用程序，具有预先建立好的表、窗体、报表、查询、宏和表关系等。如果模板设计满足用户的需求，则通过模板建立数据库以后，就可以立即利用数据库工具开始工作；如果模板设计不能够完全满足用户的需求，则可以使用模板作为基础，对所创建的数据库进行修改，从而得到符合用户特定需求的数据库。

用户也可以通过【可用模板】中的【空数据库】或【空白 Web 数据库】选项创建一个空数据库。此时进入如图 2-3 所示的工作界面。

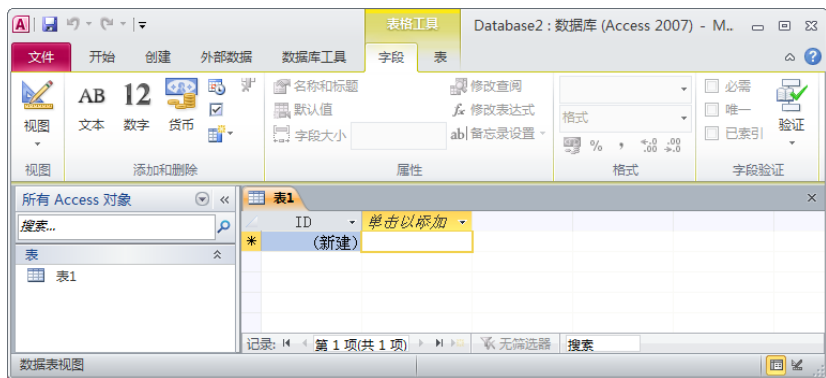
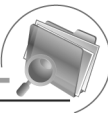



图 2-3 工作界面

由图 2-3 可以看出，Access 2010 的工作界面与 Windows 标准的应用程序相似，包括标题栏、功能区选项卡、状态栏、导航栏、数据库对象窗口以及帮助等部分。

2.2.2 标题栏

【标题栏】位于 Access 2010 工作界面的最上端，用于显示当前打开的数据库文件名。在标题栏的右侧有 3 个小图标，从左到右依次用于最小化、最大化(还原)和关闭应用程序窗口，这是标准的 Windows 应用程序的组成部分。

标题栏最左端的 Access 图标  是控制符，单击控制符会出现如图 2-4 所示的控制菜单。通过该菜单可以控制 Access 2010 窗口的还原、移动、大小、最小化、最大化和关闭等。双击控制符，可以直接关闭 Access 2010 窗口。

控制符的右边是【自定义快速访问工具栏】，如图 2-5 所示，单击工具栏右侧的下拉按钮，将打开【自定义快速访问工具栏】下拉菜单，可以定义工具栏中显示的快捷操作图标。



图 2-4 Access 控制菜单

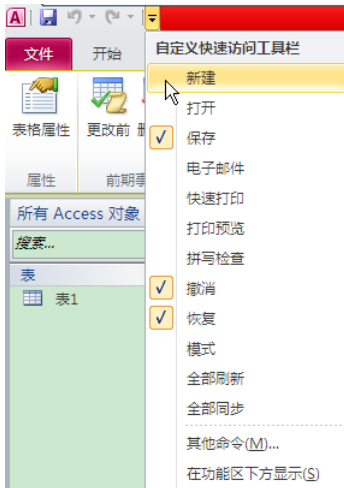


图 2-5 自定义快速访问工具栏





2.2.3 功能区

功能区是一个带状区域，位于程序窗口的顶部，标题栏的下方，它以选项卡的形式将各种相关的功能组合在一起，提供了 Access 2010 中主要的命令界面，如图 2-6 所示为功能区的【创建】选项卡。

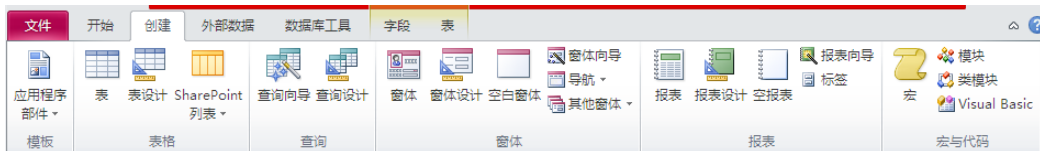


图 2-6 功能区

通过 Access 2010 的功能区，可以更快速地查找相关命令组。例如，如果要创建一个新的表格，可以在【创建】选项卡下找到各种创建表格的方式。

使用这种选项卡式的功能区，可以使各种命令的位置与用户界面更为接近，从而大大方便了用户的使用。由于在使用数据库的过程中，功能区是用户使用最频繁的区域，因此将在下一节详细介绍功能区。

2.2.4 导航窗格

导航窗格位于程序窗口的左侧，用于显示当前数据库中的各种数据库对象，它取代了 Access 早期版本中的数据库窗口。导航窗口有两种状态，折叠状态和展开状态，如图 2-7 和图 2-8 所示。单击导航窗格上部的「按钮或」按钮，可以展开或折叠导航窗格。如果需要较大的空间显示数据库，则可以把导航窗格折叠起来。



图 2-7 折叠状态的导航窗格

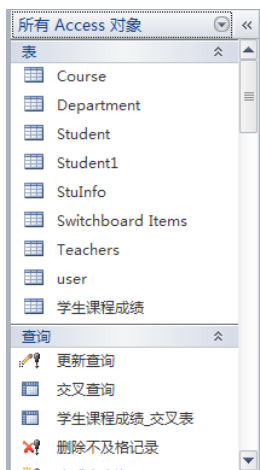
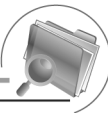


图 2-8 展开状态的导航窗格

导航窗格实现对当前数据库的所有对象的管理和对相关对象的组织。导航窗格显示数据库





中的所有对象，并按类别将它们分组。单击导航窗格右上方的箭头，可以显示如图 2-9 所示的分组列表。

在导航窗格中，可以对对象进行分组。分组是一种分类管理数据库对象的有效方法。在一个数据库中，如果某个表绑定到一个窗体、查询和报表，则导航窗格将把这些对象归组在一起。例如，当选择【表和相關视图】命令进行查看时，各种数据库对象就会根据各自的数据源表进行分类，如图 2-10 所示。



知识点

在导航窗格中，右击任何对象都会弹出快捷菜单，从中选择命令以执行所需的操作。

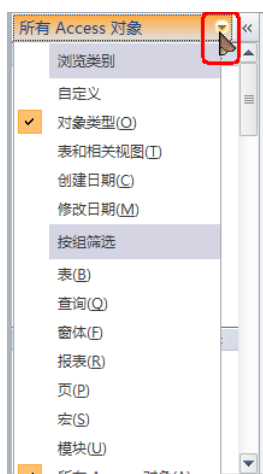


图 2-9 导航窗格中的对象分组

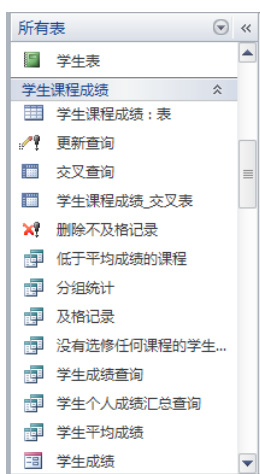


图 2-10 按【表和相關视图】查看结果

2.2.5 状态栏

状态栏位于程序窗口底部，用于显示状态信息。状态栏中还包含用于切换视图的按钮。如图 2-11 所示是表的【设计视图】中的状态栏。

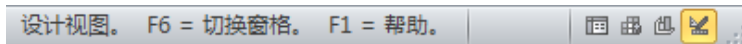
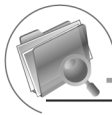


