



工业和信息化普通高等教育“十二五”规划教材

21世纪高等教育计算机规划教材

COMPUTER

Access 2010 数据库技术与应用

Access 2010 Database Application

■ 罗娜 蒲东兵 韩毅 王红岩 编著

注重理论与实践相结合

多年从事一线教学的经验积累

紧扣全国计算机等级考试（二级）大纲



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化普通高等教育“十二五”规划教材
21世纪高等教育计算机规划教材

COMPUTER

Access 2010 数据库技术与应用

Access 2010 Database Application

■ 罗娜 浦东兵 韩毅 王红岩 编著



人民邮电出版社
北京

TP 311.138Ac

204

图书在版编目 (C I P) 数据

Access 2010数据库技术与应用 / 罗娜等编著. --
北京: 人民邮电出版社, 2014.6
21世纪高等教育计算机规划教材
ISBN 978-7-115-35341-2

I. ①A… II. ①罗… III. ①关系数据库系统—程序设计—高等学校—教材 IV. ①TP311.138

中国版本图书馆CIP数据核字(2014)第079454号

内 容 提 要

本书通过大量的例题,讲解了 Access 数据库技术的相关知识以及使用 Access 2010 开发数据库应用系统的完整过程。全书共分 9 章,主要内容包括数据库系统的基础知识、数据库操作、表、查询、窗体、报表、宏、VBA 编程基础等相关知识。全书以一个完整的数据库应用系统为基础,以例题贯穿始终,书后配有适量的习题,使读者能够在学习过程中提高操作能力和实际应用能力。

本书可作为高等学校非计算机专业计算机公共基础课程的教材,也可以作为全国计算机等级考试二级 Access 数据库程序设计的培训与自学教材,还可以作为数据库开发人员的参考用书。

◆ 编 著 罗 娜 蒲东兵 韩 毅 王红岩

责任编辑 许金霞

责任印制 彭志环 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京铭成印刷有限公司印刷

◆ 开本: 787×1092 1/16

印张: 19.5

2014 年 6 月第 1 版

字数: 523 千字

2014 年 6 月北京第 1 次印刷

定价: 42.00 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

前言

计算机的发展极大地加快了社会信息化的进程,数据库技术于 20 世纪 60 年代末作为数据管理的最新技术登上了历史舞台。几十年来,数据库技术作为计算机软件领域的一个重要分支,已形成相当规模的理论体系和实用技术。Access 2010 数据库管理系统是 Microsoft Office 办公软件的一个组成部分,是世界上最流行的桌面数据库管理系统。它提供了大量的工具和向导,即使没有任何编程经验,也可以通过可视化的操作来完成大部分的数据库管理和开发工作。与许多优秀的关系数据库管理系统一样,Access 数据库可以有效地组织、管理和共享数据库的信息,并且能方便地将数据库与 Web 结合在一起。

本书结合《全国计算机等级考试二级 Access 数据库程序设计》的考试大纲,以 Access 2010 中文版作为数据库及其应用程序设计的工具和开发环境,从数据库的基础知识讲起,由浅入深、循序渐进地介绍了 Access 2010 各种数据库对象的功能及创建方法,以及宏和 VBA 面向对象程序设计的基础知识,全书共分为 9 章。

第 1 章介绍数据库的基础知识,包括数据模型、关系型数据库、关系运算和数据库系统设计的一般步骤。

第 2 章介绍 Access 2010 方面的概述性知识。

第 3 章介绍 Access 数据库和数据表的各种创建方法、设置字段的常规属性、建立表间关系、建立查阅列和表的操作等内容。

第 4 章介绍如何利用 Access 2010 创建和编辑选择查询、参数查询、交叉表查询、操作查询和 SQL 查询。

第 5 章介绍创建窗体的各种方法以及对窗体的再设计,并介绍了作为窗体的基本控件的功能及其属性。

第 6 章介绍创建报表的各种方法,创建报表的计算字段、报表中的数据排序与分组、报表的美化操作等。

第 7 章介绍宏的基本概念、创建、调试和运行。

第 8 章介绍 VBA 编程的基础知识和模块的相关概念。

第 9 章介绍 VBA 的程序设计方法及编程中的事件处理机制,数据库访问技术等。

本书具有如下特点:

(1) 步骤清晰,易懂易学——本书针对 Access 2010 的初学者,对操作中的每一步骤都进行了详细的讲解和说明,读者可以通过一边学习,一边实践的方式掌握 Access 数据库技术及其应用系统开发的方法。

(2) 问题驱动,图文并茂——本书内容的编排上体现了新的计算机教学思想和方法,以例题的提出、问题的分析、求解的步骤、归纳总结的一系列模式介绍数据库技术的基本内容与基本方法。例题的讲解中采用一步一图的讲述方式,每一个具体步骤都配以清晰的图片说明,确保读者可以看图操作。

(3) 内容呼应, 总结提高——本书各章节的例题都有着前后呼应的效果, 上一章节讲解的知识作为下一章节例题的基础反复地应用, 使读者对所学的知识能够进一步地掌握和应用。本书的每章都有对知识点的总结与归纳, 在章节最后总结教学重点和教学要点。

