

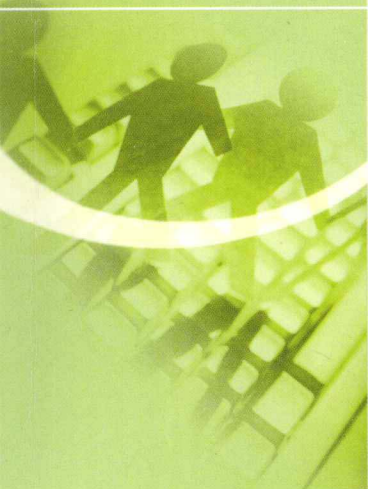
21 世纪高等院校计算机规划教材

<<< < <<<

Visual FoxPro6.0 程序设计教程

主 编 张久彪

副主编 尹荣章 李 武 何 鲲



电子科技大学出版社



21 世纪高等院校计算机规划教材

Visual FoxPro 6.0程序设计教程

主 编 张久彪
副主编 尹荣章 李 武 何 鲲
编 写 (按姓氏笔画排序)
尹荣章 叶明全 朱文婕
何 鲲 张久彪 张 钰
李 武 李六杏 姚 珺

电子科技大学出版社

图书在版编目(CIP)数据

Visual FoxPro 6.0 程序设计教程/张久彪主编. —成都:电子科技大学出版社, 2009. 8
ISBN 978-7-5647-0324-0

I. V… II. 张… III. 关系数据库-数据库管理系统, Visual FoxPro 6.0-程序设计-高等学校-教材 IV. TP311.138

中国版本图书馆 CIP 数据核字(2009)第 140768 号

内 容 简 介

本教材是“大学计算机基础”后续课程,按照全国计算机等级考试大纲和安徽省计算机水平考试大纲组织编写。在内容上以操作为主线介绍数据库管理系统 Visual FoxPro,力求叙述简练、精准。全书主要内容有:数据库管理系统基础知识、Visual FoxPro 基础知识;自由表、数据库表、数据库的基本操作;SQL 语言、查询、菜单、报表设计;结构化程序设计、面向对象程序设计;应用程序开发。

全书共分 9 章,每章均有习题、上机指导;4 个附录,全国计算机等级考试二级 VFP 大纲、全国计算机等级考试笔试和机试模拟题、安徽省计算机水平考试笔试和机试模拟题、各章习题参考答案。

本书适用于高等院校计算机专业和非计算机专业学生作教材使用,也可作为各类 Visual FoxPro 考试用书。

Visual FoxPro 6.0 程序设计教程

主 编 张久彪

出 版:电子科技大学出版社(成都市一环路东一段 159 号电子信息产业大厦 邮编:610051)

策划编辑:曾 艺

责任编辑:张蓉莉

主 页:www.uestcp.com.cn

电子邮箱:uestcp@uestcp.com.cn

发 行:新华书店经销

印 刷:安徽省天歌印刷厂

成品尺寸:185mm×260mm 印张 20.625 字数 500 千字

版 次:2009 年 8 月第 1 版

印 次:2009 年 8 月第 1 次印刷

书 号:ISBN 978-7-5647-0324-0

定 价:29.80 元

■ 版权所有 侵权必究 ■

◆ 发行部电话:028-83202463,邮购部电话:028-83208003。

◆ 本书如有缺页、破损、装订错误,请寄回印刷厂调换。

前 言

《Visual FoxPro 6.0 程序设计教程》是大学计算机基础的后续课程。本书是依据全国高等学校(安徽考区)计算机基础教育教学(考试)大纲和全国计算机等级考试大纲组织编写的。全书共9章:第1章,数据库基础知识;第2章,Visual FoxPro 6.0 基础;第3章,Visual FoxPro 6.0 数据表基本操作;第4章,Visual FoxPro 6.0 数据库;第5章,SQL 语言、查询及视图;第6章,结构化程序设计;第7章,表单设计;第8章,菜单、报表设计;第9章,应用程序。每章包括基本知识介绍、例题分析、上机实验指导和习题,以方便学生学习和上机实践。附录有全国计算机等级考试笔试和机试模拟题;安徽省计算机水平考试笔试和机试模拟题。这部分对参加考试的同学极具指导性。

