

全国中等职业技术学校计算机教材
Quanguo zhongdeng zhiye jishu xuexiao jisuanji jiaocai

数据库 及程序设计

Shujuku ji chengxu sheji

Shujuku ji chengxu sheji

(第二版)



中国劳动社会保障出版社

zhongguo laodong shehui baozhang chubanshe

全国中等职业技术学校计算机教材

数据库及程序设计

(第二版)

劳动和社会保障部教材办公室组织编写

中国劳动社会保障出版社

图书在版编目(CIP)数据

数据库及程序设计/宋明娥主编. —2 版. —北京: 中国劳动社会保障出版社, 2007
全国中等职业技术学校计算机教材

ISBN 978 - 7 - 5045 - 6190 - 9

I. 数… II. 宋… III. 数据库系统-程序设计 IV. TP311.13

中国版本图书馆 CIP 数据核字(2007)第 096988 号

中国劳动社会保障出版社出版发行

(北京市惠新东街 1 号 邮政编码: 100029)

出 版 人: 张梦欣

*

北京宏伟双华印刷有限公司印刷装订 新华书店经销
787 毫米×1092 毫米 16 开本 11.5 印张 259 千字

2007 年 7 月第 2 版 2007 年 7 月第 1 次印刷

定价: 19.00 元

读者服务部电话: 010 - 64929211

发行部电话: 010 - 64927085

出版社网址: <http://www.class.com.cn>

版权专有 侵权必究

举报电话: 010 - 64954652

前 言

为了更好地适应全国中等职业技术学校计算机专业的教学要求，我们根据劳动和社会保障部培训就业司颁发的《计算机专业教学计划与教学大纲》，修订和新开发了一批计算机专业教材。

这次教材修订工作的重点主要有以下几个方面。

第一，坚持以能力为本位，重视实践能力的培养，突出职业技术教育特色。根据计算机专业毕业生所从事职业的实际需要，合理确定学生应具备的能力结构与知识结构，对教材内容的深度、难度做了较大程度的调整。同时，进一步加强实践性教学内容，以满足社会对技能型人才的需求。

第二，根据信息技术行业发展，合理更新教材内容，尽可能多地在教材中充实新技术、新思想、新方法，力求使教材紧跟计算机科学技术的发展。同时，在教材编写过程中，严格贯彻国家有关技术标准的要求。

第三，努力贯彻国家关于职业资格证书与学历证书并重、职业资格证书制度与国家就业制度相衔接的政策精神，力求使教材内容涵盖有关国家职业标准（中级）和国家计算机等级考试的知识和技能要求。

第四，在教材编写模式方面，主要以案例教学为主，将编程思想、操作技巧、理论知识融入到案例的分析和处理过程。尽可能使用各种图示将各个知识点生动地展示出来，力求给学生营造一个更加直观的认知环境。

这次修订和新开发的教材包括：《Internet 基础与应用（第二版）》《常用办公软件（第二版）》《多媒体计算机组成与维修（第二版）》《数据库及程序设计（第二版）》《中文 FoxPro 及其程序设计（第二版）》《C/C++ 教程》《使用 PhotoShop CS2 处理图像》《Dreamweaver MX 网页设计与制作》《Flash 动画设计与制作》《3ds Max 动画设计与制作》《使用 CorelDRAW12 绘制图形》《使用 AutoCAD 2005 绘制图形》。

《数据库及程序设计（第二版）》的主要内容有：数据库基础、Visual FoxPro 系统初探、数据与数据运算、数据库建立和基本操作、查询与统计、程序设计基础、表单与控件、报表设计、项目与程序设计等。