图 2-11 状态栏



2.3 Access 2010 的功能区

Access 2010 的功能区分为多个部分，其涵盖的功能类似于老版本中的菜单，下面将分别进行介绍。





2.3.1 显示或隐藏功能区

为了扩大数据库的显示区域，Access 2010 允许把功能区隐藏起来。显示功能区最简单的方法是单击当前的命令选项卡即可，若要再次打开功能区，只需再次单击命令选项卡即可；也可以单击功能区右端的按钮  隐藏功能区，单击按钮  展开功能区。

2.3.2 常规命令选项卡

在 Access 2010 的【功能区】中有 5 个常规命令选项卡，分别是【文件】、【开始】、【创建】、【外部数据】和【数据库工具】。每个选项卡下有不同的操作工具，用户可以通过这些工具，对数据库中的数据库对象进行设置。


1. 【文件】选项卡

【文件】选项卡是 Access 2010 新增加的一个选项卡。这是一个特殊的选项卡，它与其他选项卡的结构、布局和功能完全不同。单击【文件】选项卡，打开文件窗口，这是一个由【保存】、【打开】等一组命令按钮组成的菜单，如图 2-12 所示。选择不同的命令按钮，右侧窗格中将显示不同的信息。在文件窗口中，可对数据库文件进行各种操作和对数据库进行设置。

(1) 【信息】窗格

【信息】窗格提供了【启用内容】、【压缩和修复数据库】、【用户和权限】和【设置数据库】等操作，如图 2-13 所示。

(2) 【最近使用文件】窗格

【最近使用文件】窗格显示了最近打开的数据库文件。在最近打开的每个文件的后面有一个小按钮 ，单击这个按钮可以把该文档固定在打开的列表中，从而给用户方便。

(3) 【新建】窗格

【新建】窗格就是 Access 2010 的首界面，在这个窗格中可进行数据库的创建。

(4) 【打印】窗格

【打印】窗格是打印 Access 报表的操作界面，在该窗格中，有【快速打印】、【打印】和【打印预览】3 个操作。

(5) 【保存并发布】窗格

【保存并发布】窗格是保存和转换 Access 数据库文件的窗口，如图 2-14 所示。该窗口中间部分包括【数据库另存为】、【对象另存为】和【发布到 Access Services】3 个命令。右侧窗格中显示对应中间窗格每个命令的下一级命令信息。



图 2-12 【文件】选项卡菜单

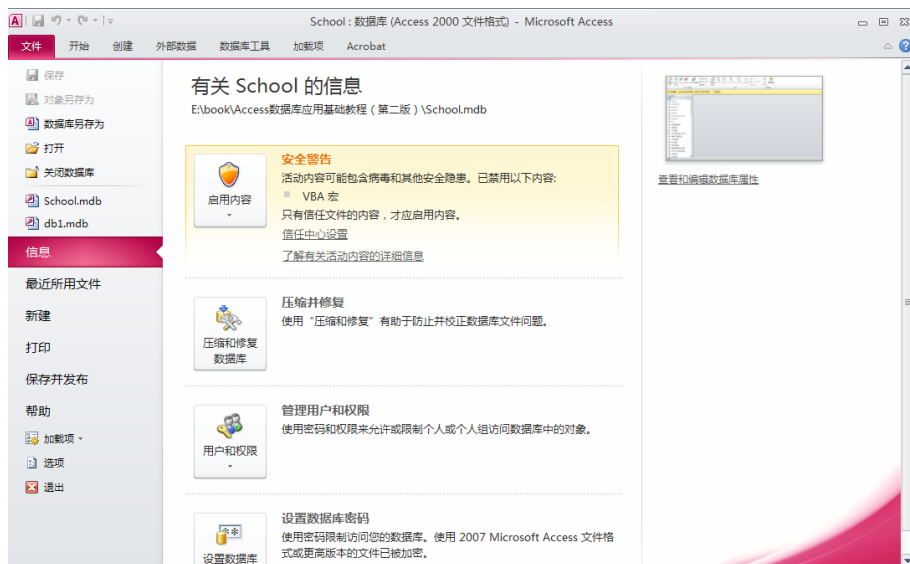
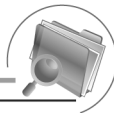


图 2-13 【信息】窗格



图 2-14 【保存并发布】窗格

(6) 【选项】窗格

单击【选项】按钮，将打开如图 2-15 所示的【Access 选项】对话框。通过该对话框，用户可以对 Access 进行个性化设置。

在【常规】选项中，可以更改默认文件格式，以便通过 Access 2010 创建与旧版本 Access 兼容的 MDB 文件。在【自定义功能区】选项中，可以对用户界面的一部分功能区进行个性化设置。例如，可以创建自定义选项卡和自定义组来包含经常使用的命令。在【快速访问工具栏】





选项中，可以自定义工具栏。

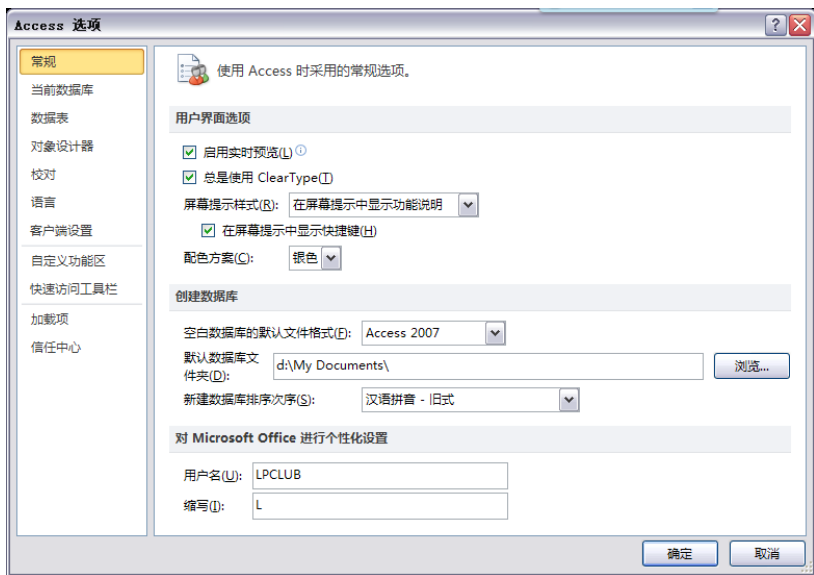


图 2-15 【Access 选项】对话框

2. 【开始】选项卡

在【开始】选项卡下，有如图 2-16 所示的一些工具组。利用这些工具，可以完成如下功能。

- 选择不同的视图。
- 从剪贴板复制和粘贴。
- 设置当前的字体格式。
- 设置当前的字体对齐方式。
- 对备注字段应用 RTF 格式。
- 操作数据记录(刷新、新建、保存、删除、汇总以及拼写检查等)。
- 对记录进行排序和筛选。
- 查找记录。

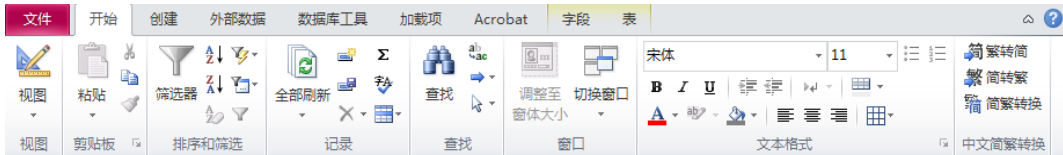


图 2-16 【开始】选项卡

3. 【创建】选项卡

【创建】选项卡如图 2-17 所示。用户可以利用该选项卡下的工具创建数据表、窗体、查询、报表和宏等各种数据库对象。



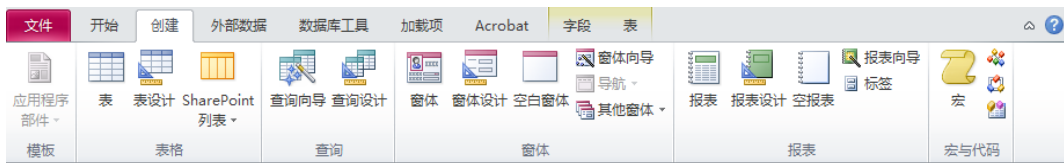
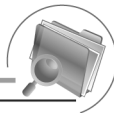


