

推荐教材与自学用书

Visual FoxPro 6.0

程序设计基础

孙晨霞 郭彦红 苑迎春 编著

- ◆ 可完全不懂 Visual FoxPro 6.0
- ◆ 直接学习 Visual FoxPro 6.0 数据库程序设计
- ◆ 轻松开启数据库程序设计之路
- ◆ 牢固打造数据库程序设计基础



清华大学出版社

Visual FoxPro 6.0

程序设计基础

孙晨霞 郭彦红 苑迎春 编著

清华大学出版社
北 京

内 容 简 介

本书是一本全面介绍 Visual FoxPro 6.0 语言和环境特点及其应用技术的实用教材。编写本书的目的有两个,一是使广大读者能够全面系统地了解 Visual FoxPro 的技术特点,二是通过 Visual FoxPro 6.0 使读者掌握数据库的基本概念、理论和应用模式。

全书共分 15 章。前两章介绍了数据库的基本理论和 Visual FoxPro 6.0 的语言因素。第 3 章~第 6 章循序渐进地介绍了 Visual FoxPro 6.0 的表、数据库及其相关操作。第 7 章~第 11 章介绍 Visual FoxPro 6.0 中基于 GUI 的数据库应用程序设计及程序调试方法。最后两章介绍了应用程序打包和帮助文件的制作。

本书适合作为普通高等院校非计算机专业的数据库应用技术教材,也可作为计算机技术爱好者自学的参考书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Visual FoxPro 6.0 程序设计基础/孙晨霞,郭彦红,苑迎春编著.—北京:清华大学出版社,2003
ISBN 7-302-07492-5

I.V… II.①孙… ②郭… ③苑… III.关系数据库—数据库管理系统, Visual FoxPro 6.0
—程序设计—教材 IV.TP311.138

中国版本图书馆 CIP 数据核字(2003)第 097445 号

出 版 者:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

地 址:北京清华大学学研大厦

邮 编:100084

客户服务:010-62776969

组稿编辑:应 勤

文稿编辑:张彦青

封面设计:陈刘源

印 刷 者:北京人民文学印刷厂

装 订 者:三河市李旗庄少明装订厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:23.75 字数:565 千字

版 次:2003 年 12 月第 1 版 2003 年 12 月第 1 次印刷

书 号:ISBN 7-302-07492-5/TP·5523

印 数:1~5000

定 价:32.00 元

前 言

在信息技术快速发展的今天，作为信息处理不可缺少的手段——数据库技术，越来越深入到人们的工作和日常生活当中。但是，数据库系统的理论和深入实践又是非常复杂的。对于非计算机专业的人来说，如果遵循计算机专业的模式来学习数据库系统相关的理论和技术的话，由于缺乏足够的背景知识和实践经验，学习起来会很困难，难以达到预期的效果。如果采用一种以实践为主，在实践中学习理论，并翻过来用实践验证理论，然后又在实践中进一步应用理论的方式，则能够激发同学们的学习兴趣，有利于快速进入数据库技术的世界。

不难看出，这种模式的关键是实践环节，这就要求我们选取一种既功能强大、实用并充分体现数据库理论，又操作方便、规模适度、适用于教学的数据库管理系统作为教学和学习工具。在众多的数据库管理系统中，我们选择了 Visual FoxPro 数据库系统。

Visual FoxPro 数据库系统是数据库技术与可视化的面向对象编程技术的有机结合。一方面，它提供了基于交互式操作的数据库管理的完整方案，无论是初学者还是有一定的 Visual FoxPro 使用经验的老用户，使用起来都会得心应手。封装完好的生成器工具，可以快速生成各种功能完好的数据处理接口，即使是初学者也能在不编程的情况下，快速完成各种数据处理工作。另一方面，对于爱好编程的用户或者程序员来说，它提供了支持事件驱动的可视化面向对象程序设计。另外该系统对计算机硬件和软件环境的要求较低，便于组织教学。

本着实践和理论并重的原则，本书在内容和组织上具有下列特点：

- **注重实用性** 在本书的各个章节，尤其是关键章节中，给出了大量的应用实例，其中有些例子几乎就可以实际应用。在给出这些例子时，考虑到读者的非计算机专业背景，还对例子做出了剖析，并引导读者进一步思考，帮助读者尽快进入角色。
- **注重系统化** 在本书中，几乎所有的讲解和例题都是以书籍管理为背景，随着章节内容的逐步深入，在潜移默化中，引导读者完成表的基本操作、数据库设计和实现、信息查询、图形用户界面的设计、报表统计和编译打包发布应用程序等工作。
- **注重实践性** 每一章的后面，都有一定量的练习题，并且几乎各个章节的练习题都以“学生成绩管理”作为应用对象。读者在学习过程中，如果能亲自上机调试书中的例题和练习题，几乎就等于完成两个小型数据库应用系统的开发。
- **注重理论结合实践** 学习知识和技术，不仅要知其然，还要知其所以然。所以，书中在相关的章节，结合例题给出了有关的关系数据库基本理论和面向对象程序设计的方法。