(4) 知识拓展, 学考统一——本书内容中有学生选学内容, 这部分内容可以拓展学生的知识面, 了解有一定难度的相关的技术与知识。本书每章都按计算机等级考试二级 Access 考试大纲进行内容的组织, 免去了学生为了等级考试再去购置其他教材的麻烦, 实现了学习、考试的统一。

本书可作为高等学校非计算机专业计算机公共基础课程的教材, 也可以作为全国计算机等级考试二级 Access 数据库程序设计的培训与自学教材, 还可以作为数据库开发人员的参考用书。

全书的第1章~第3章由罗娜编写, 第4章和第7章由蒲东兵编写, 第5章和第6章由王红岩编写, 第8章和第9章由韩毅编写。全书由张靖波统稿, 韩文峰审定。

由于编者水平有限, 时间仓促, 书中难免有疏漏和不妥之处, 敬请广大读者批评指正。

编者

2014年1月

目 录

第 1 章 数据库基础知识	1	3.1.2 创建空数据库	26
1.1 数据、信息与数据处理	1	3.1.3 数据库对象的组织	27
1.1.1 数据与信息	1	3.1.4 数据库的打开和关闭	27
1.1.2 数据处理	2	3.2 表的概念	29
1.1.3 数据管理技术的发展	2	3.2.1 表的组成	29
1.2 数据模型	4	3.2.2 命名规则	29
2.2.1 数据描述	4	3.2.3 字段的数据类型	30
2.2.2 概念模型	6	3.3 创建表	32
2.2.3 数据模型	6	3.3.1 使用数据表视图创建数据表	32
1.3 数据库系统	7	3.3.2 使用表设计创建数据表	34
1.3.1 数据库及其特点	7	3.3.3 设置主键	35
1.3.2 数据库管理系统	8	3.3.4 设置字段属性	36
1.3.3 数据库系统	9	3.3.5 数据的输入	48
1.4 关系数据库	9	3.3.6 数据的导入	53
1.4.1 关系术语	10	3.3.7 创建表间关系	61
1.4.2 对关系的限制	10	3.4 表的维护	64
1.4.3 完整性约束条件	11	3.4.1 表结构的修改	64
1.4.4 关系运算	11	3.4.2 编辑数据表中的记录	65
1.5 数据库设计基础	14	3.4.3 表的修饰	67
1.5.1 设计原则	14	3.5 表的数据操作	71
1.5.2 设计步骤	15	3.5.1 查找和替换数据	71
习题 1	16	3.5.2 记录排序	74
第 2 章 Access 2010 数据库概述	17	3.5.3 记录筛选	76
2.1 Access 2010 概述	17	习题 3	79
2.2 Access 2010 的新特点	17	第 4 章 数据查询	81
2.3 Access 2010 的操作环境	19	4.1 查询概述	81
2.3.1 Access 的启动与退出	19	4.1.1 查询的作用	81
2.3.2 Access 的工作界面	19	4.1.2 查询的类型	82
2.4 数据库对象	21	4.1.3 设置查询条件	83
习题 2	22	4.1.4 查询的视图	88
第 3 章 数据库与数据表	23	4.2 选择查询	91
3.1 数据库的创建	23	4.2.1 使用查询向导创建选择查询	91
3.1.1 使用模板创建数据库	23	4.2.2 使用设计视图创建选择查询	96
		4.2.3 创建带条件的查询	98

4.2.4 在查询中进行计算	99	5.3.3 常用控件及其使用	142
4.3 参数查询	104	5.3.4 常用控件属性	148
4.3.1 单参数查询	104	5.4 窗体布局 and 美化	155
4.3.2 多参数查询	105	5.4.1 窗体布局	155
4.4 交叉表查询	107	5.4.2 设置主题	156
4.4.1 使用向导创建交叉表查询	107	5.4.3 设置条件格式	156
4.4.2 使用设计视图创建交叉表查询	108	5.4.4 设置提示信息	157
4.5 操作查询	110	5.5 系统控制窗体	157
4.5.1 生成表查询	110	5.5.1 创建切换窗体	157
4.5.2 追加查询	111	5.5.2 创建导航窗体	163
4.5.3 删除查询	112	5.5.3 创建启动窗体	164
4.5.4 更新查询	112	习题 5	164
4.6 SQL 查询	113	第 6 章 报表	166
4.6.1 SQL 概述	113	6.1 认识报表	166
4.6.2 创建 SQL 查询	114	6.1.1 报表的组成	167
4.6.3 创建数据定义查询	117	6.1.2 报表的分类	168
4.6.4 SQL 数据操作功能	119	6.1.3 报表的视图	168
4.6.5 使用联合查询与子查询	120	6.2 报表的创建与编辑	169
4.7 对查询的操作	121	6.2.1 自动创建报表	169
4.7.1 编辑查询的数据源	121	6.2.2 报表向导	170
4.7.2 编辑查询的字段	121	6.2.3 标签报表	171
4.7.3 对查询结果排序	122	6.2.4 空报表	173
习题 4	122	6.2.5 设计报表	175
第 5 章 窗体	124	6.2.6 建立主/子报表	176
5.1 窗体概述	124	6.3 编辑报表	179
5.1.1 窗体功能	124	6.3.1 添加日期和时间	179
5.1.2 窗体类型	125	6.3.2 添加分页符和页码	180
5.1.3 窗体视图	128	6.3.3 添加直线和矩形	181
5.1.4 窗体构成	129	6.4 报表的高级功能	182
5.2 创建窗体	130	6.4.1 报表的排序与分组	182
5.2.1 自动创建窗体	131	6.4.2 添加计算控件	184
5.2.2 使用窗体向导创建窗体	132	6.4.3 报表统计计算	186
5.2.3 创建数据表窗体	133	6.4.4 报表中的常用函数	187
5.2.4 创建数据透视表窗体	134	6.5 报表的打印	187
5.2.5 创建数据透视图窗体	135	6.5.1 报表的打印预览	188
5.2.6 创建主/子窗体	136	6.5.2 报表的页面设置和打印	188
5.3 窗体常见控件及其属性设计	138	习题 6	189
5.3.1 控件类型	138	第 7 章 宏的设计与应用	191
5.3.2 操作控件对象	141	7.1 宏的概述	191