前 言

本书由宋明娥、冯雪莲、王丽荣、王胜海、李万华、陶敏、徐波、郭晶、吴迪编写，宋明娥任主编，冯雪莲、王丽荣任副主编。

劳动和社会保障部教材办公室

2007年6月

目 录

第 1 章 数据库基础.....	(1)
1.1 数据库基本概念	(1)
1.1.1 数据处理	(1)
1.1.2 数据库系统	(3)
1.1.3 数据模型	(4)
思考与练习	(8)
第 2 章 Visual FoxPro 系统初探	(9)
2.1.1 Visual FoxPro 的环境	(9)
2.1.2 Visual FoxPro 工作方式	(11)
2.2 Visual FoxPro 可视化设计工具	(12)
2.2.1 Visual FoxPro 向导	(12)
2.2.2 Visual FoxPro 设计器	(13)
2.2.3 Visual FoxPro 生成器	(14)
思考与练习	(15)
第 3 章 数据与数据运算	(16)
3.1 数据类型	(16)
3.2 常量与变量	(18)
3.2.1 常量	(18)
3.2.2 变量	(19)
3.3 表达式	(21)
3.3.1 数值型表达式	(21)
3.3.2 字符型表达式	(22)
3.3.3 日期型表达式	(23)
3.3.4 关系型表达式	(23)
3.3.5 逻辑型表达式	(25)
3.3.6 表达式的优先级	(25)
3.4 数组	(26)
3.4.1 数组的定义	(26)
3.4.2 数组的赋值	(26)
3.5 函数简介	(28)

思考与练习·····	(32)
上机练习·····	(33)
第 4 章 数据库建立和基本操作 ·····	(36)
4.1 数据库的基本操作 ·····	(36)
4.1.1 数据库的建立 ·····	(36)
4.1.2 数据库的操作 ·····	(39)
4.2 表的建立 ·····	(43)
4.2.1 表的基本概念 ·····	(44)
4.2.2 表结构建立 ·····	(45)
4.2.3 表记录数据的输入 ·····	(48)
4.3 表的编辑与修改 ·····	(50)
4.3.1 表文件的打开与关闭 ·····	(50)
4.3.2 表结构的显示与修改 ·····	(50)
4.3.3 记录的显示与修改 ·····	(52)
4.3.4 记录的追加 ·····	(53)
4.3.5 记录的删除与恢复 ·····	(54)
4.4 设置表和字段的属性 ·····	(55)
4.4.1 设置字段的属性 ·····	(55)
4.4.2 设置表的属性 ·····	(56)
4.5 排序与索引 ·····	(57)
4.5.1 排序 ·····	(57)
4.5.2 索引 ·····	(58)
4.6 多表操作 ·····	(61)
4.6.1 多工作区的概念 ·····	(61)
4.6.2 表之间的关联 ·····	(62)
4.6.3 创造参照完整性 ·····	(65)
思考与练习·····	(66)
上机练习·····	(67)
第 5 章 查询与统计 ·····	(69)
5.1 查询 ·····	(69)
5.1.1 顺序查询 ·····	(69)
5.1.2 索引查询 ·····	(71)
5.2 设置查询选项 ·····	(72)
5.2.1 创建查询 ·····	(72)
5.2.2 运行查询 ·····	(76)
5.2.3 修改查询文件 ·····	(76)
5.2.4 设置查询的条件 ·····	(77)

5.2.5 设置排序依据	(78)
5.2.6 数据的分组查询	(79)
5.2.7 定向输出查询结果	(80)
5.3 查询多个表	(81)
5.4 视图	(83)
5.4.1 建立本地视图	(83)
5.4.2 运行本地视图	(85)
5.4.3 修改本地视图	(86)
5.5 统计命令	(86)
5.5.1 记录数统计命令	(86)
5.5.2 求和命令	(86)
5.5.3 求平均值命令	(87)
5.5.4 计算命令	(87)
5.5.5 分类汇总	(88)
思考与练习	(88)
上机练习	(90)
第 6 章 程序设计基础	(91)
6.1 程序设计概述	(91)
6.1.1 程序的概念	(91)
6.1.2 程序文件的建立与执行	(93)
6.1.3 程序中的常见命令	(94)
6.2 程序的控制结构	(96)
6.2.1 顺序结构	(96)
6.2.2 分支结构	(97)
6.2.3 循环结构	(100)
6.3 程序的模块化设计	(104)
6.3.1 过程的定义和调用	(105)
6.3.2 参数传递	(107)
6.3.3 变量的作用域	(108)
6.3.4 自定义函数	(108)
思考与练习	(109)
上机练习	(111)
第 7 章 表单与控件	(113)
7.1 面向对象的概念	(113)
7.2 创建与管理表单	(115)
7.2.1 表单的建立与运行	(115)
7.2.2 编辑表单	(121)