本书在编写过程中,充分考虑教、学、考一体化的需要,既讲述必要的理论基础,也从应用的角度、结合具体的实践加以讲述,做到知识性、实用性、操作性的有机结合。本书深入浅出、通俗易懂、重点突出、图文并茂,书中例题均调试正确。本教材可作为普通高等学校(高职高专)计算机和非计算机专业数据库应用课程使用,也可供各类信息管理人员学习参考。建议学习者先模仿例题,再做实验,多上机实践,将取得事半功倍的效果。

本书由张久彪任主编,尹荣章、李武、何鲲任副主编。参加编写的还有叶明全、朱文婕、张钰、李六杏、姚璐等教师。

计算机技术的发展日新月异,加之时间仓促,编者水平有限,书中难免错误和不妥之处,敬请广大读者批评指正。

编 者
2009 年 6 月

目 录

第 1 章 数据库基础知识	1
1.1 数据管理技术的发展	2
1.2 数据库系统	4
1.3 数据模型	4
1.4 关系数据库	6
1.5 数据库设计基础	7
习题一	9
第 2 章 Visual FoxPro 6.0 基础	11
2.1 Visual FoxPro 系统概述	12
2.1.1 Visual FoxPro 的发展史	12
2.1.2 Visual FoxPro 6.0 的安装、启动与退出	13
2.1.3 Visual FoxPro 6.0 的用户界面	14
2.1.4 Visual FoxPro 6.0 的工作方式	16
2.1.5 Visual FoxPro 6.0 辅助设计工具	16
2.1.6 Visual FoxPro 6.0 文件类型	19
2.2 数据类型	19
2.3 常量与变量	21
2.3.1 常量	21
2.3.2 变量	22
2.3.3 内存变量常用命令	25
2.4 表达式	27
2.5 常用函数	30
2.5.1 数值函数	30
2.5.2 字符函数	32
2.5.3 日期时间函数	34
2.5.4 数据类型转换函数	35
2.5.5 测试函数	36
2.5.6 其他常用函数	38
实验 2-1 Visual FoxPro 6.0 中常量、变量和函数的使用	39

习题二	44
第3章 Visual FoxPro 6.0 数据表基本操作	47
3.1 表的设计	48
3.1.1 表结构的设计	48
3.1.2 Visual FoxPro 6.0 中的命令	49
3.2 表结构操作	50
3.2.1 表结构的建立	50
3.2.2 表结构的修改	55
3.3 表记录操作	55
3.3.1 表的打开与关闭	55
3.3.2 表记录的浏览	56
3.3.3 表记录的定位	57
3.3.4 表记录的修改	59
3.3.5 表记录的追加与插入	60
3.3.6 表记录的删除与恢复	62
3.3.7 表与表结构的复制	64
3.4 排序与索引	66
3.4.1 排序	66
3.4.2 索引	67
3.5 数据查询	75
3.5.1 顺序查询	75
3.5.2 索引查询	76
3.6 数据统计	77
3.6.1 统计记录个数命令	77
3.6.2 求和命令	78
3.6.3 求平均值命令	79
3.6.4 汇总命令	79
实验3-1 自由表的建立及其基本操作	80
实验3-2 索引的建立与使用	85
习题三	87
第4章 Visual FoxPro 6.0 数据库	90
4.1 数据库的设计与建立	91
4.1.1 数据库的建立	91
4.1.2 数据库的基本操作	92
4.1.3 数据库表的基本操作	94