除了上述特点以外，书中还就数据表的各种操作，给出了 SQL 操作实例，并鼓励读

者多使用 SQL 语句。

全书共分 15 章。前两章介绍了数据库的基本理论 Visual FoxPro 6.0 的语言因素。第 3 章~第 6 章循序渐进地介绍了 Visual FoxPro 6.0 的表、数据库及其相关操作。第 7 章~第 11 章介绍了 Visual FoxPro 6.0 中基于 GUI 的数据库应用程序设计及程序调试方法。最后两章介绍了应用程序打包和帮助文件的制作。

本书适合作为普通高等院校非计算机专业的数据库应用技术教材,也可作为计算机技术爱好者自学的参考书。

最后,由于作者水平有限,编写时间仓促,书中难免有疏漏和不足之处,恳请广大读者提出宝贵意见。

编者
2003 年 6

目 录

第 1 章 概述.....	1	2.4 变量.....	23
1.1 数据处理技术的发展.....	1	2.4.1 内存变量.....	24
1.1.1 数据处理的发展.....	1	2.4.2 字段变量.....	27
1.1.2 数据库技术的发展.....	3	2.4.3 系统变量.....	28
1.2 关系数据库基础.....	4	2.4.4 变量的命名规则.....	28
1.2.1 现实世界的描述.....	4	2.5 数组.....	28
1.2.2 关系模型.....	5	2.5.1 创建数组.....	29
1.3 认识 Visual FoxPro.....	7	2.5.2 数组元素的赋值.....	30
1.3.1 Visual FoxPro 发展简介.....	7	2.5.3 对数组操作的系统函数.....	31
1.3.2 Visual FoxPro 6.0 的 语言特点.....	8	2.5.4 数组与表.....	32
1.3.3 Visual FoxPro 6.0 的 功能特点.....	9	2.6 运算符.....	33
1.3.4 Visual FoxPro 6.0 的集成 开发环境.....	9	2.6.1 算术运算符.....	33
1.4 习题.....	14	2.6.2 字符串运算符.....	33
第 2 章 Visual FoxPro 6.0 编程基础.....	15	2.6.3 逻辑运算符.....	33
2.1 Visual FoxPro 程序文件.....	15	2.6.4 关系运算符.....	34
2.1.1 创建程序文件.....	15	2.6.5 日期和时间运算符.....	35
2.1.2 保存程序.....	16	2.6.6 运算符的优先级.....	37
2.1.3 编辑程序文件.....	17	2.6.7 NULL 值的问题.....	38
2.1.4 运行程序文件.....	18	2.6.8 程序控制结构.....	39
2.2 数据类型.....	19	2.7 过程和函数介绍.....	45
2.2.1 内存变量和表字段中都 可用的数据类型.....	20	2.7.1 过程.....	45
2.2.2 只在表字段当中出现的 数据类型.....	21	2.7.2 函数.....	48
2.3 常量.....	22	2.8 习题.....	53
2.3.1 认识常量.....	22	第 3 章 表的创建与维护.....	55
2.3.2 使用预处理命令定义常量.....	23	3.1 自由表的创建.....	55
2.3.3 释放常量.....	23	3.1.1 基本概念.....	55
		3.1.2 选择字段和字段的 数据类型.....	56
		3.1.3 利用【表设计器】 创建新表.....	57
		3.1.4 利用表向导创建新表.....	59
		3.1.5 使用 SQL 命令创建新表.....	60