7.1.1 宏的概念及类型	191	8.6.1 过程的定义与调用	244
7.1.2 宏设计视图	192	8.6.2 参数传递	247
7.2 创建与设计宏	193	8.6.3 过程的作用域	249
7.2.1 创建操作序列宏	193	8.7 面向对象程序设计的基本概念	250
7.2.2 创建条件操作宏	195	8.7.1 对象及对象三要素	250
7.2.3 创建宏组	197	8.7.2 常用窗体、控件事件	251
7.3 宏的运行与调试	198	8.7.3 DoCmd 对象	255
7.3.1 运行宏	198	8.7.4 文件操作	258
7.3.2 调试宏	198	8.8 VBA 程序调试	262
7.4 事件触发宏	199	8.8.1 VBA 程序错误的分类	262
习题 7	201	8.8.2 断点及其设置	263
第 8 章 模块与 VBA 程序设计	203	8.8.3 调试工具与使用	263
8.1 VBA 的编程环境	203	8.8.4 VBA 错误处理	265
8.1.1 进入 VBE 编辑器	203	习题 8	266
8.1.2 VBE 编辑环境简介	205	第 9 章 VBA 数据库编程	276
8.2 模块的基本概念	206	9.1 VBA 数据库编程基础	276
8.2.1 标准模块	206	9.1.1 数据库引擎及接口	276
8.2.2 类模块	207	9.1.2 数据访问对象 (DAO)	277
8.3 创建模块	208	9.1.3 ActiveX 数据对象 (ADO)	278
8.3.1 在模块中加入过程	208	9.1.4 域聚合函数	283
8.3.2 将宏转换为模块	208	9.2 VBA 数据库编程应用举例	284
8.3.3 在模块中执行宏	209	9.2.1 记录的浏览	284
8.4 VBA 程序设计基础	209	9.2.2 记录的查询	286
8.4.1 VBA 基本语句	209	9.2.3 添加新记录和记录更新	287
8.4.2 VBA 中的数据类型	211	9.2.4 删除记录	288
8.4.3 常量与变量	213	习题 9	288
8.4.4 运算符与表达式	219	附录 1 常用函数	291
8.4.5 常用函数	221	附录 2 窗体属性及其含义	293
8.4.6 数据类型之间的转换	228	附录 3 控件属性及其含义	295
8.5 VBA 程序的流程控制	229	附录 4 常用宏命令	297
8.5.1 顺序结构	230	附录 5 常用事件	300
8.5.2 选择结构	230	参考文献	302
8.5.3 循环结构	236		
8.6 VBA 过程与作用域	243		

第 1 章

数据库基础知识

随着科学技术和社会经济的飞速发展,人们掌握的信息量急剧增加,要充分地开发和利用这些信息资源,就必须有一种新技术能对大量的信息进行处理。数据库技术是从 20 世纪 60 年代末作为数据处理的一门技术而发展起来的,所研究的问题就是如何科学地组织和存储数据,如何高效地获取和处理数据。本章主要介绍数据库的基础知识,相关知识点如下:

- ◆ 数据、信息与数据处理的概
- ◆ 数据模型
- ◆ 数据库的概念及特点
- ◆ 数据库管理系统的概念及主要功能
- ◆ 数据库系统的概念及组成
- ◆ 关系数据库理论
- ◆ 数据库设计基础

1.1 数据、信息与数据处理

数据库系统的核心任务是数据管理。数据库技术是一门研究如何存储、使用和管理数据的技术,是计算机数据管理技术的最新发展阶段。数据库应用涉及数据、信息、数据处理和数据管理等相关概念。

1.1.1 数据与信息

1. 数据 (Data)

数据是数据库系统研究和处理的对象,是保存在存储介质上能够被计算机识别的符号。在实际应用中,有两种基本形式的数据:一种是可以参与数值运算的数值型数据,如图书的价格;另一种是由字母、文字和其他特殊字符组成的文字数据,它是使用最多和最基本的数据,如图书名称。广义上讲,数据并不仅仅是数字或者文字,还可以是图形、图像、动画、影像和声音等多媒体形式,这些形式的数据都是将原始的内容经过数字化后存入计算机的,能够反映或描述事物的特征。

2. 信息 (Information)