4.2 多工作区操作	95
4.2.1 工作区	95
4.2.2 多表之间的关系	98
4.2.3 永久关系	98
4.2.4 临时关系	100
4.3 数据词典	102
4.3.1 字段属性的设置	102
4.3.2 设置记录规则	105
4.3.3 主索引	106
4.3.4 数据完整性	108
实验 4-1 数据库的建立与操作	110
习题四	113
第 5 章 SQL、查询与视图	115
5.1 SQL 概述	116
5.2 SQL 的查询功能	116
5.2.1 SELECT 查询命令	116
5.2.2 简单查询	117
5.2.3 嵌套查询	121
5.2.4 联接查询	124
5.2.5 分组和计算查询	126
5.2.6 集合的并操作查询	127
5.3 SQL 的数据定义和数据修改	128
5.3.1 SQL 的数据定义	128
5.3.2 SQL 的数据修改	129
5.4 查询	130
5.4.1 一个查询实例	130
5.4.2 查询设计器简介	133
5.4.3 查询的运行	134
5.4.4 查询的修改	134
5.4.5 查询的去向	134
5.4.6 创建一个基于多表的查询	135
5.5 视图	138
5.5.1 一个视图实例	138
5.5.2 视图设计器	139
5.5.3 利用视图更新基本表数据	140
5.5.4 视图的其他操作	141
实验 5-1 SQL 语句	142

实验 5-2 查询设计器	144
习题五	146
第 6 章 结构化程序设计	150
6.1 Visual FoxPro 程序文件	151
6.1.1 程序的基本概念	151
6.1.2 程序文件的编辑与执行	152
6.1.3 程序调试	153
6.1.4 程序中常用的输入输出语句	156
6.2 Visual FoxPro 程序结构	158
6.2.1 顺序结构	159
6.2.2 分支结构	159
6.2.3 循环结构	164
6.3 Visual FoxPro 模块化程序设计	174
6.3.1 模块的定义与调用	174
6.3.2 模块间的参数传递	177
6.3.3 模块中变量的作用域	178
实验 6-1 程序设计的三种基本结构	180
实验 6-2 模块化程序设计	183
习题六	185
第 7 章 表单设计	195
7.1 表单设计器	196
7.1.1 表单文件操作	196
7.1.2 表单设计器	197
7.1.3 表单属性	198
7.2 表单向导	199
7.3 表单中的常用控件	202
7.3.1 控件的种类	202
7.3.2 常用控件	203
7.4 面向对象程序设计	219
7.4.1 对象	219
7.4.2 类	223
7.5 典型例题	225
实验 7-1 表单设计	231
习题七	235

第 8 章 菜单、报表设计	238
8.1 菜单设计	239
8.1.1 菜单系统设计过程	239
8.1.2 菜单设计器	240
8.1.3 “显示”菜单	242
8.1.4 菜单设计实例	244
8.1.5 快速菜单与快捷菜单设计	247
8.1.6 在应用程序中使用菜单	247
8.2 报表设计	248
8.2.1 创建报表文件	248
8.2.2 在报表中使用控件	260
8.2.3 数据分组	263
实验 8-1 菜单的建立与使用	266
习题八	269
第 9 章 应用程序	272
9.1 项目管理器	273
9.1.1 创建项目	273
9.1.2 “项目管理器”	274
9.2 应用程序开发	275
9.2.1 应用程序开发基本步骤	275
9.2.2 项目信息	276
9.2.3 设置文件的“排除”与“包含”	276
9.2.4 设置主文件	277
9.2.5 连编项目	278
9.2.6 应用程序发布	278
9.3 应用程序开发实例——学生管理系统	280
实验 9-1 应用程序开发	287
习题九	288
附录 A 2009 年全国计算机等级考试二级 VFP 大纲	290
附录 B 全国计算机等级考试二级 VFP 笔试、机试模拟题	293
附录 C 全省计算机水平考试二级 VFP 笔试、机试模拟题	302
附录 D 习题参考答案	310
参考文献	319

