

教育部考试中心制订的考试大纲配套用书

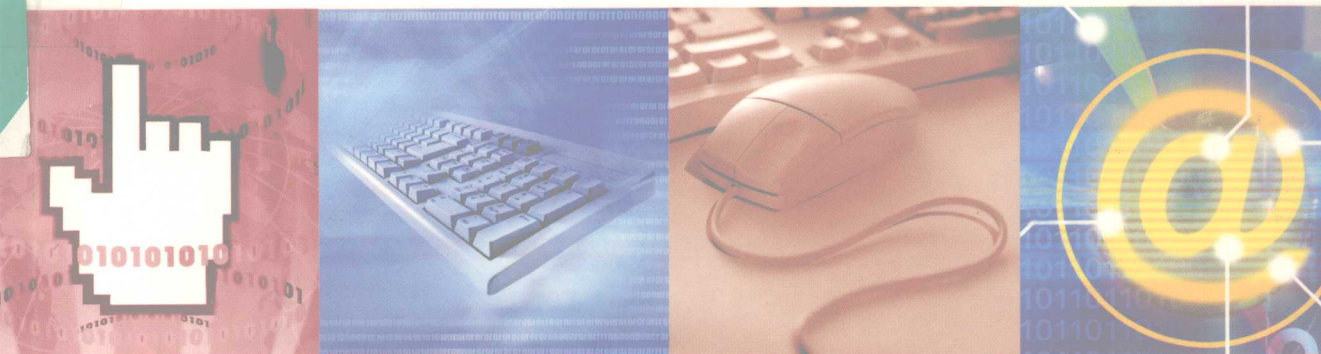


全国计算机
等级考试用书

二级教程

—Visual FoxPro程序设计

全国计算机等级考试用书编委会 组编



世界知识出版社

内容提要

本书是依据 2002 年版《全国计算机等级考试二级考试大纲 (Visual FoxPro 程序设计)》编写的,讲解简明扼要,理论联系实际。在每一章均有丰富而且实用性很强的例题和同步训练,同步训练附有解题分析。全书共分十二章。本书主要内容包括:Visual FoxPro 6.0 概述、数据与数据运算、数据库、查询与视图、SQL 使用、程序设计基础、表单、菜单与工具栏、报表设计、应用程序发布等。

本书可以作为全国计算机等级考试二级 Visual FoxPro 培训教材,同时也可以作为其他人员学习 Visual FoxPro 6.0 的教材和参考用书。

图书在版编目 (CIP) 数据

二级教程. Visual FoxPro 程序设计 / 于忠诚等编著.

北京:世界知识出版社, 2003

全国计算机等级考试用书

ISBN 7-5012-2029-8

I. 二... II. 于... III. ①电子计算机—水平考试—教材②关系数据库—数据库管理系统, Visual FoxPro—水平考试—教材 IV. TP3

中国版本图书馆 CIP 数据核字 (2003) 第 029565 号

责任编辑 / 郑志国

封面设计 / 易 红

责任出版 / 夏凤仙

出版发行 / 世界知识出版社

地址电话 / 北京东城区干面胡同 51 号

邮政编码 / 100010

经 销 / 新华书店

开本印张 / 787 × 1092 毫米 1/16 开本 17 印张 390 千字

印 刷 / 北京大中印刷厂

版 次 / 2003 年 6 月第 1 版 2003 年 6 月第 1 次印刷

印 数 / 1-5000 册

定 价 / 25.00 元

版权所有 翻印必究

前 言

Visual FoxPro 6.0 是由美国 Microsoft 推出的程序开发工具软件,主要用于小型数据库系统的开发。它具有功能强大的面向对象程序设计工具以及客户/服务器能力。该系统使得在程序开发过程中组织数据、定义数据库规则和建立应用程序等工作变得简单易行。