信息是人脑对现实世界中的客观事物以及事物之间联系的抽象反映。它是一种被加工成特定形式的数据,通过对原始数据的提炼和加工给人们以有用的知识。

数据与信息既有区别,又有联系。一方面数据是信息的符号表示,但并非任何数据都能表示信

息,信息是数据的内涵,是对数据的定义的解释。另一方面信息不随表示它的数据形式的变化而变化,它是反映客观现实世界的知识,而数据则具有任意性,用不同的数据形式可以表示同样的信息。

1.1.2 数据处理

数据处理是指将数据转换成信息的过程。数据处理的根本目的是从大量的复杂数据中整理出对人们有价值、有意义的信息,作为行动和决策的依据。

1.1.3 数据管理技术的发展

计算机发展的初期,计算机主要用于科学计算,此时存在的数据管理问题都是采用人工的方式解决的,之后发展到文件系统,再后来才是数据库系统阶段。

1. 人工管理阶段

在计算机出现之前,人们运用纸张等手段从事记录、存储和数据加工,利用计算工具(算盘、计算尺)来进行计算,并主要依靠人的大脑来管理和使用这些数据。20世纪50年代以前,计算机仍主要用于数值计算。从当时的硬件发展水平看,外存只有纸带、卡片、磁带,没有磁盘等直接存取的存储设备,存储量非常小;软件方面,没有操作系统,没有高级语言和管理数据的软件;从数据看,数据量小,数据无结构,由用户统一管理,数据的处理方式是批处理,且数据间缺乏逻辑组织,数据依赖于特定的应用程序,缺乏独立性。在此阶段,存在着数据不保存、数据不具有独立性、数据不共享和由应用程序管理数据等缺点。

以一个图书馆的图书管理为例,在人工管理阶段,应用程序与数据之间的关系如图1.1所示。

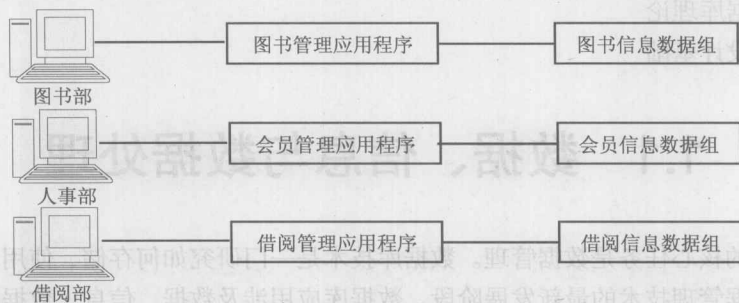


图 1.1 人工管理阶段

2. 文件管理阶段

20世纪50年代后期至20世纪60年代后期,计算机不仅被用于科学计算,而且开始大量用于数据管理。从硬件方面看,外存储器有了磁盘等可以进行直接存取的存储设备,因此数据可以重复使用。数据不再单独属于某个特定的程序,而可以由多个程序反复使用。从软件方面看,操作系统中已有了专门的数据管理软件,称为文件系统。对数据自身而言,数据的物理结构和逻辑结构有了区别,程序不必关心数据的物理位置而直接通过文件名和数据打交道,文件系统负责对数据的读/写。从处理方式上看,文件系统为程序和数据提供了一个公共接口,使应用程序采用统一的存取方法来存取、操作数据,程序和数据之间不再直接对应,因而有了一定的独立性。此外对数据能够联机实时处理,即在需要数据的时候随时从存储设备中查询、修改或更新。

文件管理的出现使计算机在数据管理方面弥补了人工管理的缺点,然而当数据量增加、使用数据的用户越来越多时,文件管理便不能适应更有效地使用数据的需要了,其缺点主要表现在:

(1) 数据的共享性差,冗余度大。当不同的应用程序所需的数据有部分相同时,仍需建立各自

的独立数据文件，而不能共享相同的数据。因此，数据冗余大，空间浪费严重。并且相同的数据重复存放，各自管理，当相同部分的数据需要修改时比较麻烦，稍有不慎，就造成数据的不一致。