3.2 输入数据	60	5.1.2 系统分析	90
3.2.1 在数据表的尾部追加记录	62	5.1.3 系统设计	90
3.2.2 在数据表的指定的位置上 插入记录	64	5.1.4 系统实施	90
3.2.3 用 SQL 语句插入记录	64	5.1.5 系统运行与维护	91
3.3 表数据编辑、修改	65	5.2 用 Visual FoxPro 6.0 开发信息 系统的基本步骤	91
3.3.1 交互方式的数据修改 和浏览	65	5.3 【项目管理器】概述	93
3.3.2 用 Visual FoxPro 命令 修改数据	68	5.4 建立项目的实例	95
3.3.3 用 SQL 语句修改数据	69	5.5 【项目管理器】的使用	96
3.4 记录删除	70	5.5.1 【数据】选项卡	96
3.5 修改表结构	72	5.5.2 【文档】选项卡	97
3.5.1 利用【表设计器】修改 表结构	72	5.5.3 【代码】选项卡	97
3.5.2 使用 SQL 命令修改表结构	73	5.5.4 【其他】选项卡	98
3.6 表文件的删除	73	5.5.5 利用项目查找文件	99
3.7 数据的计数、求和及汇总的实例	74	5.5.6 文件的创建修改	99
3.8 多工作区操作	75	5.5.7 文件的添加及删除	100
3.8.1 工作区的标识	75	5.5.8 浏览表文件中的内容	100
3.8.2 工作区的选择和使用	76	5.5.9 项目间的文件共享	100
3.8.3 工作区的互访	77	5.6 定制【项目管理器】	100
3.8.4 设置表间的临时关系	78	5.6.1 【项目管理器】的停泊	100
3.9 习题	78	5.6.2 【项目管理器】的折叠	101
第 4 章 表的索引和排序	80	5.6.3 分离【项目管理器】 选项卡	101
4.1 何谓索引	80	5.6.4 【项目管理器】的“双击” 操作功能设置	102
4.2 索引的类别	81	5.7 习题	102
4.3 建立索引	82	第 6 章 数据库的创建和管理	104
4.4 使用索引	84	6.1 设计图书数据库	104
4.5 索引文件	85	6.1.1 书籍管理系统的组成	105
4.5.1 索引文件类型	85	6.1.2 确定实体	105
4.5.2 常用索引命令	86	6.1.3 分割表	106
4.6 表的排序	87	6.1.4 建立表间的关系	106
4.7 习题	87	6.2 数据库的创建和管理	108
第 5 章 使用【项目管理器】	89	6.2.1 创建数据库	108
5.1 管理信息系统概述	89	6.2.2 管理数据库	110
5.1.1 系统规划	89	6.3 在数据库中添加和移去表	110
		6.3.1 添加数据表	110
		6.3.2 移去数据表	112

6.4 使用数据库表的新功能设置		7.3 使用 SELECT-SQL 语句实现	
字段属性	113	查询	157
6.4.1 设置字段标题	113	7.3.1 简单的 SELECT 语句	157
6.4.2 设置字段输入掩码和		7.3.2 多表查询	158
显示格式	114	7.3.3 统计、分组和排序输出	
6.4.3 为字段输入注释	116	结果	159
6.4.4 设置字段默认值	116	7.3.4 输出查询结果	160
6.4.5 设置有效性规则和		7.4 设计视图	160
有效性说明	117	7.4.1 创建本地视图	161
6.5 设置表属性	120	7.4.2 【更新条件】选项卡	162
6.5.1 长表名	120	7.4.3 定制视图	165
6.5.2 表注释	121	7.5 习题	167
6.5.3 触发器	121		
6.6 设置表间的关联与参照完整性		第 8 章 表单	169
规则	122	8.1 面向对象程序设计的基本概念	169
6.6.1 建立永久性关联	123	8.1.1 对象、属性、事件和方法	170
6.6.2 设置参照完整性	124	8.1.2 类	170
6.6.3 用 SET RELATION 命令		8.1.3 事件驱动的程序运行机制	171
建立临时关联	129	8.2 用表单向导创建表单	172
6.7 习题	130	8.2.1 使用单表向导建立表单	172
第 7 章 查询与视图	132	8.2.2 使用一对多表单向导	
7.1 设计查询	132	建立表单	175
7.1.1 启动【查询设计器】	132	8.3 使用【表单设计器】建立表单	178
7.1.2 【查询设计器】介绍	133	8.3.1 用快速表单添加字段	178
7.1.3 【字段】选项卡	134	8.3.2 【表单设计器】工具栏	179
7.1.4 【联接】选项卡	136	8.3.3 使用【表单控件】工具栏	
7.1.5 【筛选】选项卡	138	添加控件	180
7.1.6 【排序依据】选项卡	141	8.3.4 使用控件生成器添加控件	181
7.1.7 【分组依据】选项卡	142	8.3.5 从数据环境中向表单添加	
7.1.8 【杂项】选项卡	143	控件	182
7.1.9 定义查询结果的输出		8.3.6 对控件的操作	185
方式	144	8.3.7 设置属性和编写事件、	
7.1.10 对查询文件的操作	147	方法代码	188
7.2 使用查询向导建立查询文件	150	8.4 表单的使用	190
7.2.1 启动查询向导	150	8.4.1 表单类型和使用方法	190
7.2.2 标准查询向导	150	8.4.2 表单外观及行为控制	
7.2.3 图形向导	153	属性	191
7.2.4 交叉表格向导	155	8.4.3 表单的管理	192
		8.4.4 表单中对象的引用方法	195