图 2-17 【创建】选项卡

利用【创建】选项卡下的工具，可以完成如下功能。

- 插入新的空白表。
- 使用表模板创建新表。
- 在 SharePoint 网站上创建列表，在链接到新创建的列表的当前数据库中创建表。
- 在设计视图中创建新的空白表。
- 基于活动表或查询创建新窗体。
- 基于活动表或查询创建新报表。
- 创建新的数据透视表或图表。
- 创建新的查询、宏、模块或类模板。

4. 【外部数据】选项卡

【外部数据】选项卡如图 2-18 所示，用户可以利用该选项卡下的工具导入和导出各种数据。

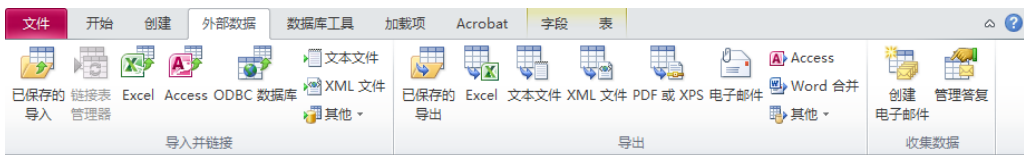


图 2-18 【外部数据】选项卡

利用【外部数据】选项卡下的工具，可以完成如下功能。

- 导入或链接到外部数据。
- 导出数据为 Excel、文本、XML 文件、PDF 等格式。
- 通过电子邮件收集和更新数据。
- 使用联机 SharePoint 列表。
- 将部分或全部数据库移至新的或现有的 SharePoint 网站。

5. 【数据库工具】选项卡

【数据库工具】选项卡如图 2-19 所示，用户可以利用该选项卡下的各种工具进行数据库 VBA、表关系的设置等。

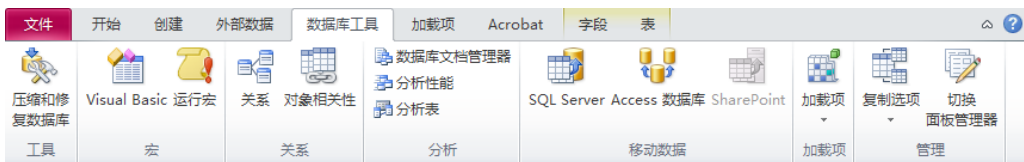


图 2-19 【数据库工具】选项卡





利用【数据库工具】选项卡下的工具，可以完成如下功能。

- 启动 Visual Basic 编辑器或运行宏。
- 创建和查看表关系。
- 显示/隐藏对象相关性或属性。
- 运行数据库文档或分析性能。
- 将数据移至 Microsoft SQL Server 或 Access(仅限于表)数据库。
- 运行链接表管理器。
- 管理 Access 加载项。
- 创建或编辑 VBA 模块。

2.3.3 上下文命令选项卡



上下文命令选项卡就是根据用户正在使用的对象或正在执行的任务而显示的命令选项卡。例如，当用户在数据视图下编辑一个数据表时，会出现【表格工具】下的【字段】选项卡和【表】选项卡，如图 2-20 和图 2-21 所示。



图 2-20 【表格工具】下的【字段】选项卡

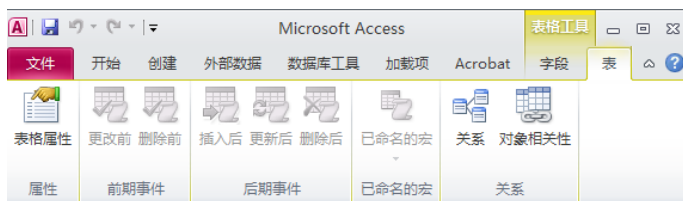


图 2-21 【表格工具】下的【表】选项卡

而在设计视图中设计一个数据表时，会出现【表格工具】及其下方的【设计】选项卡，如图 2-22 所示。

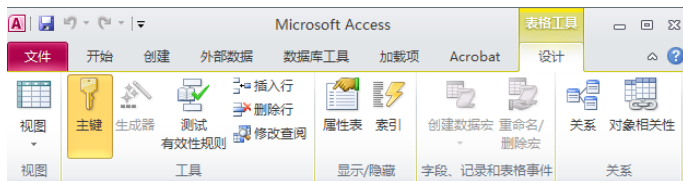
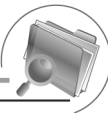


图 2-22 【表格工具】下的【设计】选项卡



在【查询】的设计视图中,会出现【查询工具】的【设计】选项卡,如图 2-23 所示。通过该选项卡可以设计不同类型的查询。



图 2-23 【查询工具】下的【设计】选项卡

2.3.4 自定义功能区

Access 2010 允许用户对界面的一部分功能区进行个性化设置。例如,可以创建自定义选项卡和自定义组来包含经常使用的命令。具体操作步骤如下。

(1) 单击【文件】功能区中的【选项】按钮,打开【Access 选项】对话框。单击【Access 选项】对话框左侧窗格中的【自定义功能区】选项,打开【自定义功能区】窗格,如图 2-24 所示。

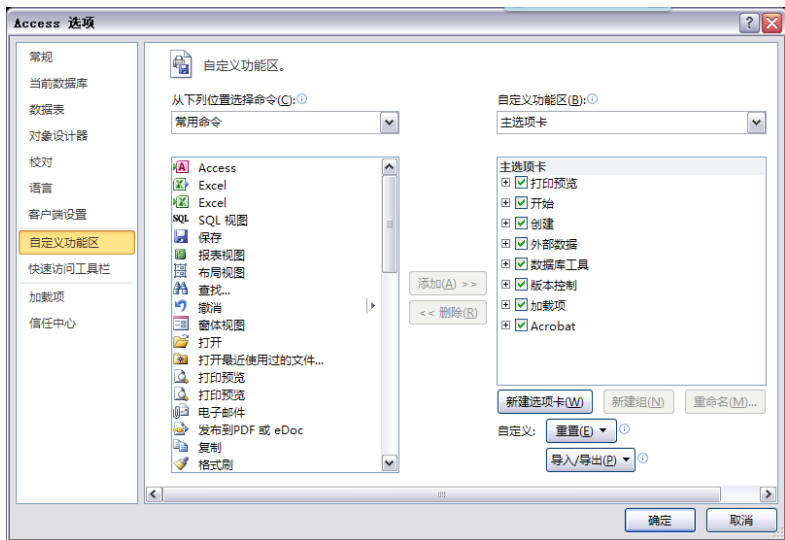


图 2-24 【自定义功能区】窗格

(2) 单击其中的【新建选项卡】按钮,【主选项卡】列表框中将会添加【新建选项卡(自定义)】和【新建组(自定义)】,如图 2-25 所示。

(3) 选中【主选项卡】列表框中的【新建选项卡(自定义)】,单击【重命名】按钮,打开【重命名】对话框,如图 2-26 所示,在对话框中输入“我的选项卡”,然后单击【确定】按钮。



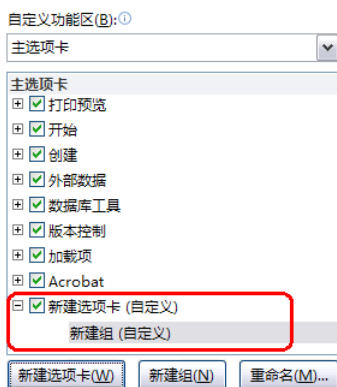


图 2-25 【新建选项卡】和【新建组】



图 2-26 【重命名】对话框

(4) 使用同样的方法，将【新建组(自定义)】重命名为“我的常用工具”，并为其选择一个自己喜欢的图标，如图 2-27 所示。

(5) 在【从下列位置选择命令】下拉列表中单击下拉箭头，选择【所有命令】。在下方的列表框中选择需要添加的命令即可，例如本例我们选择【用户级安全机制向导】、【用户与组权限】和【用户与组帐户】3 个命令，单击【添加】按钮，将它们添加到【我的选项卡(自定义)】的【我的常用工具(自定义)】组中，如图 2-28 所示。