(2) 数据和程序缺乏足够的独立性。文件中的数据是面向特定的应用的，文件之间是孤立的，不能反映现实世界事物之间的内在联系。

在文件管理阶段，图书管理中应用程序与数据文件之间的关系如图 1.2 所示。

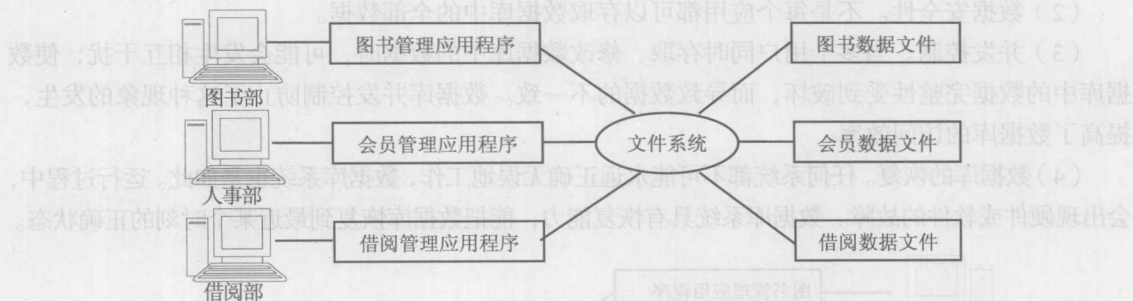


图 1.2 文件管理阶段

随着数据量的增长，文件管理系统已经不能满足人们的需要。美国在 20 世纪 60 年代进行阿波罗计划的研究时，阿波罗飞船由 200 万个零部件组成，其零部件的制造分散在世界各地。为了掌握计划进度及协调工程进展，阿波罗计划的主要合约者洛克威尔（Rockwell）公司曾研制了一个计算机零件管理系统。系统共用了 18 盘磁带，虽然可以工作，但效率极低，维护困难。18 盘磁带中 60% 是冗余数据，这个系统一度成为实现阿波罗计划的严重障碍。应用的需要推动了技术的发展，文件管理系统面对大量数据时的困境促使人们去研究新的数据管理技术，数据库技术应运而生了。

3. 数据库管理阶段

20 世纪 60 年代后期，数据管理进入数据库系统阶段。这一时期计算机管理的应用规模日益庞大，数据量急剧增长，数据要求共享的呼声越来越强。这种共享的含义是多种应用、多种语言互相覆盖地共享数据集合。此时的计算机有了大容量的磁盘，强大的计算能力。随着硬件价格下降，软件价格上升，编制和维护软件的相对成本增加，联机实时处理的要求增多，并行处理能力也被提上了日程。在这样的背景下，数据管理技术进入数据库系统阶段。

1968 年，IBM 公司研制的 IMS（Information Management System）层次数据库标志着数据管理技术进入了数据库阶段。1969 年，美国数据系统语言协会（Conference On Data System Language）公布了数据库工作组报告，对研制开发网状数据库起了巨大推动作用。1970 年，IBM 公司的研究员 E. F. Codd 连续发表的论文，奠定了关系数据库的基础。

与文件管理相比，数据库技术有了很大的改进，主要表现在：

(1) 数据库能够根据不同的需要按不同的方法组织数据，以最大限度地提高用户或应用程序访问数据的效率。

(2) 数据库中的数据是结构化的。在文件系统中，不同文件中的记录之间没有联系，联系仅存在于数据项之间。数据库系统不仅考虑数据项之间的联系，还要考虑记录之间的联系，保证了数据修改的一致性。

(3) 数据库中的数据是面向系统的，对于任何一个系统来说，数据库中的数据结构是透明的。任何应用程序都可以通过标准化接口访问数据库，如图 1.3 所示。通过这个统一的接口，用户可以用数据库系统提供的查询语言和交互式命令操纵数据库。用户也可以用高级语言编写程序来访问数据库，扩展了数据库的应用范围。

(4) 数据库系统比文件系统有更高的数据独立性。数据的组织和存储方法与应用程序相互独立,互不依赖,从而大大降低了应用程序的开发和维护代价。

不仅如此,数据库技术的发展使数据管理上了一个新台阶,数据库管理系统在数据完整性、安全性、并发访问和数据恢复方面都提供了非常完善的功能支持。

(1) 数据完整性。保证数据库存储数据的正确性。

(2) 数据安全性。不是每个应用都可以存取数据库中的全部数据。

(3) 并发控制。当多个用户同时存取、修改数据库中的数据时,可能会发生相互干扰,使数据库中的数据完整性受到破坏,而导致数据的不一致。数据库并发控制防止了这种现象的发生,提高了数据库的访问效率。

(4) 数据库的恢复。任何系统都不可能永远正确无误地工作,数据库系统也是如此。运行过程中,会出现硬件或软件的故障。数据库系统具有恢复能力,能把数据库恢复到最近某个时刻的正确状态。

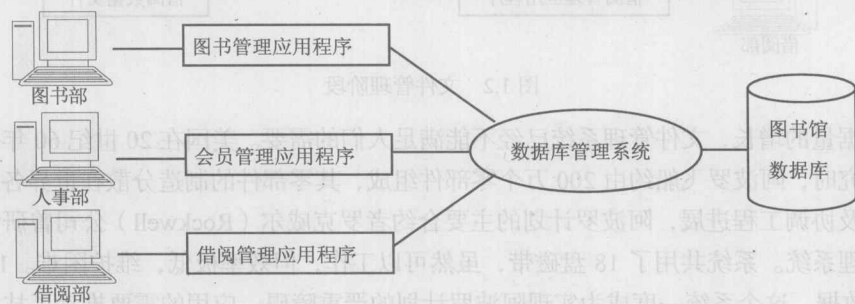


图 1.3 数据库系统阶段

1.2 数据模型

由于计算机不可能直接处理现实世界中的具体事物,所以人们需要将事物以数据的形式存储到计算机中。整个过程经历了对现实生活中事物特征的认识、概念化到计算机数据库里的具体表示的逐级抽象。这一过程划分成 3 个主要阶段,即现实世界阶段、信息世界阶段和机器世界阶段。现实世界中的数据经过人们的认识和抽象形成信息世界。在信息世界中用概念模型来描述数据及其联系,概念模型按用户的观点对数据和信息进行建模,不依赖于具体的机器,独立于具体的数据库管理系统,是对现实世界的第一层抽象。

1.2.1 数据描述

1. 实体 (Entity)

