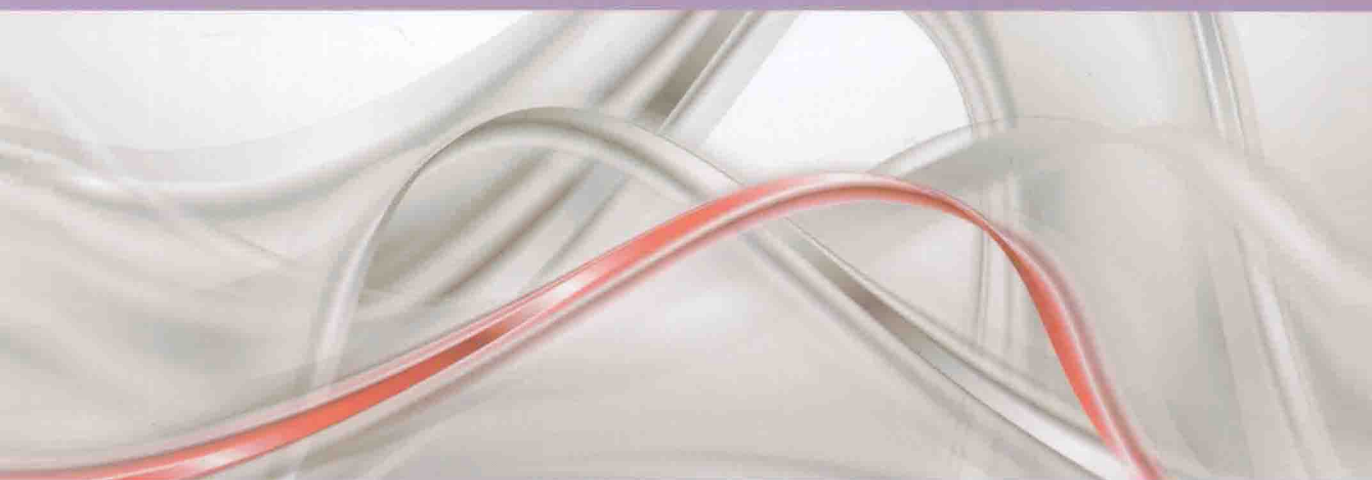




普通高等教育“十二五”重点规划教材 计算机基础教育系列



Visual FoxPro 数据库程序设计实用教程

高昱 李纪文 赵文硕◎主编



 科学出版社

普通高等教育“十二五”重点规划教材·计算机基础教育系列

Visual Foxpro 数据库 程序设计实用教程

	高	昱	李纪文	赵文硕	主	编
赵	慧	宋敏杰	赵	亮	佟	力
	尚丹梅	吴	静	耿	或	参
						编

科学出版社

内 容 简 介

本书根据教育部考试中心《全国计算机等级考试二级 Visual FoxPro 数据库程序设计考试大纲》要求编写,基于多年的教学实践和计算机二级考试培训工作经验,力求做到全面解读 Visual FoxPro 系统面向对象的程序设计和结构化程序设计两种方式,并有意将侧重点从命令窗口的命令操作模式向鼠标操作模式倾斜,以适应 2013 年开始的取消纸考全部上机考试的现状。

本书还配有《Visual FoxPro 数据库程序设计实用实践教程》。理论教程结构更合理,思路更清晰,例题更贴切。实践教程提供大量上机操作例题和习题,操作步骤详细,结构清晰。

本书可作为高等学校非计算机专业学生的学习用书,也可以作为 Visual FoxPro 程序设计的教材使用,还可以作为全国计算机等级考试二级 Visual FoxPro 程序设计的辅导和培训教材。

图书在版编目(CIP)数据

Visual Foxpro 数据库程序设计实用教程/高昱,李纪文,赵文硕主编.

—北京:科学出版社,2014

(普通高等教育“十二五”重点规划教材·计算机基础教育系列)