本书以 Visual FoxPro 6.0 为语言背景,通过大量的实例,深入浅出地介绍了 Visual FoxPro 的基础知识、Visual FoxPro 的编程工具及编程步骤。在介绍面向对象编程方法的同时,也介绍了面向过程的设计方法,这对于没有数据库设计经验的人来说,无疑是很有帮助的。

本书每章后面均附有同步训练,并对所有的题目既给出了参考答案又作了详细的解题分析,这对于快速、深入地理解 Visual FoxPro 的理论内容是非常有效的。

全书共分十二章,第一章至第四章由于忠诚编写;第五章由齐文斌编写;第六章至第七章由王世明编写;第八章至第十二章由赵逢达编写。

本书除作为全国计算机等级考试用书外,还可用作大学、高职及各类中等专业学校的教材。

由于时间仓促,水平有限,书中难免有不足之处,恳请广大读者批评指正。

编 者
2003.5

目 录

第一章 Visual FoxPro 基础	1
1.1 数据库基础	1
1.1.1 数据库系统及其特点与发展历史	1
1.1.2 数据模型	2
1.1.3 关系运算	4
1.1.4 数据库管理系统	4
1.2 数据库设计与保护	5
1.2.1 数据库的设计原则	5
1.2.2 数据库设计过程	5
1.2.3 数据库的保护	6
1.3 Visual FoxPro 简介	7
1.3.1 Visual FoxPro 的版本	7
1.3.2 Visual FoxPro 6.0 的特点	7
1.3.3 VFP 中的数据类型	8
1.3.4 Visual FoxPro 中使用的文件类型	9
1.3.5 VFP 的工作方式和命令格式	9
同步训练	10
参考答案及解题分析	11
第二章 Visual FoxPro 系统的编程工具介绍	13
2.1 Visual FoxPro 系统的主界面介绍	13
2.2 项目管理器	14
2.2.1 创建和打开项目	14
2.2.2 项目管理器的操作	16
2.3 设计器简介	17
2.4 工具栏简介	17
2.5 生成器简介	18
2.6 向导简介	18
2.7 表单设计器	19
2.8 Visual FoxPro 编程步骤	20
2.8.1 添加控件	20
2.8.2 由控件创建对象的方法	21
2.8.3 修改属性	21
2.8.4 编写代码	22
2.8.5 运行表单	22
2.8.6 修改表单	22
同步训练	22
参考答案及解题分析	23

第三章 Visual FoxPro 数据与数据运算	25
3.1 常量	25
3.2 变量	26
3.2.1 变量的概念	26
3.2.2 内存变量	26
3.2.3 字段变量	29
3.2.4 数组	29
3.3 表达式	29
3.3.1 算术运算符及表达式	30
3.3.2 字符表达式	30
3.3.3 日期时间表达式	30
3.3.4 关系表达式	31
3.3.5 逻辑运算符及逻辑表达式	32
3.4 常用函数	33
3.4.1 函数的分类	33
3.4.2 常用函数举例	35
同步训练	39
参考答案及解题分析	40
第四章 Visual FoxPro 数据库及其操作	42
4.1 创建数据库与表	42
4.1.1 数据库和表的基本概念	42
4.1.2 创建数据库	42
4.1.3 数据库的使用及操作	43
4.2 表的设计与操作	45
4.2.1 创建表	45
4.2.2 对表结构的操作	48
4.3 对表记录的操作	48
4.3.1 对表记录操作的方法和命令	48
4.4 对表的高级操作	51
4.4.1 索引	51
4.5 自由表	54
4.5.1 自由表的建立	54
4.5.2 将自由表添加到数据库	54
4.5.3 从数据库中移出表	55
4.6 多个表的同时使用	56
4.6.1 多工作区的概念	56
4.6.2 使用不同工作区的表	57
4.6.3 表之间的关联	57
4.7 排序	57
同步训练	58
参考答案及解题分析	60