在现实世界阶段,实体是客观存在并可以相互区分的事物。实体不仅可以是实际存在的东西,还可以指抽象的事件。实体可以指人,如学生、图书馆员等。实体可以指物,如图书、书架等,实体可以指抽象的事件,如比赛、项目、演出等。实体还可以指事物与事物之间的联系,如借阅图书、学生选课等。

(1) 属性

每个实体都具有一定的特征(性质),即描述实体的特性称为属性。如图书实体可以有书名、作者、出版社、出版日期、价格等属性。读者实体有借书证编号、姓名、性别、工作单位等属性。每个属性有一个取值范围,该取值范围称为属性的值域。如读者实体中的性别属性,其取值范围

是“男”或“女”。

(2) 实体型、实体值和实体集

用实体名及其属性名集合来描述实体,称为实体型。具有相同属性的实体必然具有相同的特征和性质。例如,图书实体型是:

图书(图书编号,书名,作者,出版社,出版日期,价格)

图书“何谓文化”的实体值是:

(L00004,何谓文化,余秋雨,长江文艺出版社,2012/10/1,38.00)

同型实体的集合构成了实体集。例如:图书馆中的全部藏书构成了图书实体集。

(3) 属性型和属性值

属性型就是属性名及其取值类型,属性值就是属性在其值域中所取的具体值。如图书实体中的书名属性,“书名”及其使用的字符类型是属性型,而“何谓文化”是属性值。

在 Access 中,用“表”来存放同一类实体,即实体集。例如,读者表、借阅表、图书表等。Access 的一个“表”包含若干个字段,“表”中的字段就是实体的属性。字段值的集合组成表中的一条记录,代表一个具体的实体,即每一条记录表示一个实体。

2. 实体之间的联系

实体之间的对应关系称为联系(Relationship),这些联系在信息世界中反映为实体内部的联系和实体之间的联系。实体内部的联系通常指组成实体的各属性之间的联系;实体之间的联系通常指不同实体集之间的联系。这些联系归纳起来有三种类型:

(1) 一对一联系(1:1)。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,在实体集 B 中至多只有一个实体与之对应,反之亦然,则称实体集 A 与实体集 B 具有一对一联系。

例如,“图书馆”是一种实体,“馆长”也是一种实体。按照语义,一个图书馆只能有一个馆长,而一个馆长只能管理一个图书馆,则“图书馆”和“馆长”实体之间的联系就是一对一的联系。这种联系可以用图 1.4(a)来表示。

在 Access 中,一对一联系表现为主表中的每一条记录只与相关表中的一条记录相关联。

(2) 一对多联系(1:n)。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,在实体集 B 中可以有多个实体与之对应;而对于实体集 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 与实体集 B 之间存在一对多的联系。

例如,“图书馆”是一种实体,“馆员”也是一种实体。按照语义,一个图书馆可以有多个馆员,而一个馆员只能工作在一个图书馆,则“图书馆”和“馆员”实体之间的联系就是一对多的联系。这种联系可以用图 1.4(b)来表示。

在 Access 中,一对多联系表现为主表中的每条记录与相关表中的多条记录相关联。

(3) 多对多联系(m:n)。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,在实体集 B 中可以有多个实体与之对应;而对于实体集 B 中的任意一个实体,在实体集 A 中也可以有多个实体与之对应,则称实体集 A 与实体集 B 之间存在多对多的联系。

例如,“图书”是一种实体,“读者”也是一种实体。按照语义,一本图书馆可以由多位读者阅读,而一位读者可以阅读多本图书,则“图书”和“读者”实体之间的联系就是多对多的联系。这种联系可以用图 1.4(c)来表示。

在 Access 中,多对多的联系表现为一个表中的多条记录在相关表中同样可以有多条记录与之对应。

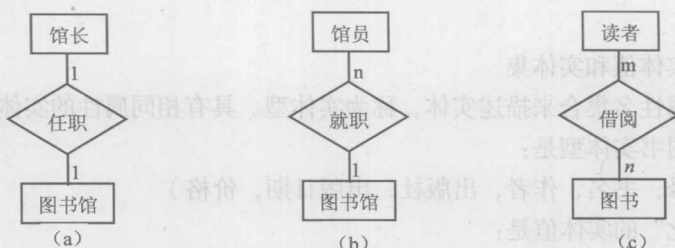


图 1.4 实体间的联系

1.2.2 概念模型

概念模型是对信息世界的建模。因此，概念模型应该能够方便、准确地表示出信息世界中的常用概念。概念模型有多种表示方法，其中，最常用的是实体-联系模型，简称 E-R 模型。

E-R 模型用矩形表示现实世界中的实体，用椭圆形表示实体的属性，用菱形表示实体间的联系，实体名、属性名和联系名分别写在相应框内，并用线段将各框连接起来。图 1.5 是图书借阅系统中的 E-R 图，该图建立了读者和图书两个不同的实体及其联系的模式。