ISBN 978-7-03-039980-9

I. ①V… II. ①高… ②李… ③赵… 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 040983 号

责任编辑:宋丽 张斌 余梦洁 / 责任校对:刘玉靖

责任印制:吕春珉 / 封面设计:东方人华平面设计部

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

骏立印刷厂印刷

科学出版社发行 各地新华书店经销

*

2014 年 3 月第 一 版 开本:787×1092 1/16

2014 年 8 月第二次印刷 印张:17 1/4

字数:415 000

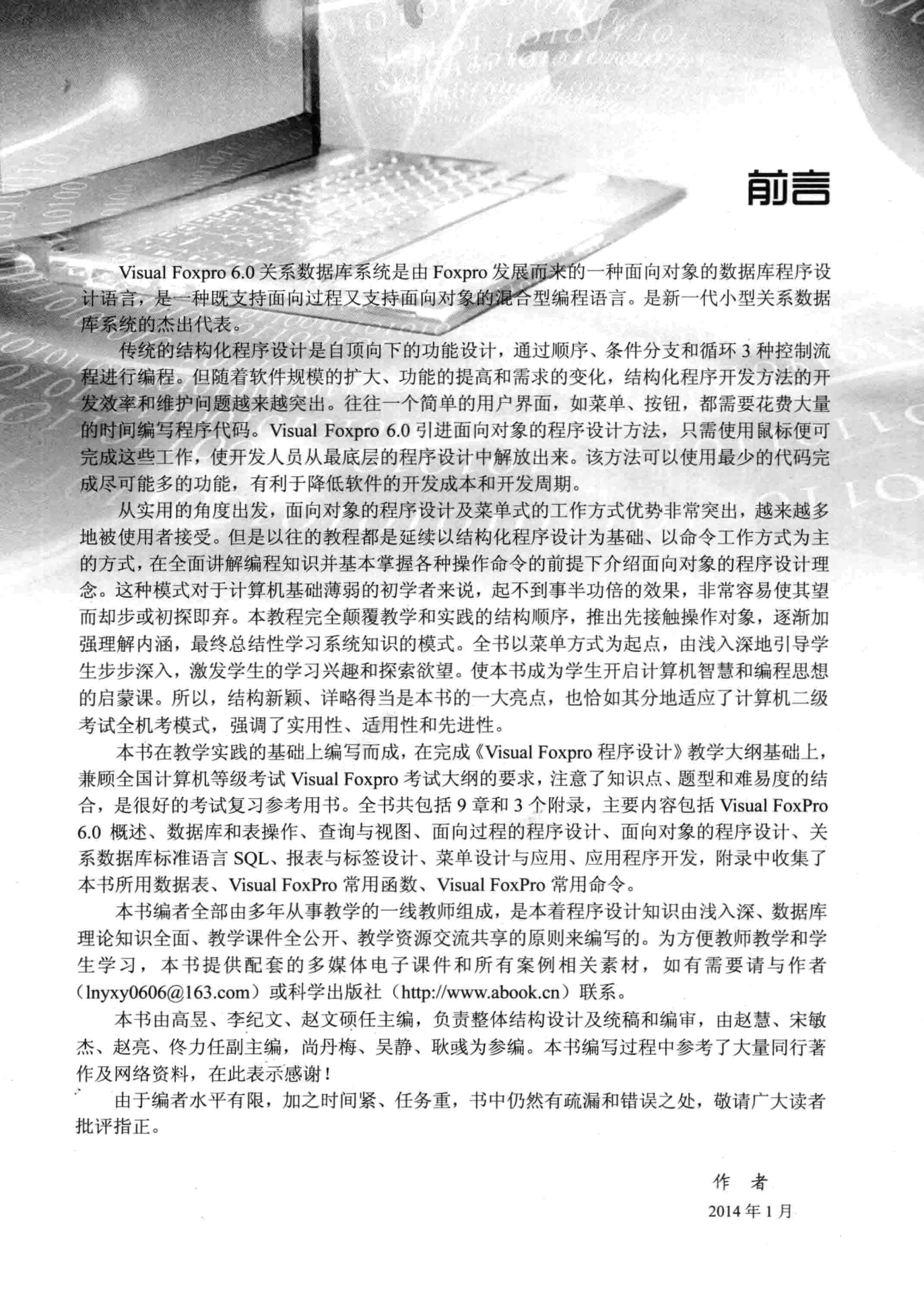
定价:35.00 元

(如有印装质量问题,我社负责调换<骏杰>)

销售部电话 010-62134988 编辑部电话 010-62135741 (VF07)

版权所有,侵权必究

举报电话:010-64030229; 010-64034315; 13501151303



前言

Visual Foxpro 6.0 关系数据库系统是由 Foxpro 发展而来的一种面向对象的数据库程序设计语言，是一种既支持面向过程又支持面向对象的混合型编程语言。是新一代小型关系数据库系统的杰出代表。

传统的结构化程序设计是自顶向下的功能设计，通过顺序、条件分支和循环 3 种控制流程进行编程。但随着软件规模的扩大、功能的提高和需求的变化，结构化程序开发方法的开发效率和维护问题越来越突出。往往一个简单的用户界面，如菜单、按钮，都需要花费大量的时间编写程序代码。Visual Foxpro 6.0 引进面向对象的程序设计方法，只需使用鼠标便可完成这些工作，使开发人员从最底层的程序设计中解放出来。该方法可以使用最少的代码完成尽可能多的功能，有利于降低软件的开发成本和开发周期。

从实用的角度出发，面向对象的程序设计及菜单式的工作方式优势非常突出，越来越多地被使用者接受。但是以往的教程都是延续以结构化程序设计为基础、以命令工作方式为主的方式，在全面讲解编程知识并基本掌握各种操作命令的前提下介绍面向对象的程序设计理念。这种模式对于计算机基础薄弱的初学者来说，起不到事半功倍的效果，非常容易使其望而却步或初探即弃。本教程完全颠覆教学和实践的结构顺序，推出先接触操作对象，逐渐加强理解内涵，最终总结性学习系统知识的模式。全书以菜单方式为起点，由浅入深地引导学生步步深入，激发学生的学习兴趣 and 探索欲望。使本书成为学生开启计算机智慧和编程思想的启蒙课。所以，结构新颖、详略得当是本书的一大亮点，也恰如其分地适应了计算机二级考试全机考模式，强调了实用性、适用性和先进性。