8.5 Visual FoxPro 常用事件与属性.....199	绑定控件..... 229
8.5.1 常用事件199	9.10.1 ActiveX 控件 230
8.5.2 常用属性 :.....199	9.10.2 ActiveX 绑定控件 231
8.6 容器对象和控件对象.....200	9.11 表单设计举例..... 233
8.6.1 容器对象和控件对象的 概念200	9.11.1 改变鼠标的形状 233
8.6.2 集合和记数属性.....201	9.11.2 数据维护的表单 233
8.7 习题202	9.12 使用外部控件 236
第 9 章 常用控件204	9.13 习题..... 240
9.1 控件简介204	第 10 章 多用户应用程序设计 244
9.1.1 控件的常见属性.....204	10.1 共享数据访问方式..... 244
9.1.2 数据绑定控件与非数据 绑定控件205	10.2 锁定数据..... 246
9.1.3 在设计时如何适时选择 合适的控件205	10.2.1 锁定类型..... 246
9.2 显示信息的控件.....206	10.2.2 锁定方式..... 246
9.2.1 标签控件206	10.2.3 管理锁定冲突..... 248
9.2.2 线条控件208	10.2.4 解锁..... 250
9.2.3 形状控件208	10.3 数据缓冲..... 251
9.2.4 图像控件209	10.3.1 记录缓冲与表缓冲 251
9.3 命令按钮和命令按钮组.....210	10.3.2 缓冲的锁定方式..... 252
9.3.1 命令按钮210	10.3.3 启用缓冲..... 252
9.3.2 命令按钮组212	10.3.4 使用缓冲..... 255
9.4 接收用户输入的控件.....214	10.4 事务处理..... 257
9.4.1 文本框214	10.4.1 简单事务处理..... 258
9.4.2 编辑框216	10.4.2 嵌套事务处理..... 259
9.5 给出一组确定的值的控件.....217	10.5 习题..... 261
9.5.1 选项按钮组217	第 11 章 程序调试 264
9.5.2 复选框219	11.1 错误类型..... 264
9.5.3 列表框220	11.1.1 语法错误 264
9.5.4 组合框224	11.1.2 逻辑错误 266
9.6 微调控件226	11.1.3 运行错误 267
9.7 计时器控件226	11.1.4 Visual FoxPro 中程序的 主要状态 270
9.8 表格控件227	11.2 【调试器】介绍..... 272
9.8.1 表格的常用属性.....227	11.2.1 打开【调试器】 272
9.8.2 对表格的列进行调整.....228	11.2.2 【调试器】窗口的主要 元素 272
9.9 页框控件229	11.3 调试程序..... 276
9.10 ActiveX 控件和 ActiveX	11.3.1 跟踪程序运行 276

11.3.2 设置断点.....	277	13.5 报表控件.....	309
11.3.3 监视程序.....	279	13.5.1 常用的报表控件介绍.....	310
11.3.4 使用【调用堆栈】 窗口.....	280	12.5.2 加入报表控件.....	310
11.3.5 使用【输出】窗口.....	280	13.5.3 域控件的格式.....	314
11.3.6 使用【事件跟踪】 对话框.....	281	13.5.4 控件的字体、颜色、 背景色等属性的设置.....	315
11.4 处理运行时错误.....	283	13.5.5 图片及 OLE 对象的 报表输出.....	316
11.4.1 预防错误.....	283	13.6 报表中插入日期、页码.....	317
11.4.2 错误处理.....	284	13.7 报表预览和打印.....	318
11.5 习题.....	287	13.8 报表设计举例.....	319
第 12 章 菜单和工具栏.....	289	13.9 标签的设计.....	319
12.1 菜单系统设计.....	289	13.10 习题.....	320
12.1.1 菜单设计器的使用.....	290	第 14 章 编辑与发布应用程序.....	322
12.1.2 菜单设计示例.....	292	14.1 设计主文件.....	322
12.1.3 创建快捷菜单.....	294	14.2 连编.....	325
12.2 创建自定义工具栏.....	295	14.2.1 包含或排除文件.....	325
12.2.1 定义工具栏类.....	295	14.2.2 连编.....	326
12.2.2 在自定义工具栏类中 添加对象及设置属性.....	296	14.3 制作安装程序.....	328
12.2.3 设置自定义工具栏操作.....	297	14.3.1 制作安装程序的过程.....	329
12.2.4 在表单集中添加工具栏.....	297	14.3.2 使用【安装向导】创建 发布磁盘.....	330
12.3 协调菜单和自定义工具栏.....	298	14.4 习题.....	336
12.4 将相关的工具栏和菜单添加到 表单集中.....	299	第 15 章 帮助文件的制作.....	338
12.5 习题.....	299	15.1 HTML 样式的帮助.....	338
第 13 章 报表与标签设计.....	301	15.1.1 制作帮助文件的基本 过程.....	338
13.1 创建一个报表文件.....	301	15.1.2 如何使用帮助文件.....	343
13.2 【报表设计器】介绍.....	302	15.2 DBF 样式的帮助.....	346
13.2.1 【报表设计器】的 窗口布局.....	302	15.2.1 设计帮助数据表.....	346
13.2.2 【报表】菜单.....	304	15.2.2 使用.dbf 样式的帮助.....	347
13.2.3 【报表设计器】工具栏.....	305	15.2.3 【请参阅】交叉引用.....	347
13.3 报表布局设计.....	305	15.2.4 定制.dbf 样式帮助.....	348
13.4 报表数据源.....	307	15.2.5 控制【帮助】窗口的 位置.....	349
13.4.1 建立报表数据源.....	307	15.3 习题.....	349
13.4.2 报表输出的数据控制.....	308		