目 录

7.2.3 美化表单	(125)
7.2.4 创建多表表单	(129)
7.3 菜单设计	(132)
7.3.1 创建菜单	(132)
7.3.2 修改菜单	(135)
思考与练习	(136)
上机练习	(137)
第8章 报表设计	(139)
8.1 快速报表	(139)
8.1.1 使用报表向导创建报表	(139)
8.1.2 利用快速报表创建报表	(141)
8.2 使用报表设计器设计报表	(143)
8.2.1 报表设计区的带区	(143)
8.2.2 报表控件工具栏	(145)
8.2.3 报表的数据源和报表的布局	(146)
8.3 数据分组报表	(147)
8.3.1 设计数据分组报表	(147)
8.3.2 报表页面设置	(148)
8.4 打印报表	(149)
思考与练习	(149)
上机练习	(150)
第9章 项目与程序设计	(151)
9.1 项目管理器	(151)
9.1.1 创建项目	(151)
9.1.2 项目管理器窗口	(152)
9.2 项目管理器的操作	(157)
9.2.1 文件的创建及修改	(157)
9.2.2 文件的添加及删除	(159)
9.3 项目文件的连编与运行	(159)
9.4 实例应用	(161)
思考与练习	(172)
上机练习	(172)
附录1 Visual FoxPro 的属性说明	(173)
附录2 常用文件格式	(176)

第1章 数据库基础

本章要求：

- 理解数据与信息概念，了解数据与信息之间的关系。
- 了解数据管理经历的3个阶段以及各个阶段的特点。
- 掌握有关数据库的基本概念。
- 理解与实体相关的基本概念，了解实体间的联系。
- 理解数据模型的相关知识，重点掌握关系模型的基本知识。

随着社会信息化进程的加快，如何有效地进行信息管理成为人们面临的一大问题。为了合理地组织数据，高效地进行信息处理，人们引进了数据库技术。数据库技术诞生于20世纪60年代中期，是数据管理的最新技术，是计算机科学的重要分支，它的出现极大地促进了计算机应用向各行各业的渗透。

1.1

数据库基本概念

1.1.1 数据处理

数据(Data)是存储在某一媒体上能够识别的物理符号。其含义有两方面：一是描述事物特性的数据内容；二是存储在某一媒体上的数据形式。例如，王芳出生日期为1990年9月17日，身高1.56 m，学号020801。其中，王芳、1990年9月17日、1.56 m、020801都是数据，它们描述了王芳的基本特征。数据的形式可以多种多样，如王芳的出生日期又可以表示为“1990/09/17”，其含义并没有改变。

数据不仅包括数字、字母、文字及其他特殊字符组成的文本形式的数据库，而且还可包括图形、图像和声音等多媒体数据。但是，使用最多、最基本的仍然是文字数据。

数据处理是指将数据转换成信息的过程。从数据库处理的角度而言，信息是一种被加工成特定形式的数据库，这种数据形式对于数据接收者来说是有意义的。现实世界中的数据往往是非规范化的，但它是数据的原始集合，通过对这些原始数据库的处理，才能产生新的数据库(信