本书在教学实践的基础上编写而成，在完成《Visual Foxpro 程序设计》教学大纲基础上，兼顾全国计算机等级考试 Visual Foxpro 考试大纲的要求，注意了知识点、题型和难易度的结合，是很好的考试复习参考用书。全书共包括 9 章和 3 个附录，主要内容包括 Visual FoxPro 6.0 概述、数据库和表操作、查询与视图、面向过程的程序设计、面向对象的程序设计、关系数据库标准语言 SQL、报表与标签设计、菜单设计与应用、应用程序开发，附录中收集了本书所用数据表、Visual FoxPro 常用函数、Visual FoxPro 常用命令。

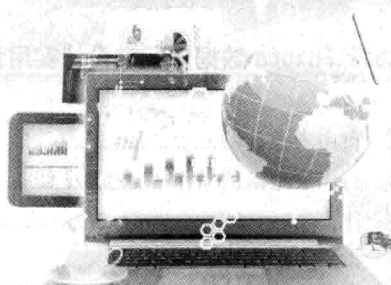
本书编者全部由多年从事教学的一线教师组成，是本着程序设计知识由浅入深、数据库理论知识全面、教学课件全公开、教学资源交流共享的原则来编写的。为方便教师教学和学生学学习，本书提供配套的多媒体电子课件和所有案例相关素材，如有需要请与作者 (lnxy0606@163.com) 或科学出版社 (<http://www.abook.cn>) 联系。

本书由高昱、李纪文、赵文硕任主编，负责整体结构设计及统稿和编审，由赵慧、宋敏杰、赵亮、佟力任副主编，尚丹梅、吴静、耿彧为参编。本书编写过程中参考了大量同行著作及网络资料，在此表示感谢！

由于编者水平有限，加之时间紧、任务重，书中仍然有疏漏和错误之处，敬请广大读者批评指正。

作 者

2014 年 1 月



目录

第 1 章 程序设计基础.....	1
1.1 计算机语言种类及特点.....	1
1.1.1 计算机语言种类.....	1
1.1.2 计算机高级语言种类和特点.....	2
1.2 数据的存储及文件类型.....	2
1.2.1 数据的概念.....	2
1.2.2 计算机硬件存储设备.....	3
1.2.3 数据在计算机硬件上的存储.....	4
1.3 数据库基础.....	4
1.3.1 数据库基本概念.....	5
1.3.2 数据模型.....	5
1.3.3 关系数据库.....	9
1.3.4 关系运算.....	11
1.3.5 数据库设计.....	14
1.4 关系数据库 Visual FoxPro 6.0 初步.....	14
1.4.1 Visual FoxPro 6.0 界面及环境设置.....	14
1.4.2 Visual FoxPro 6.0 工作方式.....	16
1.4.3 Visual FoxPro 项目管理器.....	17
第 2 章 数据库和表操作.....	20
2.1 数据库操作.....	20
2.1.1 建立数据库.....	20
2.1.2 关闭数据库.....	21
2.1.3 打开数据库.....	21
2.1.4 修改数据库.....	23
2.1.5 删除数据库.....	23
2.2 数据库表操作.....	24
2.2.1 建立数据库表.....	24
2.2.2 将数据库表移出.....	29
2.2.3 将自由表移入数据库.....	30
2.2.4 数据库表记录的浏览、修改和追加.....	31
2.2.5 数据库表记录的删除和恢复.....	34
2.2.6 数据库表设计器.....	35
2.3 自由表操作.....	38
2.3.1 建立自由表.....	38

2.3.2	自由表设计器.....	39
2.3.3	自由表记录的操作.....	39
2.4	表索引.....	41
2.4.1	索引文件类型.....	41
2.4.2	索引类型.....	41
2.4.3	建立索引文件.....	42
2.4.4	主控索引.....	44
2.5	创建和编辑关系.....	45
2.5.1	建立和编辑永久性关系.....	45
2.5.2	建立和编辑临时性关系.....	46
2.6	建立参照完整性.....	48
第 3 章	查询与视图.....	50
3.1	查询与视图.....	50
3.1.1	查询.....	50
3.1.2	视图.....	50
3.1.3	查询与视图的区别.....	50
3.2	结构化查询语言 SQL.....	51
3.2.1	SQL 简介.....	51
3.2.2	SQL 的格式.....	51
3.2.3	SQL 命令使用举例.....	52
3.2.4	SQL 语句在 Visual FoxPro 的使用方法.....	53
3.3	查询数据.....	53
3.3.1	建立简单查询.....	53
3.3.2	为查询结果排序.....	57
3.3.3	查询结果的分组.....	59
3.3.4	创建多个表的查询.....	60
3.4	视图查询.....	61
3.4.1	视图文件的建立.....	62
3.4.2	控制视图字段的显示与输入.....	64
3.4.3	为视图添加筛选表达式.....	65
3.4.4	建立远程数据连接.....	65
3.4.5	建立远程视图.....	66
3.4.6	用视图更新数据.....	66
3.4.7	控制更新数据的条件.....	66
3.4.8	控制视图更新的方法.....	67
3.4.9	为视图传递参数.....	67
第 4 章	面向过程的程序设计.....	69
4.1	数据及其运算.....	69