第五章 查询与视图	61
5.1 查询	61
5.1.1 查询设计器	61
5.1.2 查询的建立	62
5.1.3 查询设计器的局限性	64
5.1.4 运行查询	64
5.2 视图	66
5.2.1 视图的概念	66
5.2.2 创建视图	67
5.2.3 使用视图	71
5.2.4 使用视图更新数据	72
同步训练	73
参考答案及解题分析	75
第六章 关系数据库标准语言 SQL	79
6.1 SQL 概述	79
6.2 SQL 查询功能	80
6.2.1 简单查询	83
6.2.2 嵌套查询	85
6.2.3 联接查询	87
6.2.4 计算查询	91
6.2.5 集合的并运算	93
6.3 SQL 的数据更改功能	94
6.3.1 删除	94
6.3.2 插入	94
6.3.3 更新	95
6.4 SQL 的数据定义功能	95
6.4.1 表的定义	95
6.4.2 表结构的修改	97
6.4.3 表的删除	98
同步训练	98
参考答案及解题分析	100
第七章 程序设计基础	102
7.1 程序与程序文件	102
7.1.1 程序的概念	102
7.1.2 程序文件的建立与执行	102
7.2 顺序结构	106
7.2.1 基本语句	106
7.2.2 基本输入输出命令	110
7.2.3 顺序结构	114
7.3 选择结构	115
7.3.1 单分支选择语句 (IF...ENDIF)	115

7.3.2	双分支选择语句 (IF...ELSE...ENDIF)	115
7.3.3	多向分支选择语句 (DO CASE...ENDCASE)	117
7.3.4	分支结构的嵌套	119
7.4	循环结构	121
7.4.1	当型循环语句 (DO WHILE...ENDDO)	122
7.4.2	步长型循环语句 (FOR... ENDFOR)	124
7.4.3	基于数据库的循环 (SCAN...ENDSCAN)	126
7.4.4	编程实例	127
7.5	子程序设计及调用	131
7.5.1	子程序的作用	131
7.5.2	子程序的调用	131
7.6	自定义过程和函数	131
7.6.1	自定义过程	132
7.6.2	自定义函数	134
7.6.3	参数传递	135
7.6.4	变量的作用域	136
7.7	程序设计中常见的语法错误	138
7.8	程序调试	139
7.8.1	调试方法	139
7.8.2	设置断点	139
7.8.3	调试菜单	140
	同步训练	141
	参考答案及解题分析	145
第八章	利用表单设计应用程序	147
8.1	面向对象的概念	147
8.1.1	对象与类	147
8.1.2	深入了解 Visual FoxPro 中的类	148
8.2	Visual FoxPro 基类简介	148
8.2.1	Visual FoxPro 基类	148
8.2.3	对象的引用	149
8.3	创建与管理表单	151
8.3.1	使用表单向导创建表单	151
8.3.2	使用表单设计器创建表单	151
8.4	表单设计器	153
8.4.1	表单设计器环境	153
8.4.2	控件的操作与布局	157
8.5	管理表单和常用表单控件的属性和方法	161
8.5.1	管理表单属性和方法	161
8.5.2	标签 (Label) 控件 	165
8.5.3	文本框 (TextBox) 控件 	166
8.5.4	命令按钮 (CommandButton) 控件 	169