息)。这一处理包括对数据的收集、记录、分类、排序、存储、计算/加工、传输、制表和递交等操作。

随着社会的发展，数据管理也经历了由低级到高级的发展过程。随着计算机技术的发展，数据管理经历了以下 3 个阶段：

1. 人工管理阶段

在 20 世纪 50 年代中期以前，计算机主要用于科学计算。当时的硬件状况是：只有卡片、磁带，没有磁盘等可以直接存取的存储设备；软件状况是：没有操作系统，没有管理数据的软件，数据的处理方式是批处理。人工管理数据具有以下特点：

(1) 数据不保存

当时计算机主要用于科学计算，一般不需要长期保存数据，只有在用时才将数据输入，用完后数据同时撤走。

(2) 数据由应用程序自己管理

没有相应的软件系统负责数据管理工作，数据需要由应用程序自己管理，程序员负担很重。

(3) 数据不共享

一组数据只能对应一个应用程序，当多个应用程序同时涉及某些相同的数据时，无法相互利用。

(4) 数据不具有独立性

数据发生变化后，必须对应用程序作相应修改，这就进一步加重了程序员的负担。

2. 文件系统管理阶段

20 世纪 50 年代后期到 60 年代中期，计算机不仅用于科学计算，而且还大量用于管理。这时已有了磁盘、磁鼓等直接存储设备，在软件方面也已经有了专门的数据管理软件。

用文件系统管理数据具有以下特点：

(1) 程序与数据有了一定的独立性

程序和数据分开存储，有了程序文件和数据文件的区别。

(2) 数据可以长期保存

数据文件可以长期保存在外存储器上，从而可多次存取使用数据。

(3) 数据共享性差

文件系统中的数据文件是为了满足特定业务领域或某部门的专门需要而设计的，服务于某一特定的应用程序。数据和程序相互依赖，无法避免不同应用程序间相同数据的重复存储等操作。

3. 数据库系统管理阶段

自 20 世纪 60 年代后期以来，计算机技术发展越来越快，计算机用于管理的规模更为庞大，应用更为广泛。为解决多用户、多应用共享数据的需求，使数据尽可能多地为应用服务，就出现了数据库技术，出现了统一管理数据的专门软件系统——数据库管理系统。

用数据库管理数据具有数据结构化、数据共享、数据独立、冗余度可控、统一的数据控制功能等特点。

1.1.2 数据库系统

1. 有关数据库的基本概念

(1) 数据库

数据库 (DataBase, DB) 是存储在计算机存储设备上结构化的相关数据集合。它不仅包括描述事物的数据本身, 而且还包括相关事物之间的联系。

数据库中的数据往往不像文件系统那样, 只面向某一特定的应用, 而是面向多种应用, 可以被多个用户、多个应用程序共享。其数据结构独立于使用数据的程序, 对于数据的增加、删除、修改和查询由系统软件进行统一控制。

(2) 数据库应用系统

数据库应用系统是指系统开发人员利用数据库系统资源开发出来的, 面向某一类实际应用的应用软件系统。例如, 以数据库为基础的财务管理系统、人事管理系统、图书管理系统、教学管理系统等。

(3) 数据库管理系统

为了让多种应用程序并发地使用数据库中具有最小冗余度的共享数据, 必须使数据与程序具有较高的独立性。这就需要一个软件系统对数据实行专门管理, 提供安全性和完整性等统一控制机制, 方便用户以交互命令或程序方式对数据库进行操作。