4.1.1 数据类型及非格式化数据输出命令.....	69
4.1.2 常量及其类型.....	70
4.1.3 变量及其类型.....	72
4.1.4 函数及其类型.....	75
4.1.5 数据运算表达式及其类型.....	80
4.2 数据库和表操作命令.....	84
4.2.1 数据库操作命令.....	84
4.2.2 表操作命令.....	85
4.2.3 记录指针定位.....	87
4.2.4 记录显示.....	88
4.3 文件操作命令.....	89
4.4 表记录的修改和维护命令.....	90
4.4.1 记录的修改.....	90
4.4.2 记录的插入和删除.....	90
4.5 表的排序和索引.....	92
4.5.1 排序.....	92
4.5.2 索引.....	93
4.5.3 记录查询.....	96
4.5.4 过滤器命令.....	97
4.6 统计命令.....	98
4.6.1 求和.....	98
4.6.2 求平均值.....	98
4.6.3 计数.....	98
4.6.4 统计.....	98
4.6.5 分类汇总.....	98
4.7 使用多个表.....	99
4.7.1 工作区.....	100
4.7.2 设置表间的临时关系.....	101
4.8 命令文件的建立与运行.....	103
4.8.1 命令文件的建立和编辑.....	103
4.8.2 命令文件的运行.....	103
4.8.3 辅助命令及过程化程序设计规则.....	104
4.8.4 交互式输入命令.....	105
4.8.5 输出命令.....	106
4.9 顺序结构程序设计.....	106
4.10 选择结构程序设计.....	107
4.11 循环结构程序设计.....	111
4.12 过程及其调用.....	116



第 5 章 面向对象的程序设计	124
5.1 面向对象的程序设计概念	124
5.1.1 对象、属性、事件及方法的概念	124
5.1.2 类与子类继承概念	125
5.1.3 容器类与控件类及控件引用	125
5.2 表单	127
5.2.1 表单的概念	127
5.2.2 表单的建立、编辑、保存和运行	128
5.3 控件及其属性设置	133
5.3.1 表单控件基本操作	133
5.3.2 表单控件	135
5.4 常用事件方法	155
5.4.1 常用的事件和方法列表	155
5.4.2 编辑代码	156
5.5 添加新属性和方法	156
5.5.1 创建新属性	156
5.5.2 创建新方法	157
5.6 创建子类	158
5.6.1 菜单方式	158
5.6.2 命令方式	159
5.7 数据表的表单设计	159
5.8 库、表、视图与表单的综合例题	164
第 6 章 关系数据库标准语言 SQL	169
6.1 SQL 概述	169
6.2 定义功能	169
6.2.1 表的定义	169
6.2.2 表结构的修改	172
6.2.3 表的删除	174
6.3 查询功能	174
6.3.1 简单查询	175
6.3.2 联接查询	181
6.3.3 嵌套查询	185
6.3.4 SELECT 语句的几个重要选项	189
6.4 操作功能	190
6.4.1 插入数据	190
6.4.2 更新数据	191
6.4.3 删除数据	191

6.5 视图管理	192
6.5.1 定义视图	192
6.5.2 视图的删除	193
6.5.3 关于视图的说明	194
6.6 SQL 查询与表单综合例题	194
第 7 章 报表与标签设计	200
7.1 报表概述	200
7.1.1 报表样式	200
7.1.2 创建报表方法	200
7.2 利用报表向导创建报表	201
7.2.1 用“报表向导”创建简单报表	201
7.2.2 用“报表向导”创建一对多报表	204
7.3 快速报表	206
7.4 利用报表设计器创建报表	208
7.4.1 利用报表设计器创建报表	209
7.4.2 设计报表	214
7.4.3 输出报表	215
7.5 标签设计	216
7.6 报表与表单综合例题	218
第 8 章 菜单设计与应用	222
8.1 菜单概述	222
8.1.1 菜单结构	222
8.1.2 菜单设计基本过程	222
8.2 系统菜单	224
8.2.1 系统菜单概述	224
8.2.2 系统菜单的配置	225
8.3 下拉式菜单设计	226
8.3.1 定义菜单	226
8.3.2 设置“常规选项”	228
8.3.3 设置“菜单选项”	229
8.3.4 菜单程序的生成和运行	230
8.3.5 制作顶层表单的菜单	234
8.4 快捷菜单设计	235
8.4.1 快捷菜单	235
8.4.2 建立快捷菜单步骤	236
8.5 菜单与表单综合例题	237