第1章

数据库基础知识

本章要点

- 掌握数据、数据管理、数据库管理系统的基本概念
- 掌握数据模型的基本概念
- 掌握关系数据库、关系运算的基本概念
- 了解数据库设计步骤和过程

数据库技术是计算机领域的一个重要分支。在计算机应用的三大领域(科学计算、数据处理和过程控制)中,数据处理约占其中的70%,而数据库技术就是作为一门数据处理技术发展起来的。随着计算机应用的普及和深入,数据库技术变得越来越重要了,而了解、掌握数据库系统的基本概念和基本技术是应用数据库技术的前提。

1.1 数据管理技术的发展

1. 数据与数据处理

数据(Data)一般被认为是对客观事物特征所进行的一种抽象化、符号化的表示。数据的含义主要包括以下两个方面: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。

数据的概念在科学计算领域中已经大大拓宽了,数据不仅包括数字、字母、文字及其他特殊字符组成的数据,而且还包括图形、图像、动画、声音等多媒体数据,其中使用得最多的是文字数据。从信息处理角度看,任何事物的属性都是通过数据来表示的。用数据符号表示信息,其形式通常有3种:数值型数据,即对客观事物定量记录的符号,如体重、价格等;字符型数据,即对客观事物定性记录的符号,如姓名、住址等;特殊型数据,如声音、图像等。从计算机的角度来看,凡是能够被计算机接受并被计算机处理的符号都称为数据。

数据处理就是将数据转换成信息的过程,包括数据的收集、存储、加工、分类、排序、传输等一系列活动。通过处理数据可以获得信息,提取对人们有用的信息,作为决策的根据。可见,信息与数据是密切相关的,数据是信息的载体,信息是数据的内涵。例如,一个人的“出生日期”是不可以改变的,属于一个原始数据,而“年龄”是通过现年与出生日期相减而得到的二次数据。根据年龄、性别等有关此人的信息和离退休年龄的规定,就可以判断此人何时应该办理离退休手续。

2. 计算机数据管理

计算机对数据的管理包括对数据的收集、整理、组织、存储、维护、检索、统计、传输等一系列的工作。计算机对数据处理一般分为以下几个环节:

- 对原始数据的收集。
- 数据的规范化及编码。
- 数据输入。
- 数据处理。
- 数据输出。

计算机数据管理随着计算机硬件、软件技术和计算机应用范围的发展而不断发展,数据库技术的发展也使数据处理进入了一个崭新阶段。根据计算机所提供的数据独立性、数据共享性、数据完整性、数据存取方式等水平的高低,计算机数据管理主要经历了人工管理、文件系统、数据库系统、分布式数据库系统和面向对象数据库系统等几个阶段。

(1) 人工管理

在20世纪50年代中期以来,计算机主要用于科学计算,计算处理的数据量比较小,数据管理处于人工管理阶段。这一时期计算机数据管理的特点是:

数据管理没有统一的数据管理软件,也没有像磁盘这样的外存储设备,对数据处理没有一定的格式,主要靠应用来管理数据;数据是面向应用程序的,不具有独立性,一组数据

只能对应一个应用程序,数据不能共享;数据不能长期保存,程序运行结束后就退出计算机。一个程序中的数据不能被其他程序应用,所以程序与程序之间存在着大量的重复数据,称为数据冗余。

(2) 文件系统

在20世纪50年代后期至60年代中后期,数据管理进入了文件系统阶段。随着操作系统的产生和发展,程序设计人员可以利用操作系统提供的文件系统功能,将数据的内容、结构及作用等组成若干相互独立的数据文件,这一时期计算机数据管理的特点是:

在文件系统阶段,文件系统为程序和数据提供了一个公共接口,所以应用程序可以采用统一的存取方法来存取数据,程序和数据之间具有相对的独立性,也就是数据不再属于某个特定的应用程序,数据可以重复使用;程序和数据分开存储,有了程序文件和数据文件的区别;数据可以长期保存在外存储器上被多次存取。