图 2-27 重命名新建组

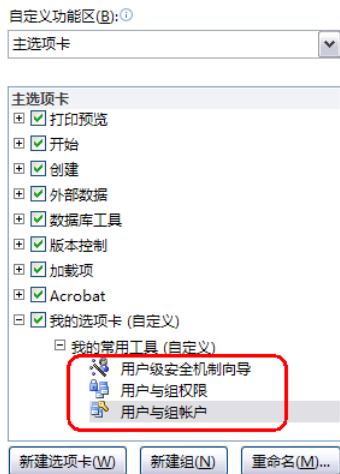
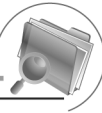


图 2-28 添加后的结果

(6) 单击【选项】对话框中的【确定】按钮，完成自定义功能区，此时可以看到，功能区多了一个选项卡【我的选项卡】，如图 2-29 所示。



图 2-29 自定义的功能区选项卡



2.4 Access 2010 数据库对象

数据库对象是 Access 最基本的容器对象，它是一些关于某个特定主题或目的的信息集合，具有管理本数据库中所有信息的功能。在数据库对象中，用户可以将自己的数据分别保存在各自独立的存储空间中，这些空间称为数据表；可以使用联机窗体来查看、添加和更新数据表中的数据；使用查询来查找并检索所需的数据；也可以使用报表以特定的版面布局来分析及打印数据。总之，创建一个数据库对象是应用 Access 建立信息系统的第一步工作。

早期的 Access 中有 7 种不同类别的数据库对象，即表、查询、窗体、报表、数据访问页、宏和模块。而 Access 2010 不再支持数据访问页对象。如果希望在 Web 上部署数据输入窗体并在 Access 中存储所生成的数据，则需要将数据库部署到 Microsoft Windows SharePoint Services 3.0 服务器上，使用 Windows SharePoint Services 所提供的工具实现所需的目标。

不同的对象在数据库中有着不同的作用：表是数据库的核心与基础，存放着数据库中的全部数据；报表、查询和窗体都是从数据库中获得数据信息，以实现用户的某一特定的需求，如查找、计算统计、打印、编辑修改等；窗体可以提供一种良好的用户操作界面，通过它可以直接或间接地调用宏或模块，并执行查询、打印、预览和计算等功能，甚至可以对数据库进行编辑修改操作。

需要注意的是，Access 2010 中不再支持数据访问页对象。

2.4.1 表

表是数据库中用来存储数据的对象，是整个数据库系统的基础。建立和规划数据库，首先要做的就是建立各种数据表。数据表是数据库中存储数据的唯一单位，它将各种信息分门别类地存放在各种数据表中。Access 允许一个数据库中包含多个表，用户可以在不同的表中存储不同类型的数据。通过在表之间建立关系，可以将不同表中的数据联系起来，以供用户使用。

表中的数据以行和列的形式保存，类似于 Excel 电子表格。表中的列称为字段，字段是 Access 信息的最基本载体，说明了一条信息在某一方面的属性。表中的每一行称为记录，记录是由一个或多个字段组成的。一条记录就是一个完整的信息。

在数据库中，应该为每个不同的主题建立不同的表，这样不但可以提高数据库的工作效率，还可以减少数据输入产生的错误。

2.4.2 查询

查询是数据库中应用得最多的对象之一。它可执行很多不同的功能，最常用的功能是从表中检索符合某种条件的数据。查询是数据库设计目的的体现，数据库创建完成后，数据只有被使用者查询使用才能真正体现它的价值。





查询是用来操作数据库中的数据记录，利用它可以按照一定的条件或准则从一个或多个表中筛选出需要的字段，并将它们集中起来，形成动态数据集，这个动态数据集就是用户想看到的来自一个或多个表中的字段，它显示在一个虚拟的数据表窗口中。用户可以浏览、查询、打印，甚至修改这个动态数据集中的数据，Access 会自动将所作的任何修改更新到对应的表中。执行某个查询后，用户可以对查询的结果进行编辑或分析，并可以将查询结果作为其他对象的数据源。

查询到的数据记录集合称为查询的结果集。结果集以二维表形式显示出来，但它们不是基本表。每个查询只记录该查询的查询操作方式，这样，每进行一次查询操作，其结果集显示的都是基本表中当前存储的实际数据，它反映的是查询的那个时刻数据表的情况，查询的结果是静态的。

查询对象的运行形式与数据表对象的运行形式几乎完全相同，但它只是数据表对象所包含数据的某种抽取与显示，本身并不包含任何数据。需要注意的是，查询对象必须建立在数据表对象之上。



2.4.3 窗体

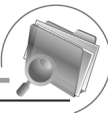
窗体是 Access 数据库对象中最灵活的一种对象，其数据源可以是表或查询。窗体有时被称为“数据输入屏幕”。窗体是用来处理数据的界面，通常包含一些可执行各种命令的按钮。可以说窗体是数据库与用户进行交互操作的最好界面。利用窗体，用户能够从表中查询、提取所需的数据，并将其显示出来。通过在窗体中插入宏，用户可以把 Access 的各个对象很方便地联系起来。

窗体的类型比较多，大致可以分为如下三类。

- 提示型窗体：主要用于显示文字和图片等信息，没有实际性的数据，也基本没有什么功能，主要用于作为数据库应用系统的主界面。
- 控制型窗体：使用该类型的窗体，可以在窗体中设置相应菜单和一些命令按钮，用于完成各种控制功能的转移。
- 数据型窗体：使用该类型的窗体，可以实现用户对数据库中相关数据进行操作的界面，这是 Access 数据库应用系统中使用得最多的窗体类型。

2.4.4 报表

数据库应用程序通常要打印输出数据，在 Access 中，如果要对数据库中的数据进行打印，使用报表是最简单且有效的方法。利用报表可以将数据库中需要的数据提取出来进行分析、整理和计算，并将数据以格式化的方式发送到打印机。可以在一个表或查询的基础上创建报表，也可以在多个表或查询的基础上创建报表。利用报表可以创建计算字段；还可以对记录进行分



组，以便计算出各组数据的汇总等。在报表中，可以控制显示的字段、每个对象的大小和显示方式，还可以按照所需的方式来显示相应的内容。

2.4.5 宏

Access 的宏对象是 Access 数据库中的一个基本对象。宏是指一个或多个操作的集合，其中每个操作实现特定的功能，例如打开某个窗体或打印某个报表。宏可以使某些普通的、需要多个指令连续执行的任务能够通过一条指令自动完成，而这条指令就称为宏。例如，可创建某个宏，在用户单击某个命令按钮时运行该宏，打印某个报表。因此，宏可以看作是一种简化的编程语言。利用宏，用户不必编写任何代码，就可以实现一定的交互功能。

通过宏，可以实现的功能主要有以下几项。

- 打开或关闭数据表、窗体、打印报表和执行查询。
- 弹出提示信息框，显示警告。
- 实现数据的输入和输出。
- 在数据库启动时执行操作等。
- 查找数据。



知识点

Microsoft Office 提供的所有工具中都提供了宏的功能。利用宏可以简化这些操作，使大量重复性操作自动完成，从而使管理和维护 Access 数据库更加简单。

宏可以是包含一个操作序列的宏，也可以是若干个宏的集合所组成的宏组。一个宏或宏组的执行与否还可以使用一个条件表达式是否成立予以判断，即可以通过给定的条件来决定在哪些情况下运行宏。

2.4.6 模块

模块对象是 Access 数据库中的一个基本对象。在 Access 中，不仅可以通过从宏列表中以选择的方式创建宏，还可以利用 VBA(Visual Basic for Applications)编程语言编写过程模块。