第 9 章 应用程序开发.....	245
9.1 应用程序开发的基本步骤.....	245
9.1.1 基本步骤.....	245
9.1.2 主程序设计.....	245
9.1.3 连编应用程序.....	246
9.1.4 控制事件循环.....	247
9.1.5 调试器.....	247
9.2 研究生院信息管理系统开发.....	248
附录.....	250
附录 1 书中所用的数据表.....	250
附录 2 Visual FoxPro 常用函数.....	256
附录 3 Visual FoxPro 常用命令.....	261
参考文献.....	266

第 1 章 程序设计基础

计算机所做的每一次动作、每一个步骤，都是按照已经用计算机语言编好的程序来执行的。程序是计算机要执行的指令的集合，而程序是用我们所掌握的语言来编写的。在过去的几十年间，大量的程序设计语言被发明、取代、修改或组合在一起。尽管人们多次试图创造一种通用的程序设计语言，却没有一次尝试是成功的，因此多种不同的编程语言同时存在着。

不同的计算机语言有各自的特点，编程人员可以根据条件各取所需。选用语言不同可能因为编写程序的目标和应用方向各不相同；也可能因为新程序员与有经验的程序员之间技术差距很大，而选用难易程度不同的语言；还可能因为不同程序之间运行成本的差别。

1.1 计算机语言种类及特点

1.1.1 计算机语言种类

计算机语言（Computer Language）指用于人与计算机之间通信的语言。为了使电子计算机进行各种工作，需要有一套用以编写计算机程序的数字、字符和语法规划，由这些字符和语法规则组成计算机各种指令（或各种语句），这些就是计算机能接受的语言。

计算机语言的种类非常多，总体来说可以分成机器语言、汇编语言、高级语言 3 大类。高级语言向可视化方向发展，如图 1-1 所示。

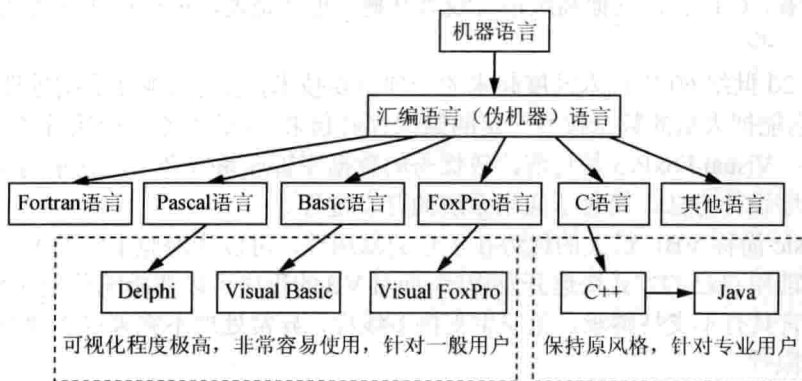


图 1-1 计算机语言种类

1. 机器语言

机器语言是最低级的语言，由二进制码组成，最早期的程序员通过在纸带上打点来写程序。

2. 汇编语言

汇编语言用助记符和地址符代替了二进制码，更易于编写。

3. 高级语言

高级语言更接近于自然语言,如 C 语言、Pascal、Java、C++、Visual FoxPro、FoxPro 等都是高级语言。其中可视化的高级语言采用面向对象程序设计,是语言发展的方向。

1.1.2 计算机高级语言种类和特点

1. 高级语言种类

计算机高级语言主要是相对于汇编语言而言,它并不是特指某一种具体的语言。高级语言种类繁多,如 BASIC (True basic、Qbasic、Visual Basic)、C、C++、Pascal、FORTRAN、FoxPro、Visual FoxPro、智能化语言 (LISP、Prolog、CLIPS、OpenCyc、Fazzy)、动态语言 (Python、PHP、Ruby、Lua) 等。这些语言的语法、命令格式都各不相同。

高级语言是绝大多数编程者的选择。和汇编语言相比,它不但将许多相关的机器指令合成为单条指令,并且去掉了与具体操作有关但与完成工作无关的细节,如使用堆栈、寄存器等,这样就大大简化了程序中的指令。由于省略了很多细节,所以编程者也不需要具备太多的专业知识。

2. 典型高级语言的个性特点

在 C 语言诞生以前,系统软件主要是用汇编语言编写的。但是汇编语言程序依赖于计算机硬件,其可读性和可移植性都很差;而一般的高级语言又难以实现对计算机硬件的直接操作。C 语言应运而生,兼有汇编语言和高级语言的特性,非常适合用于开发操作系统。C 语言的优点很多,如简洁紧凑、灵活方便、数据类型丰富、生成目标代码质量高、程序执行效率高等。缺点主要表现在数据的封装性上,这一点使得 C 语言在数据的安全性上有很大缺陷。从应用的角度看,C 语言比其他高级语言较难掌握。也就是说,对用 C 语言的人,要求对程序设计更熟练一些。