虽然用文件系统管理数据已有了长足的进步,但是,同一数据项可能重复出现在多个文件中,导致数据冗余度大。这不仅浪费存储空间,增加更新开销,更严重的是,由于不能统一修改,容易造成数据的不一致性。例如,数据文件之间相互独立、缺少联系;数据冗余度大;数据无集中管理,安全性得不到保证等。

由于文件系统没有统一的模型来存储数据,仍有较高的数据冗余,而且文件系统存在的这些问题已经不能满足日益增长的信息需求,由此数据库系统就产生了。

(3) 数据库系统

在20世纪60年代后期,随着计算机系统性价比的持续提高,软件技术的不断发展,为实现计算机对数据的统一管理,达到数据共享的目的,数据库系统应运而生。数据库系统克服了文件系统的不足,将数据管理技术推向了数据库管理阶段,数据库管理系统运用数据库技术进行数据管理。这一时期计算机数据管理的特点是:

数据库技术使数据有了统一的结构,对所有的数据实行统一管理,提高了数据的共享性,使多个用户能够同时访问数据库中的数据,提高了数据管理效率;减小数据的冗余度,以提高数据的一致性和完整性;有较高的数据独立性,从而减小应用程序的开发和维护代价;为用户提供了方便的用户接口。

数据库也是以文件方式存储数据的,但它是数据的一种高级组织形式。在应用程序和数据库之间,由DBMS把所有应用程序中的相关数据汇集起来,按统一的数据模型存储在数据库中,各应用程序能方便、快捷的查询和调用。

(4) 分布式数据库系统

在20世纪70年代后期以前,数据库系统大多数是集中式的,网络技术的发展为数据库提供了分布式运行环境。分布式数据库系统是数据库技术和计算机网络技术紧密结合的产物。

数据库技术与网络技术的结合分为紧密结合和松散结合两大类,因此,分布式数据库系统分为物理上分布、逻辑上集中的分布式数据库结构和物理上分布、逻辑上分布的分布式数据库结构两种。

(5) 面向对象数据库系统

面向对象数据库系统是20世纪80年代引入计算机科学领域的一种新的设计程序的

数据库,它的发展非常快,影响计算机科学及其应用的各个领域。

面向对象数据库是数据库技术与面向对象程序设计相结合的产物。VFP 不但支持标准的过程化程序设计,还具有面向对象程序设计的强大功能和更大的灵活性。

1.2 数据库系统

1. 数据库 (DataBase)

数据库(DB)是以一定的组织方式存贮在一起的、能为多个用户所共享的、与应用程序彼此独立的相关数据的集合。由于数据模型分为层次模型、网状模型和关系模型三种基本类型,与之相对应,数据库也分为三种基本类型:层次型数据库、网状型数据库和关系型数据库。

2. 数据库管理系统 (DataBase Management System)

为了帮助用户建立、使用和管理数据库,就需要一套系统软件,称作数据库管理系统(DBMS),它是数据库与用户之间的接口。

3. 数据库系统 (DataBase System)

数据库系统简称 DBS,是指对一个具体的问题开发的应用系统,如图书管理系统、财务管理系统、学生成绩管理系统等。它包括和数据库有关的全部:数据库、数据库管理系统、应用程序以及数据库管理员和用户等等。

1.3 数据模型

1. 实体及属性

所谓实体是指客观存在的并可相互区别的任何事物。例如学校、工厂、人、设备等。

所谓属性是指实体代表的特定事物所具有的某方面的特征。例如把人作为一个实体,人的身高、体重、肤色就是属性;在教师档案中,用姓名、性别、职称、出生日期、基本工资、籍贯等属性来描述教师的特征,如表 1-1 所示。

表 1-1 教师情况表

姓 名	性 别	职 称	出生日期	基本工资	籍 贯
张红平	女	教授	58/09/24	1530.00	江苏苏州
李明友	男	讲师	70/05/15	810.00	安徽黄山