模块是将 VBA 的声明、语句和过程作为一个单元进行保存的集合，也就是程序的集合。创建模块对象的过程也就是使用 VBA 编写程序的过程。Access 中的模块可以分为类模块和标准模块两类。类模块中包含各种事件过程，标准模块包含与任何其他特定对象无关的常规过程。

尽管 Microsoft 在推出 Access 产品之初就将该产品定位为不用编程的数据库管理系统，而实际上，要在 Access 的基础上进行二次开发来实现一个数据库应用系统，用 VBA 编写适当的





程序是必不可少的。也就是说，若需要开发一个 Access 数据库应用系统，其间必然包括 VBA 模块对象。

2.5 上机练习

本章的上机练习将自定义快速访问工具栏，【快速访问工具栏】包含一组独立于当前显示的功能区上选项卡的命令。

(1) 单击【文件】功能区中的【选项】按钮，打开【Access 选项】对话框。

(2) 单击【Access 选项】对话框左侧窗格中的【快速访问工具栏】选项，打开【快速访问工具栏】窗格，如图 3-30 所示。

(3) 在【从下列位置选择命令】列表中选择【常用命令】，然后从命令列表中选择要添加到快速访问工具栏中的命令，这里我们选择【导出到 Excel 电子表格】，单击【添加】按钮，将其添加到右侧的列表中。

(4) 通过最右端的两个方向按钮，可以调整快速访问工具栏中各命令按钮的排列顺序。

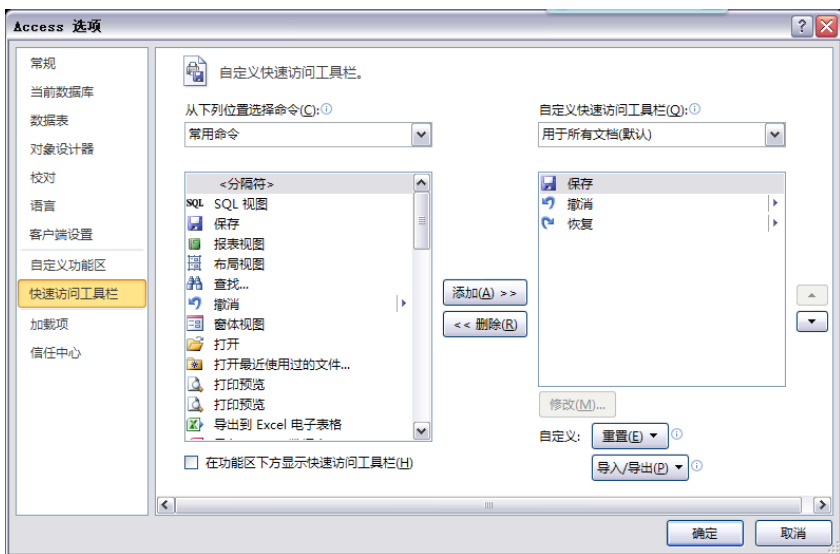


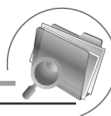
图 2-30 【快速访问工具栏】选项

(5) 单击【确定】按钮单关闭【Access 选项】对话框，此时的【快速访问工具栏】如图 2-31 所示。



图 2-31 快速访问工具栏

(6) 在 2.2.2 节，已经介绍过直接单击【快速访问工具栏】右侧的下拉按钮，通过弹出的快捷菜单快速定义【快速访问工具栏】，选择快捷菜单中的【其他命令】选项，也会打开【Access 选项】对话框，并定位到【快速访问工具栏】选项页。



2.6 习题

1. 练习启动和关闭 Access 2010。
2. Access 2010 的工作界面由哪几部分组成？
3. Access 2010 包含哪些数据库对象？简述它们的含义和功能。
4. 上机练习：设置 Access 选项。





数据库的创建与使用

学习目标

在 Access 中，数据库是一个容器，用于存储数据库应用系统中的任何对象，也就是说，构成数据库应用系统的对象都存储在数据库中。Access 2010 数据库是一个独立的数据库文件，扩展名为 .accdb。在 Access 数据库中，可以包含 6 种数据库对象。本章主要介绍创建数据库的方法以及数据库和数据库对象的基本操作。

本章重点

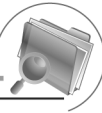
- Access 2010 数据库结构
- 从模板创建数据库
- 创建空白数据库
- 复制数据库对象
- 数据库对象的组织方式

3.1 Access 数据库概述

在第 1 章中曾经介绍过，数据库(Database)就是数据存储的位置，是针对特定的需求所整理与组织出的相关信息的汇集处。例如，病人的病历数据，网上订票的订单数据，银行的帐户交易记录等。在学习创建 Access 数据库之前，首先来了解一下 Access 数据库结构和 Access 数据库文件。

3.1.1 Access 数据库结构

Access 是关系型数据库。在 Access 数据库中，任何事物都可以称之为对象，也就是说，



Access 数据库由各种对象组成,包括表、查询、窗体、报表、数据访问页、宏和模块 7 种对象。其中,可以利用表对象存储信息,利用查询对象搜索信息,利用窗体查看信息,利用报表对象打印信息,利用页对象显示数据访问页信息,利用宏对象完成自动化工作,利用模块实现复杂功能。

此外,数据库也记录了字段和记录的验证规则、各个字段的标题和说明、各个字段的默认值、各个表的索引、各个关联表间的关联性、数据参考完整性等。Access 数据库具备存储、组织和管理各项相关信息的功能。

3.1.2 Access 数据库文件

由于 Access 数据库与传统的数据库概念有所不同,它采用特有的全环绕数据库文件结构组成数据库文件,因此,它可以以一个单独的数据库文件存储一个数据库应用系统中包含的所有对象。基于 Access 数据库文件的这一特点,创建一个 Access 数据库应用系统的过程就是创建一个 Access 数据库文件,并在其中设置和创建各种对象的过程。



知识点

Access 不同版本数据库文件的后缀名也不一样,早期的 Access 版本数据库文件的扩展名为 .mdb,最新的 Access 2010 的扩展名是 .accdb。

开发一个 Access 2010 数据库应用系统的第一步工作就是创建一个 Access 数据库文件,其操作的结果是在磁盘上建立一个扩展名为 .accdb 的数据库文件;第二步工作则是在数据库中创建数据表,并建立数据表之间的关系;接着创建其他对象,最终即可形成完备的 Access 2010 数据库应用系统。整个数据库应用系统仅以一个文件存储于文件系统中,显得极为简洁,使得该数据库应用系统的创建和发布变得非常简单。这也是很多小型数据库应用系统开发者偏爱 Access 的原因之一。

实际上,对于 Access 数据库管理系统来说,数据库是一级容器对象,其他对象均置于该容器对象之中,因此,数据库是其他对象的基础,即其他对象必须建立在数据库中,或者说,其他对象包含于特定的数据库中。

3.2 创建数据库

Access 提供了两种建立数据库的方法:一种是使用模板创建数据库,一种是创建空白数据库。使用模板创建数据库又分为样品模板、根据现有内容创建、我的模板、最近打开的模板以及从 Office.com 模板几种选择方式。另外,Access 2010 提供了两类数据库的创建,即 Web 数据库和传统数据库,本书以介绍传统数据库的创建和设计为主。





3.2.1 创建空白数据库

如果在数据库模板中找不到满足需要的模板，或在另一个程序中有要导入的 Access 数据，最好的办法就是创建一个空白数据库，这种方法适合于创建比较复杂的数据库，并且没有合适的数据库模板的情况。其实空白数据库就是建立的数据库的外壳，但是没有对象和数据而已。

空白数据库创建成功后，可以根据实际需要，添加所需要的表、窗体、查询、报表、宏和模块等对象。这种方法非常灵活，可以根据需要创建出各种数据库，但是由于用户需要自己动手创建各个对象，因此操作比较复杂。