数据库是 20 世纪 60 年代末发展起来的一项重要技术,它的出现使数据处理进入了一个崭新的时代,它能把大量的数据按照一定的结构存储起来,在数据库管理系统的集中管理下,实现数据共享。Visual FoxPro 是目前比较优秀的数据库管理系统之一,它采用了可视化的、面向对象的程序设计方法,简化了应用系统的开发过程。

Visual Basic 简称 VB,最大的优势在于它的易用性,可以让经验丰富的 VB 程序员或刚刚入门的人都能用自己的方式快速开发程序。而且 VB 的程序可以非常简单地和数据库连接。但由于 VB 语言具有不支持继承、无原生支持多线程、异常处理不完善等 3 项明显缺点,使其应用有所局限性。

1.2 数据的存储及文件类型

1.2.1 数据的概念

1. 数据定义

在现实生活和工作中,人们通常使用各种各样的物理符号来表示客观事物的特性和特征,

这些符号及其组合就是数据。数据 (Date) 的概念包括两个方面, 即数据内容和数据形式。数据内容是指所描述客观事物的具体特性, 也就是通常所说数据的“值”; 数据形式则是指数据内容存储在媒体上的具体形式, 也就是通常所说数据的“类型”。例如, 某位老师的出生日期是“1985 年 8 月 22 日”, 基本工资是“1560”元等, 分别归结于日期型和数值型。

简言之, 数据指存储在某一媒体上能够被识别的物理符号。数据包括数字、文字、图形、图像、声音、动画、影像等多种表现形式。使用最多、最基本的主要有数字、文字。

2. 数据处理

数据处理就是将数据转换为有用信息的过程。数据处理的内容主要包括: 数据的收集、整理、存储、加工、分类、维护、排序、检索和传输等一系列活动的总和。数据处理的目的是从大量的数据中, 根据数据自身的规律及其相互联系, 通过分析、归纳、推理等科学方法, 利用计算机技术、数据库技术等技术手段, 提取有效的信息资源, 为进一步分析、管理、决策提供依据。数据处理也称信息处理。

例如, 学生各门成绩为原始数据, 经过计算得出平均成绩和总成绩等有意义的信息, 计算处理的过程就是数据处理。

在计算机中, 通过计算机软件来管理数据, 通过应用程序来对数据进行加工处理。用外存储器来存储数据。

1.2.2 计算机硬件存储设备

1. 计算机硬件

计算机硬件 (Hardware) 是软件系统赖以存在的物质基础, 是指有形的物理设备, 它是计算机系统中实际物理装置的总称。主要包括中央处理器、主存储器、辅助存储器、输入/输出设备、总线等 5 个部分。

中央处理器 (CPU): 主要由控制器、运算器等组成, 采用由大规模集成电路工艺制成的芯片, 用来对数据进行各种算术运算和逻辑运算, 是计算机的执行单元。

主存储器: 也称内存, 直接与 CPU 相连, 是计算机中的工作存储器, 计算机当前正在运行的程序与数据必须存放在内存内。其存取速度快, 但存储容量小。

辅助存储器: 也称外存, 存储容量大, 几乎存放计算机中所有的信息, 在计算机实际执行程序和处理数据时, 辅助存储器中的信息需要先传送入内存后才能被 CPU 使用。

输入/输出设备: 简称 I/O 设备, 是计算机与外界联系的桥梁。输入设备是指能向计算机系统输入信息的设备, 包括键盘、鼠标、扫描仪等。输出设备是指能从计算机系统输出信息的设备, 包括显示器、打印机、绘图仪等。

总线: 连接计算机中 CPU、内存、外存、各种输入/输出部件的一组物理信号线及其相关的控制电路, 是计算机中用于在各部件间运载信息的公共机构。

2. 外存储器

外存储器是计算机记忆或暂存数据的部件。计算机中的全部信息, 包括原始的输入数据、经过初步加工的中间数据及最后处理完成的有用信息都存放在外存储器中。而且, 指挥计算

运行的各种程序,即规定对输入数据如何进行加工处理的一系列指令也都存放在存储器中。

外存包括硬盘、光盘、U 盘、移动硬盘、存储卡等。

1.2.3 数据在计算机硬件上的存储

在计算机中数据是以文件的形式存储在存储介质上的。

1. 文件的定义

文件是用文件名来标识的一组相关信息的集合体。文件名格式为:主文件名.扩展名。其中扩展名一般为 3 个英文字母,如 AA.txt,毕业照 1.jpg, mm.gif 等。

主文件名命名规则:

- 1) 允许文件或者文件夹名称不得超过 255 个字符。
- 2) 文件名除了开头之外任何地方都可以使用空格。
- 3) 文件名中不能有下列符号:“?”、“、”、“\”、“*”、“”、“/”、“<”、“>”、“|”。
- 4) Windows 文件名不区分大小写,但在显示时可以保留大小写格式。
- 5) 文件名中可以包含多个间隔符,如“我的文件.我的图片.001”。