一张表只有表头的空表称实体型,表中的数据叫做实体值。属性型指表中栏目,属性型下所填的数据称为属性值。

在数据库中实体型相当于记录型,实体值相当于记录值,属性型相当于字段名,属性值相当于字段值。

2. 数据模型

数据模型是表示实体及实体之间联系的数据组织结构和形式,即数据在数据库中排列、组织所遵循的规则。在现实世界中,实体之间的联系可分为“一对一”、“一对多”、“多对多”的联系。

将实体之间的这种联系经过适当的转换,用一定的数据结构表示出来,就得到了数据模型,目前在数据库技术中应用的数据模型可分为三种基本模型。

(1) 层次模型

层次模型采用树型结构表示实体及联系。树中的每个结点代表实体,连线代表其联系。上下层的结点是一对多(1:n)的联系。层次模型的特点:①只有一个结点没有父结点,这个结点也称根结点。②其他结点有且仅有一个父结点,如图1-1所示。

(2) 网状模型

网状模型描述实体间多对多(n:m)的联系。其特点:①允许一个以上的结点无父结点。②每个结点可以有多个父结点,如图1-2所示。因此相比层次模型,网状模型更适于描述复杂的现实世界,但其存储结构和操作也较复杂。

图中用“○”表示数据实体,用“→”表示联系。

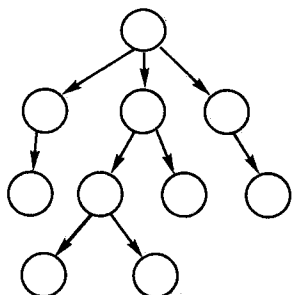


图 1-1 层次模型

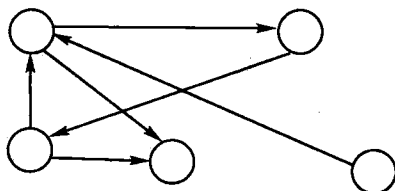


图 1-2 网状模型

(3) 关系模型

关系模型用二维表格来表示实体间的联系,数据特征为一对一的关系,如表1-2所示。关系模型以关系数学理论为基础。

表 1-2 学生情况关系

学 号	姓 名	性 别	成 绩
A11984001	李 磊	男	88
A11984002	刘 红	女	78
A11984003	王 丽	女	86
A11984004	张丽华	女	81

1.4 关系数据库

1. 关系模型

关系模型是一种表格数据模型,如表 1-2 所示,一般来说具有如下性质的一张二维表才能称为一个关系:

- (1) 每一列中的数据属于同一类型,例如,“姓名”这一列中的数据都为字符型。
- (2) 各列必须有不同的名字,例如,表 1-2 各列的名称为学号、姓名、性别、成绩,名称都不相同。
- (3) 表中行和列的顺序可以任意。
- (4) 表中各行相异,不允许有重复的行。
- (5) 表中的数据项是不可再分的最小数据项。