【例 3-1】创建一个空白数据库 School.accdb。

(1) 启动 Access 2010，打开 Access 的启动窗口。在中间窗格上方的【可用模板】中选择【空数据库】，在右侧窗格的文件名文本框中，是默认的文件名 Database1.accdb，这里将数据库名称命名为 School.accdb，如图 3-1 所示。

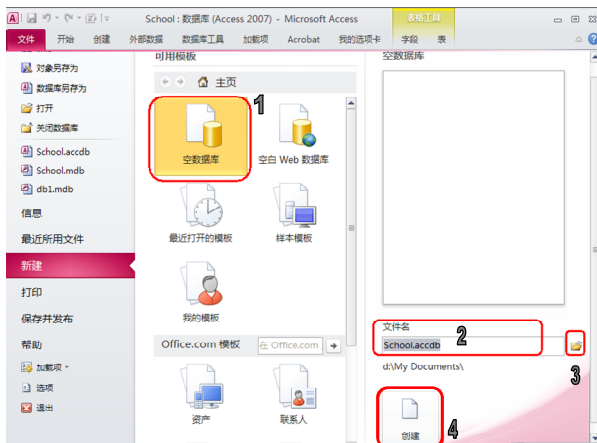



图 3-1 创建空白数据库

(2) 单击【浏览】按钮，在打开的【文件新建数据库】对话框中，选择数据库的保存位置。为了保证数据库的安全，创建的数据库最好不要保存在 Windows 系统盘。

(3) 在右侧窗格下面，单击【创建】按钮，即可创建一个空白数据库，并以数据工作表视图方式打开一个默认名为【表 1】的数据表，如图 3-2 所示。

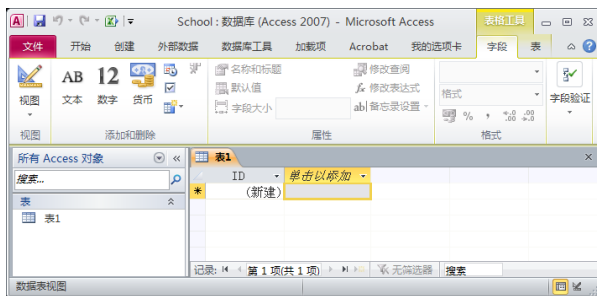
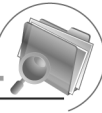


图 3-2 新建空白数据库【表 1】的数据工作表视图



(4) 空白数据库创建好以后, 就可以添加表和数据了, 用户可以在该空白数据库中逐一创建 Access 的各种对象。

3.2.2 使用模板创建数据库

使用模板创建数据库是创建数据库的最快方式, 用户只需要进行一些简单的操作, 就可以创建一个包含了表、查询等数据库对象的数据库系统。如果能找到并使用与要求最接近的模板, 此方法的效果最佳。除了可以使用 Access 提供的本地方法创建数据库之外, 还可以利用 Internet 网上的资源, 如果能在 Office.com 的网站上搜索到所需的模板, 就可以把模板下载到本地计算机中, 从而快速创建出所需的数据库。

【例 3-2】就利用 Access 2010 中的模板, 创建一个数据库。

【例 3-2】使用模板创建一个【联系人 Web 数据库】, 具体操作步骤如下。

(1) 启动 Access 2010, 打开 Access 的启动窗口。在启动窗口中的【可用模板】窗格中, 单击【样本模板】选项, 可以看到 Access 2010 提供的 12 个示例模板。这 12 个模板可以分成两组: 一组是传统数据库模板; 另一组是 Web 数据库模板。Web 数据库模板是 Access 2010 新增的功能, 可以使用户能够比较快地掌握 Web 数据库的创建。

(2) 本例中, 选择【联系人 Web 数据库】模板, 在右侧窗格的文件名文本框中自动生成一个默认的文件名【联系人 Web 数据库.accdb】, 保存位置默认在【我的文档】中。用户也可以自己指定文件名和文件保存的位置。

(3) 单击【创建】按钮, 完成数据库的创建。创建的数据库如图 3-3 所示。这个窗口中提供了配置数据库和使用数据库教程的链接; 此外, 如果计算机已经联网, 则单击▶按钮就可以播放相关教程。

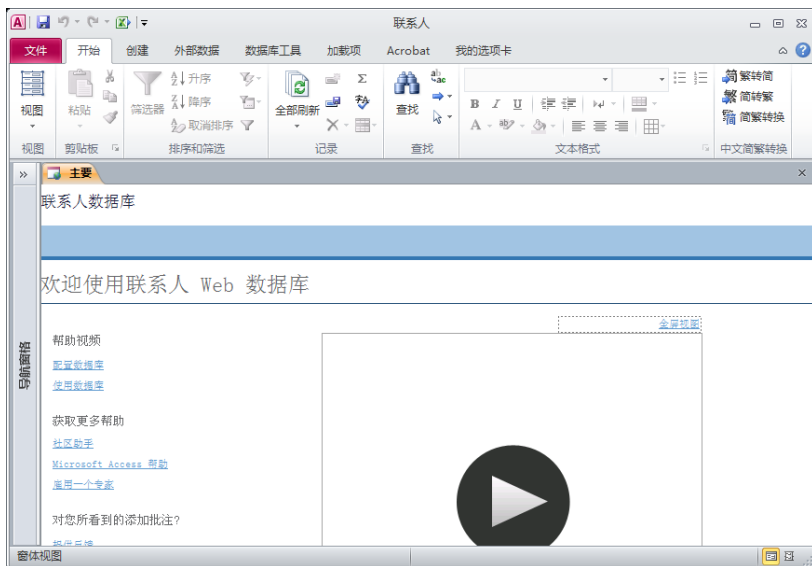


图 3-3 联系人 Web 数据库





(4) 展开【导航窗格】，可以查看该数据库包含的所有 Access 对象，如图 3-4 所示。

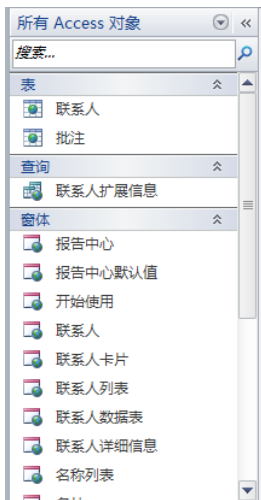


图 3-4 查看数据库中的所有对象



提示

通过数据库模板可以创建专业的数据库系统,但是这些系统有时不能够完全符合需求,因此最简单的方法就是先利用模板生成一个数据库,然后再进行修改,使其符合需求。



3.2.3 转换数据库

Access 具有不同的版本,可以将使用 Microsoft Office Access 2003、Access 2002、Access 2000 或 Access 97 创建的数据库转换成 Access 2007-2010 文件格式(.accdb)。此文件格式支持新的功能,如多值字段和附件。

这种新型文件格式(.accdb)的数据库不能用早期版本的 Access 打开,也不能与其链接,而且此新型格式不再支持复制或用户级安全性。如果需要在早期版本的 Access 中使用新型文件格式的数据库,或者需要使用复制或用户级安全性,则必须将其转换为早期版本的文件格式。



知识点

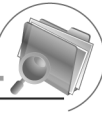
可以使用 SharePoint Server 的新组件 Access Services 将使用新文件格式的数据库发布到 Web。不可使用 Access Services 将早期文件格式的数据库发布到 Web。

1. 转换 Access 2000 或 Access 2002-2003 数据库

若要将 Access 2000 或 Access 2002-2003 数据库(.mdb)转换成新型文件格式(.accdb),则必须先 Access 2010 中打开该数据库,然后将其保存为 .accdb 文件格式。具体操作步骤如下。