2. 文件类型列表

文件类型列表如表 1-1 所示。

表 1-1 文件类型

扩展名	文件类型	扩展名	文件类型
.dbf	表文件	.hlp	图形方式帮助文件
.cdx、.idx	符合索引文件、单索引文件	.pjx、.pjt	项目、项目备注文件
.fpt	表备注文件	.frx、.frit	报表文件、报表备注文件
.dbc	数据库文件	.lbx、.lbt	标签文件、标签备注文件
.dct、.dcx	数据库备注文件/索引文件	.mnx、.mnt	菜单文件、菜单备注文件
.mem	内存变量文件	.mpr、.mpx	菜单程序文件、编译后的菜单程序文件
.dll	Windows 动态链接库文件	.ocx	OLE 控件文件
.err	编译错误文件	.qpr、.qpx	查询程序文件、编译后的查询程序文件
.esl	VFP 支持的库文件	.scx、.sct	表单文件、表单备注文件
.fil	FoxPro 动态链接库文件	.spr、.spx	表单程序文件、编译后的表单程序文件
.prg	解释执行的程序文件	.txt	文本文件
.fxp	编译后的程序文件	.vcx、.vct	可视类库文件、可视类库备注文件
.app	生成的应用程序文件	.vue	FoxPro 2.X 视图文件
.exe	可执行程序文件	.h	头文件 (VFP 或 C/C++ 程序需要包含的)

1.3 数据库基础

随着计算机技术的发展,计算机的主要应用已从科学计算转变为事务数据处理。在事务处理过程中,并不需要进行复杂的科学计算,而是要进行大量数据的存储、查找、统计等工

作,如教学管理、人事管理、财务管理等,就需要对大量数据进行管理,而数据库技术就是目前最先进的数据管理技术。数据库技术主要研究在计算机环境下如何合理组织数据、有效管理数据和高效处理数据。

简言之,数据处理的核心问题是数据管理。数据管理技术经历了3个阶段:人工管理、文件系统、数据库管理。数据库管理主要解决了数据冗余度、数据独立性、数据一致性问题。

1.3.1 数据库基本概念

数据库系统(DataBase System, DBS)由计算机硬件、数据库管理系统、数据库、应用程序和用户等部分组成。涉及以下几个概念。

1. 数据库

数据库(DataBase, DB),指以一定的组织方式存储在计算机存储设备上,且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。它不仅包括描述事物的数据本身,而且包括相关事物之间的联系。

2. 数据库管理系统

数据库管理系统(DataBase Management System, DBMS),是为数据库的建立、使用和维护而配置的软件,是数据库系统的核心组成部分。

3. 数据库系统

数据库系统(DataBase System, DBS),指引进数据库技术后的计算机系统。

4. 数据库应用程序

数据库应用程序(Application)指系统开发人员利用数据库系统资源开发出来的,面向某一类信息处理问题而建立的软件系统。

5. 数据库用户

数据库用户(User)是指管理、开发、使用数据库系统的所有人员,通常包括数据库管理员、应用程序员和终端用户。

1.3.2 数据模型

计算机信息处理的对象是现实生活中的客观事物,在对客观事物实施处理的过程中,首先要经历了解、熟悉的过程,从观测中抽象出大量描述客观事物的信息,再对这些信息进行整理、分类和规范,进而将规范化的信息数据化,最终由数据库系统存储、处理。在这一过程中,涉及不同的层次,经历了抽象和转换。

数据模型按不同的应用层次分成3种类型,它们是概念数据模型、逻辑数据模型、物理数据模型。概念数据模型又称E-R模型、逻辑数据模型又称数据模型,分为层次模型、网状模型、关系模型,物理数据模型又称物理模型。

1. 概念模型

概念模型又称实体联系模型 (E-R 模型), 是反映实体之间联系的模型。数据库设计的重要任务就是建立实体模型, 建立概念数据库的具体描述。在建立实体模型时, 实体要逐一命名以示区别, 并描述它们之间的各种联系。实体模型只是将现实世界的客观对象抽象为某种信息结构, 这种信息结构并不依赖于具体的计算机系统, 所以一般我们用 E-R 图即实体联系图 (Entity Relationship Diagram) 表示。

(1) E-R 模型的基本概念

1) 实体。客观事物在信息世界中称为实体 (Entity), 它是现实世界中任何可区分、识别的事物。实体可以是具体的人或物, 也可以是抽象概念。

2) 属性。实体具有许多特性, 实体所具有的特性称为属性 (Attribute)。一个实体可用若干属性来刻画。每个属性都有特定的取值范围即值域 (Domain), 值域的类型可以是整数型、实数型、字符型等。

3) 实体集。性质相同的同类实体的集合称为实体集 (Entity set), 如一个班的学生。

4) 实体联系。建立实体模型的一个主要任务就是要确定实体之间的联系。常见的实体联系有 3 种: 一对一联系、一对多联系和多对多联系, 如图 1-2 所示。

① 一对一联系 (1:1)。若两个不同型实体集中, 任一方的一个实体只与另一方的一个实体相对应, 称这种联系为一对一联系, 如学号与学生的联系, 一个学号只对应一个学生, 一个学生也只有一个学号 [图 1-2 (a)]。