8.5.5	命令组 (CommandGroup) 控件 	170
8.5.6	编辑框 (EditBox) 控件 	172
8.5.7	复选框 (CheckBox) 控件 	172
8.5.8	选项按钮组 (OptionButton) 控件 	173
8.5.9	列表框 (ListBox) 控件  和组合框 (ComboBox) 控件 	175
8.5.10	表格 (Grid) 控件 	178
8.5.11	页框 (PageFrame) 控件 	179
8.6	数据环境	179
8.6.1	什么是数据环境	179
8.6.2	打开“数据环境设计器”	180
8.6.3	向数据环境设计器添加表或视图	180
8.6.4	从数据环境中移去表或视图	181
8.6.5	在数据环境设计器中设置关系	181
8.6.6	在数据环境设计器中编辑关系	181
8.6.7	常用数据环境属性	181
8.6.8	为控件设置控制源	182
	同步训练	184
	参考答案及解题分析	186
第九章	使用菜单	190
9.1	设计菜单	190
9.1.1	规划菜单系统	190
9.1.2	“菜单设计器”简介	191
9.1.3	主菜单中的有关选项	194
9.1.4	在顶层表单中添加菜单	196
9.2	设计快捷菜单	198
9.2.1	快捷菜单与下拉式菜单的比较	199
9.2.2	建立快捷菜单	199
	同步训练	200
	参考答案及解题分析	201
第十章	报表设计	202
10.1	创建报表	202
10.1.1	报表的常规布局	202
10.1.2	使用报表向导创建报表	203
10.1.3	在报表设计器中创建空白报表	206
10.1.4	创建快速报表	206
10.2	修改报表布局	208
10.3	报表数据源和报表控件	209
10.3.1	设置数据源	210
10.3.2	标签控件	210
10.3.3	域控件	211

10.3.4	线条、矩形、圆角矩形	214
10.3.5	图片/ActiveX 绑定控件	214
10.3.6	设计报表控件	216
10.4	数据分组和多栏报表	217
10.4.1	数据分组	217
10.4.2	单级数据分组报表	217
10.4.3	多级数据分组报表	218
10.4.4	多栏报表	219
10.5	报表的输出	219
10.5.1	定义报表页面	220
10.5.2	预览结果	220
10.5.3	打印报表	221
10.6	举例	222
	同步训练	224
	参考答案及解题分析	224
第十一章	开发应用程序	226
11.1	应用项目综合实践	226
11.1.1	系统开发基本步骤	226
11.1.2	应用程序的连编	229
11.1.3	主程序设计	232
11.2	应用程序向导和应用程序生成器	234
11.2.1	使用应用程序向导	234
11.2.2	应用程序生成器	236
	同步训练	240
	参考答案及解题分析	241
第十二章	上机操作部分	243
12.1	考试环境	243
12.1.1	硬件环境	243
12.1.2	软件环境	243
12.2	考试系统	244
12.2.1	登录	244
12.2.2	考试界面	246
12.2.3	答题	246
12.2.4	交卷	246
12.3	考试内容	247
12.4	上机考试模拟题及简单解答	247
12.4.1	模拟题一	247
12.4.2	模拟题二	249
	模拟试卷	252
	模拟试卷参考答案	257
	二级考试大纲(Visual FoxPro 程序设计)	258

第一章 Visual FoxPro 基础

1.1 数据库基础

在数据库技术没有发展起来以前,人们常常用某种计算机高级语言作数据处理,这种方法不但执行速度慢,数据冗余大,而且使得程序设计和修改非常复杂。20 世纪 60 年代末期数据库技术兴起,使得这一切都有所改变。它是一种数据处理的新的方法,也是信息管理中一项非常重要的新技术。由于数据库具有数据结构化,数据冗余度低,程序与数据独立性高和易于扩充、易于程序设计等优点,因此,得到了迅速发展和广泛应用,使计算机在应用方面又开辟了一个新的领域。

1.1.1 数据库系统及其特点与发展历史

1. 基本概念

数据与信息是密切相关的,所谓信息 (Information) 是客观事物的特征在人脑中的反映。数据 (Data) 是符号化了的信息。所谓数据库是按照一定的组织方式,相互有关的数据的集合。数据库系统 (DataBase System DBS) 是以数据库应用为基础的计算机系统。

2. 数据库的特点

数据库可具有以下几个特点:

(1) 数据独立

应用程序不需要了解数据实际的存取方式,通过数据库的存取指令,就可得到所需数据。因此,当数据的存取结构变更时,仅需更改数据库系统的内部程序,外部的应用程序完全不需要改变。