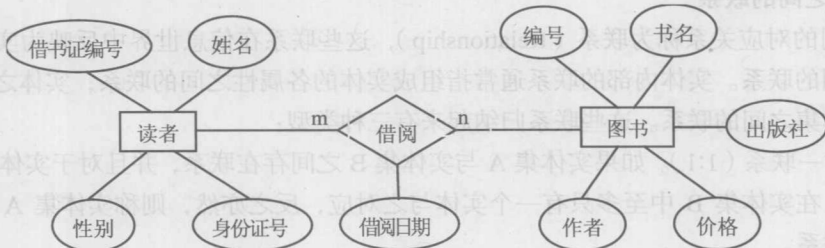


图 1.5 图书借阅系统中的 E-R 模型

1.2.3 数据模型

将信息输入到计算机后，数据就进入到计算机世界。前面讲到的概念模型是独立于计算机的，需要转换成具体的数据库管理系统所能识别的数据模型，才能将数据和数据之间的联系保存到计算机上。我们把表示事物以及事物之间联系的模型称为数据模型。任何一个数据库管理系统都是基于某种数据模型的。数据库领域常见的数据模型有以下四种。

1. 层次模型 (Hierarchical Model)

层次模型是数据库系统中最早出现的数据模型，它用树型结构表示实体及其之间的联系。IBM 公司的 IMS (Information Management System) 系统是典型的层次数据模型，且它是世界上最早出现的大型数据库系统。在这种模型中，需要满足如下两个条件：

(1) 有且仅有一个结点没有父结点，这个结点即是根结点。

(2) 其他结点有且仅有一个父结点。

层次模型具有层次清晰、构造简单、易于实现等特点，在现实生活中，很多实体间的联系本身就是层次关系，如一个学校的组织机构、一个家庭的世代关系等。但由于该模型仅能比较方便地表示出一对一和一对多的实体联系，而不能直接表示出多对多的实体联系，因而，对于复杂的数据关系，实现起来较为麻烦，这就是层次模型的局限性。

2. 网状模型 (Network Model)

网状模型是 20 世纪 70 年代产生的，其典型代表是 DBTG 系统，亦称 CODASYL 系统。网状

模型是数据系统语言研究会 CODASYL (Conference On Data System Language) 下属的数据库任务组 (Data Base Task Group, 简称 DBTG) 提出的一个系统方案。其特点为:

(1) 可以有一个以上的结点无父结点。

(2) 至少有一个结点有多于一个的父结点。

网状模型用以实体型为结点的有向图来表示实体及其之间的联系, 它克服了层次模型的缺点, 能够直接表示实体间多对多的联系。然而, 由于技术上的困难, 一些已实现的网状数据库管理系统中仍然只允许处理一对多的联系。

3. 关系模型 (Relational Model)

1970 年美国 IBM 公司 San Jose 研究室的研究员 E.F.Codd 首次提出了数据库系统的关系模型, 开创了数据库的关系方法和关系数据理论的研究, 为数据库技术奠定了理论基础。E.F.Codd 的杰出工作, 使他于 1981 年获得 ACM 图灵奖。

关系模式是目前使用最广泛的数据模型, 它与层次模型和网状模型有着本质的差别。关系模式建立在严格的数学理论基础上, 它用二维表格来表示实体及其相互之间的联系。在关系模型中, 把实体集看成一个二维表, 每一个二维表称为一个关系。每个关系均有一个名字, 称为关系名。20 世纪 80 年代以来出现的数据库管理系统几乎都支持关系模型。本书讲授的 Access 2010 就是一种关系数据库管理系统。

4. 面向对象的模型 (Object Oriented Model)

面向对象模型是近几年发展起来的一种新兴的数据模型。该模型是在吸收了以前的各种数据模型优点的基础上, 借鉴了面向对象程序设计方法而建立的一种模型。面向对象的方法为数据模型的建立提供了分类、概括、联合和聚集等四种数据处理技术, 这些技术对复杂空间数据的表达较为理想。

在面向对象模型中, 面向对象的核心概念构成了面向对象数据模型的基础, 其概念包括:

(1) 对象 (Object) 与对象标识 (OID)。现实世界中的任何实体都可以统一地用对象来表示。每一个对象都有它唯一的标识, 称为对象标识, 对象标识始终保持不变。

(2) 类 (Class)。所有具有相同属性和操作集的对象构成一个对象类 (简称类)。任何一个对象都是某一对象类的一个实例 (Instance)。

(3) 事件 (Event)。客观世界是由对象构成的, 客观世界中的所有行动都是由对象发出且能够为某些对象感受到, 把这样的行动称为事件。在关系数据库应用系统中, 事件分为内部事件和外部事件。

1.3 数据库系统

1.3.1 数据库及其特点

1. 数据库 (DataBase, DB)

数据库是指存储在计算机存储设备上、结构化的相关数据的集合。数据库中不仅包含描述事物的数据本身, 而且还包括相关事物之间的联系。数据库中的数据面向多种应用, 可以被多个用户、多个应用程序共享。其数据结构与使用数据的程序之间相互对立, 对于数据的增加、删除、修改和检索由数据库管理系统进行统一管理和控制, 用户对数据库进行的各种操作都是由数据库管理系统实现的。

2. 数据库的特点

(1) 数据结构化

数据库系统实现了整体数据的结构化, 这是数据库的最主要的特征之一。这里所说的“整体”