附录 A Visual FoxPro 中的文件 类型	附录 C 【选项】对话框的选项	351	355
附录 B Visual FoxPro 系统容量		353	

第 1 章 概 述

本章旨在引导用户建立有关数据处理和关系型数据库的基本概念，并从总体上认识 Visual FoxPro 的语言特点和功能特点，熟悉 Visual FoxPro 集成开发环境和【项目管理器】的使用方法。

本章学习重点

- 关系数据库的基本知识
- Visual FoxPro 的语言特点和功能特点
- Visual FoxPro 集成开发环境

1.1 数据处理技术的发展

在现实生活和工作中，人们会遇到各式各样的问题。有些问题涉及到数学运算，有些问题涉及到逻辑运算，也有些问题涉及到大批量数据的处理。例如，在学校里，每到期末，需要将各个班级学生的单科成绩排队、综合成绩排队、将同一个年级中所有同学的成绩排队、或者将一个年级中各个班级的单科平均成绩或者综合平均成绩排队。教师更是要将某课程期末考试的成绩进行分析，观察考试结果的分布情况，分析各个班级的差异等等，与此类似的数据处理工作在生活中到处都是。长期以来，我们习惯了“白纸+笔+计算器”的处理方式。而这种处理方式需要花费很多精力和时间，处理的效果也不一定理想，处理的结果以书面文档的形式保存，而且每学期都要重复同样的劳动。将学生成绩处理及其类似的问题用计算机进行处理，是解决问题重要途径。那么用什么样的技术实现这样的数据处理好呢？答案是数据库。

在了解数据库概念之前，我们先从数据处理发展的过程当中认识数据处理。从数据处理演变的过程，不难看出数据库技术的历史地位和发展前景。

1.1.1 数据处理的发展

数据处理是指对数据进行收集、组织、加工、存储、抽取和传播等工作，其基本目的是从大量的、杂乱无章的甚至是难以理解的数据中抽取并推导出对于某些特定的用户来说是有价值的、有意义的数据，为进一步的活动提供决策的依据。从远古到现在，数据处理一直是社会生活中必不可少的内容，随着人类生产力的进步，数据处理经历了 3 个阶段。第一阶段是手工处理，如结绳记事、垒石记事等朴素的数据处理手段。第二阶段是机械处理，如我国古代利用算盘，国外使用计算尺处理数据。计算机的出现，标志着数据处理方法跨进了“电子数据处理”阶段，即数据处理的第三阶段。数据处理手段的日趋现代化，更加扩展了这一领域的应用深度和广度。随着计算机硬件和软件技术的不断发展，使用计算机进行数据处理

先后经历了程序管理方式、文件系统方式和数据库系统方式 3 个阶段。

1. 程序管理方式阶段

电子数据管理技术发展第一个阶段(大约在 20 世纪 50 年代中期前一阶段)为程序管理方式阶段,这里的程序是指用户应用程序。这一阶段由程序员来规定数据的逻辑结构,在程序中设计数据存储的物理结构。程序中存取数据的子程序随着数据存储的改变而改变,即数据与程序不具有独立性。程序员不仅必须在数据的物理布置上花费精力,而且在数据存储有变化时,必须修改程序。该阶段数据处理的主要特点是:

- 数据不保存;
- 数据不能独立存在,它是程序的组成部分,即数据和程序完全结合成一个不可分割的整体;
- 数据是面向应用的,不同应用的数据之间是相互独立、彼此无关的,即使不同应用涉及到相同的数据,也必须各自定义、无法互相利用、互相参照。数据不仅高度冗余,而且不能共享。

2. 文件系统阶段

从 20 世纪 50 年代后期到 20 世纪 60 年代中期,操作系统中出现了文件系统。在进行数据处理时,将大量的数据保留在外存的文件中,文件的逻辑结构与存储结构由系统进行转换,程序与数据有一定的独立性。程序员可以把更多的精力放在算法上,而不必过多地考虑数据的物理细节。

文件的方式较第一阶段有了很大的改进,但是还有下面的主要缺点:

- 数据冗余大。文件系统文件基本上对应于某个应用程序,当多个应用程序需要的部分数据相同时,必须各自建立自己的数据文件,数据冗余较大;
- 数据和程序独立性差。由于文件是为特定应用设计的,不具有灵活性。想要为现有的数据添加新的应用几乎不可能。

3. 数据库系统阶段

从 20 世纪 60 年代后期开始,出现了数据库技术。利用数据库系统方式实现数据处理是在文件系统的基础上发展起来的先进技术,它克服了文件系统的弱点,提供了一个完善的、高级的数据管理技术,实现对数据集中统一的独立管理,使数据的存储和维护不受任何用户的影响,这种技术具有如下基本特点:

- 数据的结构化。例如,学生的学籍管理过程中,涉及到学生信息、课程信息、选课信息、专业信息。将这些信息结构化如图 1.1,这种不仅描述数据本身,还描述数据之间的关系的做法,就是将数据结构化。
- 数据冗余度小,易扩充。数据的结构化打破了文件系统中不同应用都需要的部分的数据不能重用的局面,减少了重复数据。
- 程序与数据独立性较高。在数据库系统中,将数据的定义和描述从应用程序中分离。应用程序通过数据库的外部逻辑映像与之交互,数据的物理存储变化不会直接影响到应用程序。

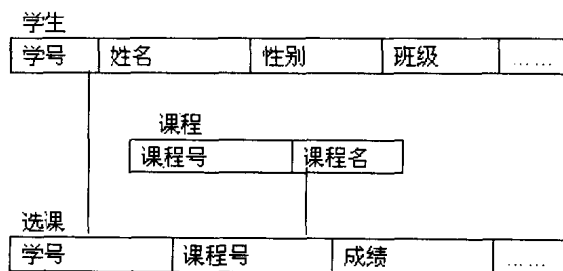


图 1.1 学生选课信息

除此以外，它还有统一的数据控制功能，数据的最小存取单位是数据项等特点。在实际应用中，一个完整的数据库系统通常由数据库、数据库管理系统和数据库应用系统组合而成，下面是这些基本概念的具体含义。

- **数据库(Database):** 是按一定组织方式存储在一起的相关数据的集合。也就是说用户要处理的数据都按一定组织方式存储在数据库里。
- **数据库管理系统(DBMS, Database Management System):** 是对数据库中的数据进行统一管理、控制和维护的数据库管理软件。用户通过数据库管理系统可以使用数据库中的数据，实现对数据的存取和检索操作。
- **数据库应用系统(DBAS, Database Application System):** 是根据用户任务需要用高级语言编写的可以实现对数据库中的数据进行存取和检索操作的程序。这样用户就可以不必直接操作数据库管理系统，而可以通过界面友好的数据库应用程序来实现数据操作。

本书中要介绍的 Visual FoxPro 既是一种数据库管理系统，同时又自备有编写应用程序的语言，它提供了与高级语言中各种语句相类似的命令。这样用户既可以在 Visual FoxPro 集成开发环境中直接管理数据，又可以使用这些命令编写应用程序，通过应用程序管理数据。

利用数据库技术实现各种数据处理，是目前流行的数据处理方式。这种技术不但正在公司的日常办公、企业经营管理、学校的学生管理、银行财务处理、医药收费管理等各个方面深入应用，还正在延伸到社会的各个角落，如家庭财务、家庭物品等家政管理，因此数据库技术有着广阔的应用空间。

1.1.2 数据库技术的发展

数据库技术及其应用系统最早出现于 20 世纪 60 年代。70 年代，E·F·Codd 先生提出了数据库的关系模型，并对关系理论做出了创造性的贡献，从而使数据库从技术上升到理论。在数据库发展史上，20 世纪 60 年代末 70 年代初出现了 3 个重要事件，标志着数据库技术日益成熟，并有了坚实的理论基础。