数据库管理系统可以对数据库的建立、使用和维护进行管理。在数据库系统中, 各层次软件之间的相互关系如图 1—1 所示。

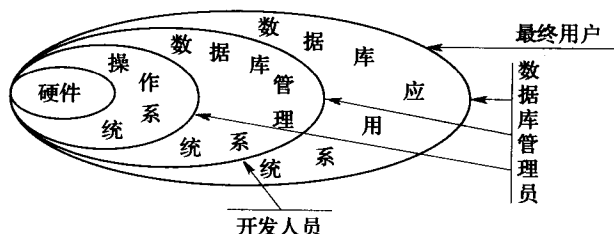


图 1—1 数据库系统

(4) 数据库系统

通常将引进数据库技术的计算机系统称为数据库系统 (DataBase System, DBS)。一般来说, 数据库系统由以下几部分组成。

1) 计算机硬件系统。用来运行操作系统、数据库管理系统、应用程序以及存储数据库的本地计算机系统和网络硬件环境。

2) 数据库集合。存储在本地计算机外存设备或网络存储设备上的若干个设计合理、满足应用需要的数据库。

3) 数据库管理系统。数据库管理系统 (DataBase Management System, DBMS) 是数据库系统的核心, 用于协助用户创建、维护和使用数据库的系统软件。

4) 相关软件。包括操作系统、编译系统、应用开发工具软件和计算机网络软件等。

5) 人员。包括数据库管理员和用户。数据库管理员负责数据库系统的建立、维护和管理。用户可分为专业用户和最终用户,专业用户侧重于设计数据库和开发应用程序,最终用户则侧重于数据库的使用。

2. 数据库系统特点

在数据库技术出现之前,计算机使用文件系统来存储、管理数据,与文件系统相比较,数据库系统具有以下特点:

(1) 数据结构化

在文件系统中,各个文件之间不存在相互联系。在数据库系统中则不同,在同一数据库中的数据文件是有联系的,且在整体上服从一定的结构形式。

(2) 数据共享

在文件系统中,数据一般是由特定的用户专用。而数据库中的数据不仅可为同一企业或结构之内的各个部门所共享,也可为不同单位、地域甚至不同国家的用户所共享。

(3) 数据独立

在文件系统中,数据结构和应用程序相互依赖、相互影响。在数据库系统中则力求摆脱这种依赖,实现数据的独立性。

(4) 冗余度可控

文件系统中数据专用,每个用户拥有和使用自己的数据,造成许多数据重复,这就是数据冗余。在数据库系统中实现共享后,将删除不必要的重复,但为了提高查询效率,有时也保留少量重复数据,其冗余度可由设计人员控制。

(5) 统一的数据控制功能

数据库可以被多个用户或应用程序共享,数据的存取往往是并发的,即多个用户同时使用同一个数据库。数据库管理系统必须提供必要的保护措施,包括并发访问控制功能、数据的安全性控制功能和数据的完整性控制功能。

1.1.3 数据模型

数据是人们对于客观事物进行抽象提取的结果,用于反映客观事物的属性。人们把客观存在的事物以数据的形式存储到计算机中,经历了对现实生活中事物特性的认识、概念化到计算机数据库的具体表示的逐级抽象过程。

1. 实体的描述

现实世界中存在着各种各样事物,事物与事物之间存在着联系。这种联系是客观存在的,是由事物本身的性质所决定的。例如,图书馆中有图书和读者,读者借阅图书;学校的教学系统中有教师、学生、课程,教师为学生授课,学生选修课程并取得成绩;在物资或商业部门有货物、客户,客户要订货、购物;在体育竞赛中有参赛代表队、竞赛项目,代表队中的运动员参加特定项目的比赛等。如果管理的对象较多或者比较特殊,事物之间的联系就可能较为复杂。

(1) 实体

现实世界中客观存在并且相互区别的事物称为实体。实体可以是具体的人、事、物,如一名教师,一个部门或一台计算机,也可以是抽象的概念或事件,如讲一堂课,给某单位供

应一台设备等。

(2) 实体的属性

描述实体的特性称为属性。属性是实体之间相互区别的标志，一个实体可以由若干个属性来刻画。例如，学生实体用学号、姓名、性别等若干个属性来描述；图书实体用总编号、分类号、书名等多个属性来描述。在 Visual FoxPro 中，属性又称为字段。

(3) 实体集和实体型

属性值的集合表示一个实体，同类型实体的集合称为实体集。例如，在学生实体集中，(020801, 徐立, 男, 1991/10/01) 表示学生名册中的一个具体的人；学校全体学生构成一个学校的学生实体集。