② 一对多联系 (1:n)。若在两个不同型实体集中, 一方的一个实体对应另一方若干个实体, 而另一方的一个实体只对应本方一个实体, 称这种联系为一对多联系, 如班主任与学生的联系, 一个班主任对应多个学生, 而本班每个学生只对应一个班主任 [图 1-2 (b)]。

③ 多对多联系 (m:n)。若两个不同型实体集中, 两实体集中任一实体均与另一实体集中若干个实体对应, 称这种联系为多对多联系, 如教师与学生的联系, 一位教师为多个学生授课, 每个学生也有多位任课教师 [图 1-2 (c)]。

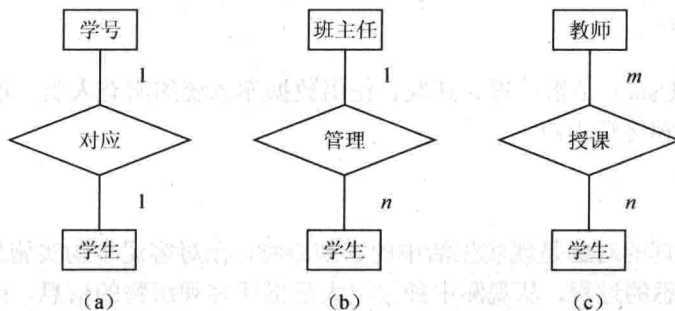


图 1-2 E-R 图

(2) E-R 模型的图示法

通常情况下, 我们用图形表示 E-R 模型。

1) 矩形表示实体集。

2) 菱形表示实体联系。

3) 椭圆形表示属性。

4) 直线表示实体集与属性间的联结关系。

例如, 商场各实体属性如下:

商场: 商场号、名称、地址、规模。

经理: 姓名、性别、电话、地址、商场号。

商品: 商品号、名称、类别、价格、生产日期、库存量、商场号、顾客编号。

顾客: 顾客编号、姓名、会员否、性别、年龄。

可以绘制出 E-R 图, 如图 1-3 所示。

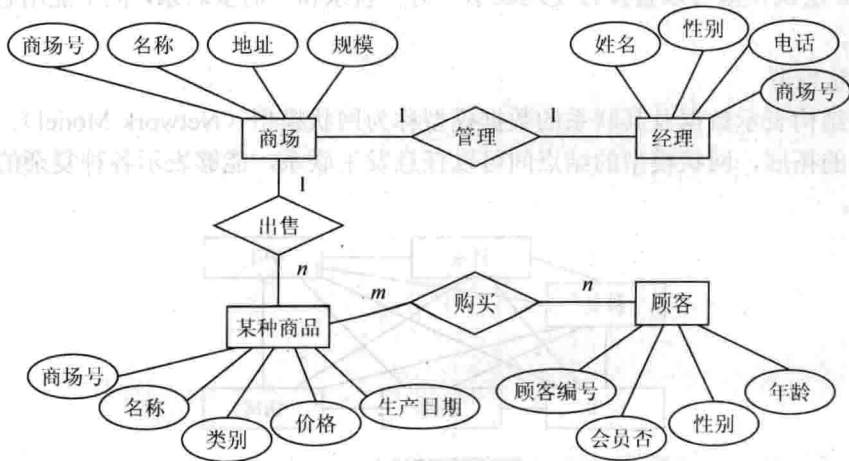


图 1-3 商场 E-R 图

2. 逻辑数据模型

逻辑数据模型通常称为数据模型, 是指数据库中数据与数据之间的关系。

数据模型是数据库系统中一个关键概念, 数据模型不同, 相应的数据库系统就完全不同, 任何一个数据库管理系统都是基于某种数据模型的。数据库管理系统常用的数据模型有下列三种: 层次模型、网状模型、关系模型。

(1) 层次模型

用树形结构表示数据及其联系的数据模型称为层次模型 (Hierarchical Model), 如图 1-4 所示。

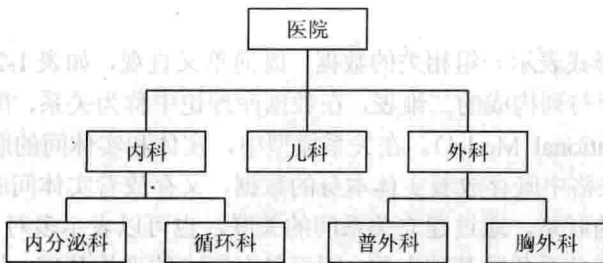


图 1-4 层次模型

树由结点和连线组成, 结点表示数据集, 连线表示数据之间的联系, 树形结构只能表示一对多联系。通常将表示“一”的数据放在上方, 称为父结点; 而表示“多”的数据放在下