(2) 数据共享

由数据库的概念可知,数据库中的数据是供所有的程序使用的,因此,数据库具有共享特点是不言而喻的。

(3) 减少冗余

由于非数据库系统中,每个应用方法使用自己的数据来处理,经常会造成数据的重复建立,而且彼此之间的数据格式也不相同,无法交互应用。在数据库系统中,仅建立共用的数据库,其余的应用程序都使用该数据库,因此,可大大减少数据的冗余。

(4) 统一控制

数据库中的数据是各用户的共享资源,即许多用户同时使用数据库,会产生并发操作,因此,数据库系统提供了三方面的数据控制功能:

- 安全性控制:防止数据被破坏和泄密设置安全措施,只让合法用户进行有限的操作。
- 完整性控制:保证数据库中的数据在输入、修改过程中始终符合原来的定义和规定。

保证数据的正确、有效、相容。

● 并发控制:当多用户并发进程同时存取、修改数据库时,可能会发生相互干扰使数据库的完整性遭到破坏,因此,数据库系统提供了对并发操作的控制功能。

3. 数据库技术的发展

随着计算机硬件和软件的发展,数据管理技术也得到了迅速的发展,大致经历了三个发展

阶段,即人工管理、文件系统、数据库系统这三个阶段。60 年代前属于人工管理阶段;60 年代末属于文件系统管理阶段;70 年代后期,随着微机进入市场,DBASE 被开发并进入了微机世界,成为相当受欢迎的数据库管理系统。它在 80 年代进入我国。随着 PC 机的广泛使用,DBASE 的产品也在不断地升级,由最开始的 DBASE II,DBASE III,最后到 DBASE IV。它的诞生和发展使数据的规范化、独立化、共享性都得到了进一步的改善。极大地推动了计算机信息管理学科的进步。

80 年代初由 Fox SoftWare 公司在 PC 机上开发出了一种新型的数据库管理系统(DBMS) FoxBASE,于 84 年正式投放市场。该产品为 DOS 操作系统平台下完成计算机数据处理发挥了巨大的作用。

90 年代,由于操作系统平台的更换,FoxBASE 已经不能适应在 Windows 操作系统环境下运行,因此,又开发出了运行于新平台的数据库管理系统 FoxPro,该产品的标准版本不但增加了 100 多条全新的命令与函数,其它方面的功能,像 SQL 结构化设计语言和直观的案例关系查询、窗口及菜单操作等都是前所未有的。

1992 年 MicroSoft 公司收购了 Fox 公司,因此也将 FoxPro 纳入了自己的产品系列中,在此基础上该公司于 1995 年推出了 Visual FoxPro 5.0 及其中文版本,1998 年发布了可视化编程语言集成包 Visual FoxPro 6.0 版本。

1.1.2 数据模型

数据模型即数据库结构,可分为三种:层次型(Hierachical DataBase)、网状型(Network)、关系型(Relation)。

1. 层次型数据库

其组织如图 1-1 所示。

层次型数据库如树状结构,具有父子关系,每个父节点(图 1-1 中的 A、B 层)可以有很多子节点,但每个子节点(B、C、D、E、F)仅可以有一个父节点。若需要子节点有很多父节点或不同的父节点的子节点间要联系,则无法使用此模式,必须改用其它模式。