- 1968 年 IBM 公司开发数据库管理系统的商品化软件 IMS(Information Management Systems)，IMS 的数据模型是层次结构的。
- 1969 年美国数据系统语言协商会 CODASYL(Conference On Data System Language)下属的数据库任务组 DBTG(Data Base Task Group)对数据库方法进行系统的研究、讨论，发表了 DBTG 报告。DBTG 报告确定并建立了数据库系统的许多概念、方法和技术。DBTG 所提议的方法是基于网状结构的，它是数据库网状模型的基础和典

型代表。

- 1970 年 IBM 公司的 San Jose 研究实验室的研究员 E·F·Codd 发表了题为“大型共享数据库数据的关系模型”论文,提出了数据库的关系模型,开创了数据库关系方法和关系数据库理论的研究,为数据库技术奠定了理论基础。由于 E·F·Codd 的杰出贡献,他于 1981 获得了 ACM 图灵奖。

20 世纪 70 年代,数据库技术有了很大的进展,关系方法的理论研究和软件系统的演示取得了很大的成果。到了 80 年代,几乎所有新开发的系统均是关系系统,微机的关系数据库管理系统也越来越丰富,性能越来越好,功能越来越强,他的应用也普及到各个领域。

20 世纪 90 年代,关系数据库和面向对象技术相结合,出现了所谓的对象-关系数据库系统(Object-Relational Database Systems),这种数据库系统除了支持常规的数据类型外,还支持正文、视频和声频等信息的存储和管理。

目前,随着计算科学技术飞速发展,数据管理作为计算机科学技术的一个重要分支,将一直发展下去。社会信息化程度越高,对数据管理系统的要求也越高,数据库作为主要的数据库管理手段,也在不断的发展和演进当中。

1.2 关系数据库基础

数据是对各种物理的或者抽象的事物的描述,采用关系数据库来组织数据、描述事物时,通过实体联系模型描述现实事物之间的关系,采用关系模型将事务以及事物之间的关系抽象化。

1.2.1 现实世界的描述

数据库是用来存储数据的,数据库中存储的数据来自于对现实世界的抽象描述。在对现实世界进行抽象描述时,经常用到下面的概念。

1. 实体

在现实世界中客观存在并可以相互区分的事物叫做实体。实体可以是人,也可以是物;可以是实际的对象(如汽车、房子),也可以是某些抽象的概念(如一门课、一次会议);可以是事物本身(如学生、课程),也可以是事物之间的联系(如学生选课)。

2. 属性

属性是实体具有的某种特性,是对实体的抽象描述。例如学生这个实体具有学号、姓名、性别、出生日期、入学日期、班级等属性,它们是对学生实体的抽象描述。

3. 实体之间的联系

现实世界的事物之间往往具有某种联系。这种实体之间的联系可分为 3 类:

(1) 一对一联系(1 : 1)

如果实体集 A 中的每一个实体最多只能和实体集 B 中的一个实体有联系,反之也成立,则称实体集 A 和实体集 B 具有一对一联系,记为 1 : 1。例如,居民及其身份证之间的联系就是一对一联系。

(2) 一对多联系(1 : n)

如果实体集 A 中的每个实体可以和实体集 B 中的 n 个实体($n>0$)有联系,但是反过来,实体集 B 中的一个实体只能和实体集 A 中的一个实体有联系,则称实体集 A 和实体集 B 具有一对多联系,记为 1 : n。例如,学生和班级之间就是一对多联系,一个学生只能属于一个班级,而一个班级可以有多个学生。

(3) 多对多联系(m : n)

如果实体集 A 中的每个实体可以和实体集 B 中的 n 个实体($n>0$)有联系,反过来,实体集 B 中的每个实体也可以和实体集 A 中的 m 个实体($m>0$)有联系,则称实体集 A 和实体集 B 具有多对多联系,记为 m : n。例如,学生和课程之间就是多对多联系,一个学生可以学习多门课,一门课也可以被多个学生学习。

4. 实体联系模型

了解了客观事物及其联系后,就可以着手建立实体联系模型进行描述。该模型用来对实体、实体的属性以及实体间的联系进行描述。E-R 图(Entity-Relationship, 也称实体联系图)是描述实体联系模型的常用工具。