(1) 在【文件】选项卡上,单击【打开】命令按钮。

(2) 在【打开】对话框中,选择要转换的数据库并将其打开。如果出现【数据库增强功能】对话框,则表明数据库使用的文件格式早于 Access 2000。



(3) 在【文件】选项卡上,单击【保存并发布】命令按钮,然后在【数据库文件类型】中选择【Access 数据库(*.accdb)】选项。

提示

在使用【将数据库另存为】命令时,如果有数据库对象处于打开状态,Access 将会提示用户在创建副本之前关闭这些数据库对象。单击【是】按钮以让 Access 关闭对象,或者单击【否】以取消整个过程。

(4) 单击【另存为】按钮,将打开【另存为】对话框,在【文件名】文本框中输入文件名,然后单击【保存】按钮即可完成转换,Access 将创建数据库副本并打开该副本。

2. 转换 Access 97 数据库

在 Access 2010 中打开 Access 97 数据库时,会出现【数据库增强功能】对话框。若要将此版本的数据库转换为 Access 2007 文件格式,则在该对话框中单击【是】按钮。Access 随后会以 .accdb 格式创建此数据库的副本。

知识点

在早于 Office Access 2007 的版本中无法使用 .accdb 文件格式,使用同样地操作,可以在 Access 2010 中将 .accdb 数据库另存为 Access 2000 或 Access 2002-2003 数据库。

3.3 操作数据库和数据库对象

创建了数据库之后,要使用数据库时就需要打开创建好的数据库;可以在数据库中进行创建数据库对象、修改已有对象等操作;当数据库不用时要关闭数据库。这些都是数据库的基本操作。

3.3.1 打开数据库

打开数据库是数据库操作中最基本、最简单的操作。打开一个已经存在的数据库,一般操作步骤如下。

(1) 启动 Access 2010,单击功能区的【文件】选项卡,选择【打开】命令,打开【打开】对话框,如图 3-5 所示。

(2) 在该对话框中选择需要打开的数据库文件,接着单击【打开】按钮旁的三角符号按钮,弹出一个下拉菜单,从中选择数据库的打开方式。



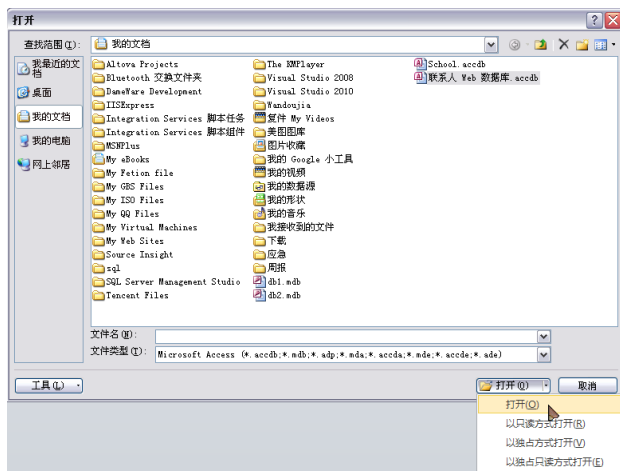


图 3-5 【打开】对话框



下面简单介绍以不同方式打开数据库的不同功能。

- **【打开】**：以这种方式打开数据库，就是以共享模式打开数据库，即允许多位用户在同一时间同时读写数据库。
- **【以只读方式打开】**：以这种方式打开数据库，只能查看而无法编辑数据库。
- **【以独占方式打开】**：以这种方式打开数据库时，当有一个用户在读写数据库，则其他用户都无法使用该数据库。
- **【以独占只读方式打开】**：如果要只读且独占的模式来打开数据库，则选择该选项。所谓的**【独占只读方式】**指在一个用户打开某一个数据库后，其他用户将只能以只读模式打开此数据库，而并非限制其他用户都不能打开此数据库。



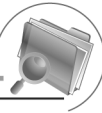
知识点

要成功地以**【独占只读模式】**打开某一个数据库，前提条件是此数据库目前必须尚未被其他用户以非只读方式打开。如果某一个数据库已被其他用户以非只读方式打开，则当尝试以独占只读方式去打开它时，Access 会以单纯的只读方式来打开它。

除了以上常规的打开数据库的方式之外，Access 还会自动记忆用户最近打开过的数据库。对于最近使用过的数据库文件，只需要单击**【文件】**选项卡，并选择**【最近所用文件】**命令，接着在右侧窗格中直接单击要打开的数据库即可。

3.3.2 数据库对象的组织

Access 提供了导航窗格对数据库对象进行组织和管理。利用导航窗格可以对 Access 中的



表、查询、窗体、报表、宏和模块等对象进行管理。

在导航窗格中，可以采用多种方式对数据库对象进行组织，以便高效地管理数据库对象。这些组织方式包括对象类型、表和相关视图、创建日期、修改日期、按组筛选、按对象类别以及自定义。

1. 对象类型

在导航窗格的上部，单击【所有对象】右侧的下拉箭头，即可打开组织方式列表，选择不同的【浏览类型】。

对象类型就是按照表、查询、窗体、报表、宏和模块等对象组织数据，这种组织方式和之前版本的组织方式相同。在对象类别中，选择其中一个对象，比如【表】，导航窗格将只显示数据库中所有的表。

2. 表和相关视图

表和相关视图是 Access 采用的一种新的组织方式。这种组织方式是基于数据库对象的逻辑关系而组织起来的。

在 Access 数据库中，数据表是最基本的对象，其他对象都是基于表作为数据源而创建的。因此，这些对象与某个表相关的对象就构成了逻辑关系，通过这种组织方式，可以使 Access 数据库开发者比较容易了解数据库内部对象之间的关系。

如果在创建数据库各个对象的过程中，没有采用表和相关视图的方式来组织对象，那么可以在数据库完成后进行组织。具体操作步骤如下。

(1) 打开指定的数据库。

(2) 在导航窗格中，单击【所有对象】右侧的下拉箭头，从打开的快捷菜单中选择【表和相关视图】选项。

(3) Access 开始对数据库对象进行组织(若数据库比较大，可能需要花费一定的时间)。

3. 自定义

自定义是一种灵活的组织方式，允许 Access 数据库开发者根据用户的需要组织数据库中的对象。例如，如果一个主窗体包含两个子窗体，那么，可以把该主窗体与这两个子窗体组织在一起；或把两个相关的查询组织在一起。

自定义数据库对象组织方式的具体操作步骤如下。

(1) 打开数据库，在导航窗格中选择【自定义】选项，如图 3-6 所示。

(2) 此时，将创建一个自定义组，在导航窗格中，把需要的对象拖到【自定义组 1】中，如图 3-7 所示。

(3) 如果需要对自定义分组重命名，可以在【自定义组 1】上单击鼠标右键，从弹出的快捷菜单中选择【重命名】命令，自定义分组的名称则处于可编辑状态，如图 3-8 所示。此时可对分组重新命名。



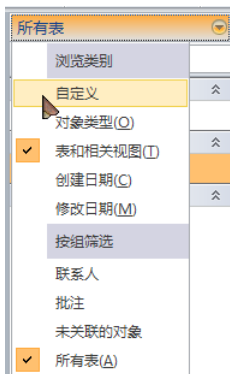


图 3-6 自定义组织方式

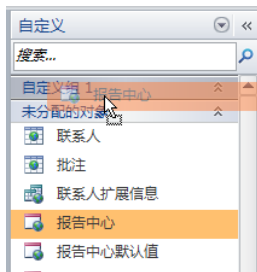


图 3-7 拖动对象到自定义组 1

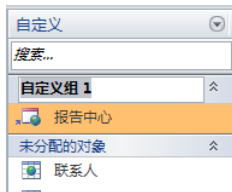


图 3-8 重命名自定义组

3.3.3 操作数据库对象



打开数据库之后，就可以创建、修改和删除数据库中的对象，对数据库对象的操作包括创建、打开、复制、删除、修改和关闭等。本节只介绍基本的打开、复制、删除和关闭操作，其他的操作将在后续章节中详细介绍。