结构化,是指在数据库中的数据不再仅针对某个应用,而是面向全组织;不仅数据内部是结构化,而且整体是结构化,数据之间有联系。

(2) 数据的共享性高,冗余度低,易扩充

因为数据是面向整体的,所以数据可以被多个用户、多个应用程序共享使用,可以大大减少数据冗余,节约存储空间,避免数据之间的不相容性与不一致性。

(3) 数据独立性高

数据独立性包括数据的物理独立性和逻辑独立性。

物理独立性是指数据在磁盘上的数据库中如何存储是由数据库管理系统管理的、应用程序不需要了解,应用程序要处理的只是数据的逻辑结构,当数据的物理存储结构改变时,应用程序不用改变。

逻辑独立性是指应用程序与数据库的逻辑结构是相互独立的,也就是说,数据的逻辑结构改变了,应用程序也可以不改变。

数据与程序的独立,把数据的定义从程序中分离出去,加上存取数据的由 DBMS 负责提供,从而简化了应用程序的编写,大大减少了应用程序的维护和修改开销。

(4) 数据统一管理和控制

数据库的共享是并发的 (Concurrency) 共享,即多个用户可以同时存取数据库中的数据,甚至可以同时存取数据库中的同一个数据。数据库管理系统提供了数据的安全性保护 (Security)、数据的完整性检查 (Integrity)、数据库的并发访问控制 (Concurrency) 和数据库的故障恢复 (Recovery) 等数据控制功能。

1.3.2 数据库管理系统

1. 数据库管理系统 (DataBase Management System, DBMS)

数据库管理系统是数据库系统的核心。它是介于应用程序和操作系统之间,用于帮助管理输入到计算机中的大量数据的应用软件。它用于建立、使用和维护数据库,并通过对数据库进行统一的管理和控制,保证数据库的安全性和完整性。

2. 数据库管理系统的主要功能

(1) 数据定义

DBMS 提供数据定义,包括定义数据库的外模式、模式和内模式三级模式结构;定义三级模式间的两级映像;定义完整性约束和保密限制等约束。

(2) 数据操作

DBMS 提供的数据库操作,包括供用户实现的对数据的追加、删除、更新、查询等操作。

(3) 数据库的运行管理

DBMS 提供运行控制、管理功能。包括多用户环境下的并发控制、安全性检查和存取限制控制、完整性检查和执行、运行日志的组织管理、事务的管理和自动恢复,保证事务的原子性等功能。这些功能保证了数据库系统的正常运行。

(4) 数据组织、存储与管理

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,需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,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,选择合适的存取方法可以提高存取效率。

(5) 数据库的保护

DBMS 对数据库的保护通过 4 个方面来实现:数据库的恢复、数据库的并发控制、数据库的

完整性控制、数据库安全性控制。DBMS 的其他保护功能还有系统缓冲区管理以及数据存储的某些自适应调节机制等。

(6) 数据库的维护

DBMS 对数据的维护, 包括数据库的数据载入、转换、转储、数据库的重组和重构以及性能监控等功能, 这些功能分别由各个子模块来完成。

(7) 数据通信

DBMS 具有与操作系统的联机处理、分时系统及远程作业输入的相关接口, 负责处理数据的传送。对网络环境下的数据库系统, 还包括了 DBMS 与网络中其他软件系统的通信功能以及数据库之间的互操作功能。

1.3.3 数据库系统

1. 数据库系统 (DataBase System)

数据库系统是指引进数据库技术后的计算机系统, 是实现有组织地、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。

从数据库、数据库管理系统和数据库系统 3 个不同的概念中, 可以看出, 数据库强调的是数据, 数据库管理系统是应用软件, 而数据库系统强调的是整体。由此我们可以看出数据库管理系统在计算机系统中的地位, 如图 1.6 所示。

2. 数据库系统的组成

数据库系统由 4 部分组成: 硬件系统、数据库、数据库管理系统及相关软件、数据库系统有关人员。

计算机硬件是数据库系统的物理基础, 其主要功能是输入并存储程序和数据, 以及执行程序把数据加工成可以利用的形式。计算机硬件主要包括主机、存储设备、输入输出设备以及计算机网络环境。

数据库管理系统中的相关软件包括操作系统、数据库管理系统及数据库应用系统等。

数据库系统的相关人员包括数据库管理员 (DataBase Administrator, DBA)、数据库应用系统开发人员和最终用户。数据库管理员负责数据库的总体信息控制。DBA 的具体职责包括: 设计具体数据库中的信息内容和结构, 决定数据库的存储结构和存取策略, 定义数据库的安全性要求和完整性约束条件, 监控数据库的使用和运行, 负责数据库的性能改进、数据库的重组和重构, 以提高系统的性能。数据库应用系统开发人员包括系统分析员、系统设计员和程序员。系统分析员负责应用系统的需求分析和规范说明, 与用户和 DBA 相配合; 系统设计员负责应用系统的设计和数据库的设计; 程序员根据设计要求进行编码。最终用户是指利用系统的接口或查询语言访问数据库的人员, 一般对数据库知识了解较少。



图 1.6 数据库系统层次示意图

1.4 关系数据库

尽管数据库领域中存在多种组织数据的方式, 但关系数据库是效率最高的一种数据库系统。关系数据库管理系统 (Relation DataBase Management System, BMS) 采用关系模型作为数据的组