在 Visual FoxPro 中，用“表”来存放同一类实体，即实体集。每一个“表”中包含若干个字段，“表”中所包含的“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，见表 1—1。该表用来存放学生实体集，其中学号、姓名、性别、出生日期、专业是描述实体的字段；对应字段的值：20060501、宋英、女、1990/08/20、会计，组合为一条记录，代表一个具体的学生。

表 1—1 学生登记表

学号	姓名	性别	出生日期	专业
20060501	宋英	女	1990/08/20	会计
20060803	王玲	女	1989/06/01	会计
20060401	李刚	男	1987/05/30	航空
20060502	张帅	男	1989/04/03	计算机

2. 实体间联系

实体之间的对应关系称为联系，反映了现实世界各种事物之间的相互关系，如图 1—2 所示。两个实体之间的联系可以归结为以下 3 种类型。

(1) 一对一联系 (1:1)

如果对于实体集 A 中的每一个实体，实体集 B 中至多有一个（也可以没有）实体与之联系，反之亦然，则称实体 A 与实体 B 具有一对一联系，记为 1:1。

例如，一个班级只有一个正班长，而一个班长又只可以在一个班中任职，则班级与班长之间具有一对一联系，如图 1—2 所示。

在 Visual FoxPro 中，一对一的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)

如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体 ($n \geq 1$) 与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中至多只有一个实体与之联系，则称实体集 A 与

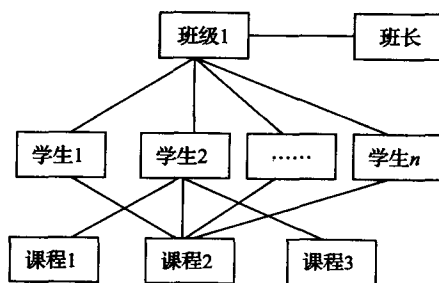


图 1—2 实体间联系

实体集 B 具有一对多联系，记为 $1:n$ 。

例如，一个班级中有若干名学生，而每个学生只能在一个班级中注册，则班级和学生之间具有一对多联系，如图 1—2 所示。

在 Visual FoxPro 中，一对多的联系表现为主表中的每一条记录与相关表中的多条记录相关联。即表 A 的一个记录可以与表 B 中的多个记录相对应，但表 B 中的一个记录最多只能与表 A 中的一个记录相对应。

(3) 多对多联系 ($m:n$)

如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体 ($n \geq 1$) 与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中也有 m 个实体 ($m \geq 1$) 与之联系，则称实体集 A 与实体集 B 具有多对多联系，记为 $m:n$ 。

例如，一门课程由多名同学选修，而一个学生又可以同时选修多门课程，则课程与学生之间具有多对多联系，如图 1—2 所示。

在 Visual FoxPro 中，多对多的联系表现为一个表中的多个记录在相关表中同样有多个记录与其相匹配。即表 A 的一条记录在表 B 中可以对应多条记录，而表 B 的一条记录在表 A 中也可以对应多条记录。

实际上，一对一联系是一对多联系的特例，而一对多联系又是多对多联系的特例。

3. 数据模型

为了反映事物本身及事物之间的各种联系，数据库中的数据必须有一定的结构，这种结构用数据模型来表示。模型是现实世界特征的模拟和抽象，而数据模型 (Data Model) 是现实世界数据特征的抽象。数据库不仅管理数据本身，而且要使用数据模型表示出数据之间的联系。可见，数据模型是数据库管理系统用来表示实体及实体间联系的方法。在数据库中用数据模型来抽象、表示和处理现实世界中的数据和信息。