2. 关系运算

一张二维表是一个关系,二维表的行在关系中称为元组,在关系数据库中称为记录;二维表的列在关系中称为属性,在关系数据库中称为字段。

关系的基本运算有选择、投影和联接。

(1) 选择

选择运算是二维表的行运算,是指在关系中选择满足某些条件的元组抽出来构成新的关系。

例:表 1-3 是对表 1-2 源数据表进行选择运算结果,选择条件是查询 80 分以上的女同学记录。

表 1-3 选择运算

学 号	姓 名	性别	成绩
A11984003	王 丽	女	86
A11984004	张丽华	女	81

(2) 投影

投影运算是二维表的列运算,是在一个关系中选择某些属性,重新排列组成一个新关系。

例:表 1-4 是对表 1-3 源数据表进行投影运算结果,只保留“学号”和“成绩”两个属性。

表 1-4 投影运算

学号	成绩
A11984003	86
A11984004	81

(3) 联接

联接运算是将两个关系按照给定的条件以一切可能的组合方式联接起来,形成一个新的关系。

例:表1-7是对表1-5和表1-6的联接运算结果。即每个学生都选修两门课而生成学生选修课成绩表。

表 1-5 学生情况表

学号	姓名
A11984001	李磊
A11984002	刘红

表 1-6 选修课程表

课号	课程名
001	C 语言
002	图论

表 1-7 学生选修课成绩表

学号	课号	成绩
A11984001	001	
A11984002	001	
A11984001	002	
A11984002	002	

1.5 数据库设计基础

要设计出结构合理、访问信息方便、日后维护省时的数据库系统,就必须按照工程的方法去思考、设计、实施。本节简述数据库设计的步骤和过程。

1. 数据库设计步骤

数据库是一个包含多种数据对象的抽象的“容器”,可以简单地理解为存放数据的仓库。数据库除了包含“表”以外,还包含一些与数据库有关的内容,我们称这些表和其他内容为数据库对象。要创建一个使用方便的数据库,必须精心地组织数据,合理地设计数据库结构。利用 Visual FoxPro 6.0 设计数据库,可以按照以下的步骤来进行。

(1) 明确目的

首先进行需求分析,确定建立数据库的目的,这有助于确定数据库需要保存哪些信息。例如要建立一个教学管理系统(JXGL),就必须明确用户需要从数据库中获得何种信息,需要对哪些信息进行管理以及用户需要对数据进行什么操作等,这就需要设计人员和用户进行多次交流。

(2) 确定需要的表

在明确了建立数据库的目的之后,就可以着手把信息进行归类,形成既相互独立又相互联系的主题,例如“XSDA”(学生档案情况)、“XSCJ”(学生成绩情况)或“XSKC”(学生课程情况)等。每个主题中的信息都可以用数据库中的一个表来存放。

(3) 确定表的结构

根据在每个表中需要保存的信息,来确定表结构的字段名、字段宽度、类型、字数等。例如,在“XSDA”表中,可以有这样的字段:“学号”、“姓名”、“性别”、“年龄”等。

(4) 确定关系

分析数据库中每个表,确定一个表中的数据和其他表中数据的关系。必要时,可以在表中加入字段或创建一个新表来明确关系。例如“XSDA”表(含学号、姓名等字段)与“XSCJ”表(含学号、课程号、成绩三个字段),可以通过两表共有的“学号”字段,建立两表间的关系。

(5) 设计优化

对设计结果进行进一步分析,查找其中的错误、重复或遗漏数据。

2. 数据库设计过程

(1) 需求分析

用户需求主要包括以下几个方面:

- 信息需求。即用户要从数据库获得的信息内容。注意描述清楚系统中数据的类型。
- 处理需求。即需要对数据完成什么处理功能及处理的方式。注意操作执行、操作对数据的影响等。
- 安全性和完整性要求。在定义信息需求和处理需求的同时必须确定数据安全性和完整性约束。

(2) 确定需要的表

数据库设计过程中最主要的一步就是确定需要的表。分析的过程实际上就是对收集到的数据进行抽象的过程。

(3) 确定所需字段

确定字段时应注意:每个字段直接和表的实体相关;以最小的逻辑单位存储信息;表中的字段必须是原始数据;确定主关键字字段。

(4) 确定联系

例如,在学生表中,“学号”是主关键字,而在成绩表中,“学号”就是外部关键字。

因此,需要分析各个表所代表的实体之间的联系。要建立两个表的联系,可以把其中一个表的主关键字添加到另一个表中,使两个表都含有该关键字。具体方法如下:

• 一对多联系

例如,在公司管理数据库中,部门表和职工表之间存在一对多联系,将部门表中的部门编码字段添加到职工表中。

• 多对多联系

为了避免数据重复存储,同时还要保持多对多联系,可以创建第三个表。所创建的第三个表包含两个表的主关键字,在两个表之间起着纽带的作用,称为“纽带表”。多对多的联系可以分解成两个一对多联系。

• 一对一联系

例如,学校图书馆的读者就是职工和学生,可以把职工表中的职工号和学生表中的学