下面以教学情况为例来建立实体联系模型。教学情况包括“学生”、“课程”、“教师”、“学习”、“任课”等方面。其中,“学生”实体具有“学号”、“姓名”、“性别”、“出生日期”、“入学日期”、“班级”等属性;“课程”实体具有“课程号”、“课程名称”等属性;“教师”实体具有“教师编号”、“姓名”、“职称”等属性;“学习”实际是“学生”实体与“课程”实体之间的一种联系,它也可看作一种实体,并具有“学号”、“课程号”、“成绩”等属性;“任课”是“教师”与“课程”之间的一种联系,该实体具有“课程号”、“教师编号”等属性。

“学生”和“课程”之间是多对多的联系,“教师”和“课程”之间也是多对多的联系。图 1.2 为描述该实体联系模型的 E-R 图。图中的矩形代表基本实体,椭圆代表属性,菱形代表联系。

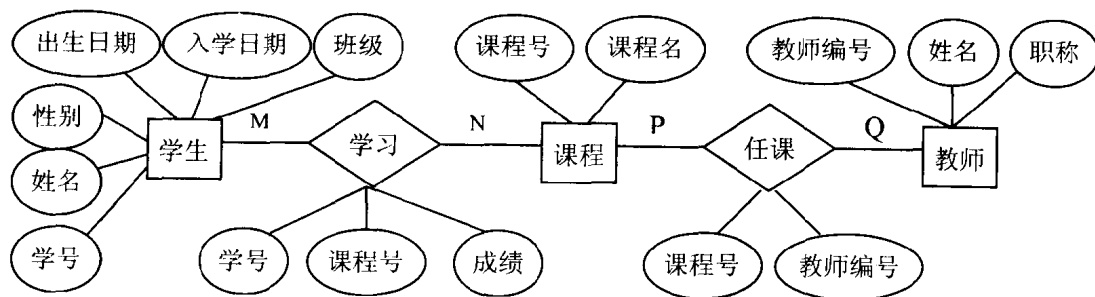


图 1.2 E-R 图

1.2.2 关系模型

使用实体联系模型可以对现实世界中的事物及其联系进行描述,而在设计数据库时,需要用数据模型对现实世界进行抽象。数据模型就是数据库系统中用于提供信息表示和操作手段的形式框架,是一种形式化描述数据、数据之间的联系,以及有关语义约束的方法。数据模型是实体联系模型的数据化,数据库设计的核心问题之一就是要设计一个好的数据模型。

在数据库技术发展过程中,先后出现过多种数据模型,其中以时下流行的关系模型最为引人注目,目前,它已经成为数据库设计事实上的标准。以关系模型建立起来的数据库称为关系型数据库,Visual FoxPro 就是当前流行的一种关系型数据库管理系统。下面介绍关系模型的有关概念。

数据模型通常由数据结构、数据操作和完整性约束条件 3 部分组成,作为数据模型的一种,关系模型也不例外。

1. 数据结构

数据结构是所研究的对象类型的集合,在数据库系统中通常按照数据结构的类型来命名数据模型。在关系数据模型中,无论是实体还是实体间的联系均由单一的结构类型——关系来表示。一个关系可以看作是一个二维表,表中的每一行表示一个实体,称为一条记录;表中的每一列是记录中的一个数据项,表示实体的一个属性,称为一个字段。

表 1.1 是一个记载学生基本情况的二维表。在这个二维表中,除标题行外,每一行代表一位学生的记录,共有两条记录,每一列用来说明学生的一个特征,代表一个字段,共有学号、姓名、性别、出生日期、入学日期、班级 6 个字段。

表 1.1 学生表

学 号	姓 名	性 别	出生日期	入学日期	班 级
01	鲁晓飞	女	1980.5.5	1999.9.1	计算机 1 班
02	贾宝玉	男	1981.1.1	1999.9.1	计算机 1 班

表 1.2 和表 1.3 是分别记载课程基本情况和学生学习课程情况的二维表。

表 1.2 课程表

课程号	课程名
0001	数据结构
0002	数据库原理

表 1.3 学习成绩表

学 号	课程号	成 绩
01	0001	90
01	0002	80
02	0001	85
02	0002	88

这样就实现了将图 1.2 所示的实体联系模型中的实体“学生”、“课程”及其联系“学习”转换为关系模型中的二维表的方式进行描述。这 3 个二维表之间通过拥有共同的字段而具有某种关联。例如,学生表和学习成绩表具有共同字段“学号”,课程表和学习成绩表具有共同字段“课程号”。若干个这样相互之间具有一定关联的二维表集合起来就构成了关系型数据库。简而言之,关系型数据库就是若干个二维表的集合,例如上面所示的 3 个表就共同构成学生成绩数据库。