任何一个数据库管理系统都是基于某种数据模型的。数据库管理系统所支持的数据模型分为以下 3 种：

(1) 层次模型

用树形结构来表示实体与实体间联系的模型称为层次模型，如图 1—3 所示。

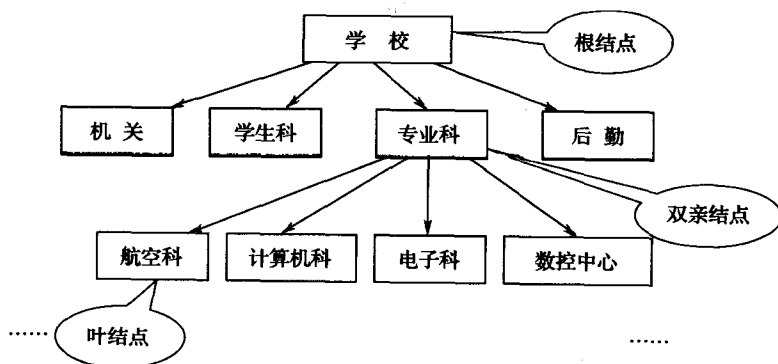


图 1—3 层次模型示例

在这种模型中,数据被组织成由“根”开始倒长的“树”,每个实体由根开始沿着不同的分支放在不同的层次上。

(2) 网状模型

用网形结构来表示实体与实体间联系的模型称为网状模型。网中的每一个结点代表一个实体类型。网状模型突破了层次模型的两点限制:允许一个以上的结点无双亲;一个结点可以有多个双亲结点。网状模型可以方便地表示各种类型的联系。如图 1—4 所示,其中用 C 表示课程, S 表示学生,用编号表示不同的课程和学生,有向边表示所选修的课程。

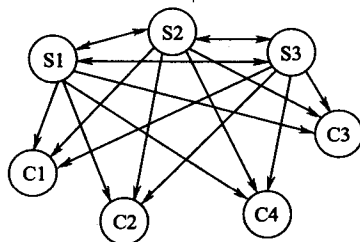


图 1—4 网状模型示例

网状模型中每个结点表示一个实体集,结点间的连线表示实体集之间一对多的父子联系。

(3) 关系模型

用二维表结构来表示实体以及实体之间联系的模型称为关系模型。在用户观点下,关系模型中数据的逻辑结构是一张二维表,它由行和列组成,见表 1—1。

以表 1—1 为例,介绍关系模型中的一些术语。

关系 (Relation): 一个关系通常对应一张二维表。例如,表 1—1 中的这张学生登记表就是一个关系,可以命名为“学生关系”。

元组 (Tuple): 表中的一行即为一个元组 (也称为记录)。例如,表 1—1 有 4 行,对应 4 个元组/记录。

属性 (Attribute): 表中的一列即为一个属性 (也称字段),给每一个属性起一个名称,即属性名 (字段名)。例如,表 1—1 中有 5 个属性/字段,属性名/字段名分别为学号、姓名、性别、出生日期、专业。

主码/关键字 (Key): 主码是表中的某个属性组,它可以唯一地确定一个元组。例如,表 1—1 中的学号可以唯一确定一个学生,也就可以作为本关系的主码。

域 (Domain): 属性的取值范围。例如,性别的域是 (男,女),专业的域是一个学校所有专业的名称集合。

分量: 元组中的一个属性值。例如,表 1—1 中的第一个元组在学号属性上的取值为 20060501,则 20060501 就是第一个元组的一个分量。

关系模式: 对关系的描述。一般表示为:

关系名 (属性 1, 属性 2, ..., 属性 n)。

例如,表 1—1 的学生关系可以描述为:

学生 (学号, 姓名, 性别, 出生日期, 专业)

在关系模型中,实体以及实体间的联系都是用关系来表示的。

思考与练习

- (1) 数据管理经历了哪几个阶段？请简述数据管理发展的最后阶段的特点。
- (2) 数据库系统由哪几部分组成？
- (3) 什么是数据？数据有什么特征？数据和信息有什么关系？
- (4) 什么是数据处理？
- (5) 什么是数据库？数据库中的数据有什么特点？
- (6) 什么是数据模型？包括哪几种？
- (7) 关系模型中的术语有哪些？各自的含义是什么？