1. 打开数据库对象

如果需要打开一个数据库对象，可以在导航窗格中选择一种组织方式，找到要打开的对象，然后双击即可直接打开该对象。

【例 3-3】 打开【联系人 Web 数据库】中的【联系人】窗体。

- (1) 打开【联系人 Web 数据库】数据库，在导航窗格中，单击【所有 Access 对象】右侧的下拉箭头。
- (2) 从弹出的快捷菜单中选择【对象类型】命令。
- (3) 在展开的对象列表中，双击【联系人】窗体图标即可。



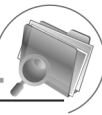
图 3-9 窗体对象的快捷菜单



知识点

右击【联系人】窗体图标，从弹出的快捷菜单中，选择【打开】命令，也可打开该窗体，如图 3-9 所示。

打开的【联系人】窗体将显示在选项卡式文档窗格中。从导航窗格打开的所有对象都显示



在文档窗格中。如果打开了多个对象，在选项卡式文档窗格中，只要单击相应的选项卡名称，就可以把相应的对象显示出来。

2. 复制数据库对象

在 Access 数据库中，使用复制方法可以创建对象的副本。通常在修改某个对象的设计之前，需要创建对象的副本，这样可以避免因修改操作错误造成数据丢失，一旦发生错误还可以用副本还原对象。

【例 3-4】复制【联系人 Web 数据库】数据库中的【联系人通讯簿】报表。

- (1) 启动 Access 2010，打开数据库【联系人 Web 数据库】。
- (2) 在【导航窗格】中，单击【所有 Access 对象】右侧的下拉箭头，从打开的组织方式下拉列表中，选择【对象类型】命令。
- (3) 找到【联系人通讯簿】报表，右击【联系人通讯簿】，从弹出的快捷菜单中选择【复制】命令，如图 3-10 所示。
- (4) 在【导航窗格】的空白处右击，从弹出的快捷菜单中选择【粘贴】命令，如图 3-11 所示。
- (5) 此时将打开【粘贴为】对话框，在该对话框中可以为复制的对象重新命名，或者使用默认的名称，如图 3-12 所示，确认名称后单击【确定】按钮。

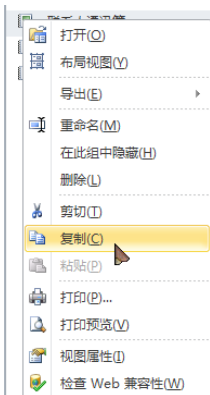


图 3-10 复制 Access 对象

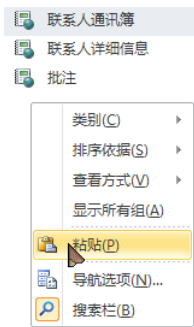


图 3-11 粘贴数据库对象



图 3-12 【粘贴为】对话框



提示

如果要把对象粘贴到另外一个数据库，则在执行复制操作后要把当前数据库关闭，然后打开另外的数据库再执行粘贴操作。

3. 关闭对象

Access 使用选项卡方式管理所有打开的数据库对象。当打开多个对象后，如果需要关闭某个对象，最简单的方法就是，首先选中想要关闭的对象，然后在选项卡对象窗格中单击右上角的 按钮，即可将该对象关闭。





4. 删除数据库对象

如果要删除某个数据库对象，需要先关闭该数据库对象。在多用户的环境中，还要确保所有的用户都已经关闭了该数据库对象。

【例 3-5】删除【例 3-4】中创建的【联系人通讯簿 的副本】报表。

- (1) 如果【联系人通讯簿 的副本】报表被打开，需要先关闭该报表。
- (2) 在【导航窗格】中找到要删除的数据库对象【联系人通讯簿 的副本】报表。
- (3) 右击该数据库对象，从弹出的快捷菜单中选择【删除】命令。或者按 Delete 键，选中的对象就被删除了。

(4) 在执行删除命令之前，Access 会弹出提示对话框要求用户确认是否真的删除，如图 3-13 所示，单击【是】按钮即可完成删除。

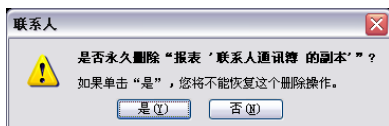


图 3-13 信息提示对话框



3.3.4 保存数据库

对数据库做了修改以后，需要及时的保持数据库，才能永远保存所做的修改操作。

保存数据库的具体操作步骤如下。

- (1) 单击【文件】选项卡，选择【保存】命令，即可保存对当前数据库的修改。
- (2) 选择【数据库另存为】命令，可更改数据库的保存位置和文件名，使用该命令时，Access 会弹出提示框，提示用户在保存数据库前必须关闭所有打开的对象，单击【是】按钮即可，如图 3-14 所示。

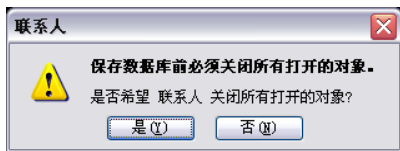


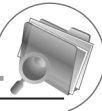
图 3-14 提示对话框

- (3) 在打开的【另存为】对话框中，选择文件的保存位置，然后在【文件名】文本框中输入文件名称，单击【保存】按钮即可。



提示

还可以通过单击快速访问工具栏中的【保存】按钮或按 Ctrl + S 组合键来保存编辑后的文件。



3.3.5 关闭数据库

当不再需要使用数据库时，可以将数据库关闭。关闭数据库的具体操作步骤如下。

- (1) 单击窗口右上角的关闭按钮，即可关闭数据库。
- (2) 单击【文件】选项卡，选择【关闭数据库】命令，也可关闭数据库。

3.4 上机练习

本章的上机练习将复制表对象，并介绍【粘贴表方式】的几种粘贴选项。

- (1) 在【导航窗格】中，按【对象类型】方式组织数据库对象。
- (2) 选择【表】命令，筛选出所有表对象，如图 3-15 所示。
- (3) 选择【联系人】表并右击，从弹出的快捷菜单中选择【复制】命令。
- (4) 在【导航窗格】的空白处，单击鼠标右键，从弹出的快捷菜单中选择【粘贴】命令，将弹出【粘贴表方式】对话框，如图 3-16 所示。

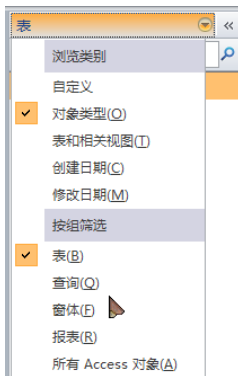


图 3-15 筛选【表】对象

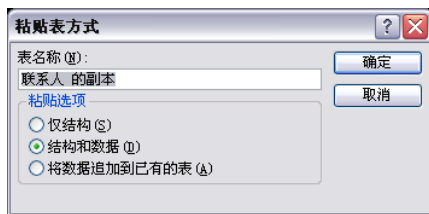


图 3-16 【粘贴表方式】对话框

各粘贴选项的含义如下。

- 【仅结构】：只粘贴表的结构，而不复制源表中的数据。
- 【结构和数据】：按表的结构建立新表，并把源表的数据粘贴到新表中。
- 【将数据添加到已有表】：在【表名称】文本框中输入需要追加数据的表名称，该表必须是已经存在的，然后将源表中的数据，追加到该表中。

(5) 本例中，选择【结构和数据】单选按钮，单击【确定】按钮完成表结构和数据的复制。此时的【导航窗格】如图 3-17 所示。

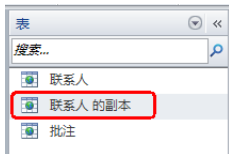
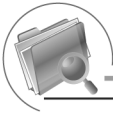


图 3-17 【导航窗格】窗口





3.5 习题

1. Access 2010 数据库文件的扩展名是什么？
2. Access 数据库由哪几种对象组成？
3. 利用模板建立一个【学生】数据库。
4. 练习数据库的打开、保存和关闭操作。
5. 请列举几种常见的数据库操作。