以一个国家系统来说,最高层为中央,下面管理许多省,省下面又管理许多县,县下面还管理许多区,等等,从而形成一个庞大的层次型数据库结构。

因此,层次型数据库结构的优点是:结构简单,易于操作;从上而下寻找数据容易;与日常生活的数据类型相似。

其缺点是:寻找非直系的节点非常麻烦,必须通过多个父节点由下而上,再向下寻找,搜寻的效率太低。

2. 网状型数据库

其组织形式如图 1-2 所示。

网状型结构比层次型数据库更有弹性,允许子节点有多个父节点,例如 F。

子节点间的关系比较接近,联系容易。但缺点是路径太多,当添加或删除数据时,牵动的相关数据很多,重建和维护数据比较麻烦。

此结构使用于稳定的数据库,当数据变动较小时,

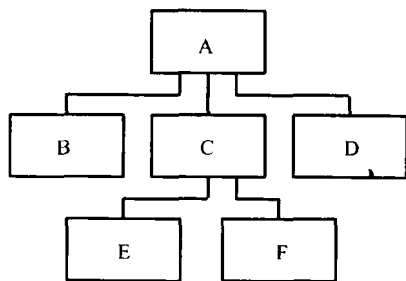


图 1-1 层次型数据库

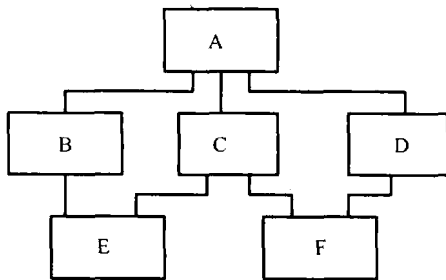


图 1-2 网状型数据库

此方法可以很容易地与其它的数据联系。其缺点是不适用于动态数据库,因为动态数据库中的数据常常需要变更,会造成重建的困难。

3. 关系型数据库

关系模型中仅存在唯一的结构类型——关系,所谓关系型数据库就是用关系模型组织的数据库结构。我们可将关系型数据库结构看成是一个二维表,这种二维表就是关系,无论是实体本身还是实体之间的联系均用二维表表示,使得实体中的数据本身能自然地反映它们之间的联系。例如:有学生情况、学生成绩和班级名称三种数据的集合。

构成学生情况关系数据模型为:

学生情况(学号,姓名,年龄,班级,...)

如表 1-1 所示:

表 1-1 学生情况表

学 号	班 级	姓 名	性 别	出 生	团 员	籍 贯	备 注 1
200001	通 讯	吕小雅	女	11-02-79	.T.	浙江	memo
200002	计算机	于海涛	女	09-12-80	.F.	江西	mcmo
200003	机 械	齐圆圆	女	09-25-80	.T.	河北	memo
200004	光电子	张富知	女	11-11-79	.T.	四川	memo
200010	测 控	李 洋	女	09-10-80	.F.	浙江	memo
200030	经 管	史 竹	女	12-20-80	.T.	河北	memo

构成学生成绩关系数据模型为:

学生成绩(学号,数学,英语,计算机,...)

如表 1-2 所示:

表 1-2 学生成绩表

学 号	班 级	英 语	数 学	计算机
200001	通 讯	88	79	78
200002	计算机	87	89	88
200003	机 械	78	76	67
200004	光电子	90	88	69
200010	测 控	69	96	93
200030	经 管	86	87	82

构成课程关系数据模型为:

组织机构(编号,学院,专业,班级,...)

如表 1-3 所示:

表 1-3 组织机构表

编 号	学 院	专 业	班 级
130601	机械工程	轧 钢	200201
130602	信息工程	计算机	200202
130603	经济管理	经 贸	200203
130604	电器工程	测 控	200204

在关系型数据库系统中,表中的一列称为字段,一行称为一条记录,一条记录由多个字段组成,多条记录的集合则构成一个数据库表文件。

关系型数据库结构主要的优点是:表格的关系比较容易建立数据库;用户容易理解表格

式中的关系;表格式的结构便于修改和查询。

基于以上这些优点,关系型数据库在计算机中得到了广泛的应用,尤其在 PC 机中用得特别多。

1.1.3 关系运算

在对关系型数据库中的数据进行操作和处理时,往往要找到用户所需要的数据,此时就要对关系进行一定的关系运算。常用的关系基本运算有三种:

1. 选择

所谓选择是指在关系中找出满足给定条件的数据操作。选择的条件以逻辑表达式给出,使逻辑表达式条件为真的数据被选取。例如,在学生成绩表中要查找出英语成绩在 80 分以上、计算机成绩在 70 分以上学生的操作,就是选择。

选择是从行的角度进行运算,在水平方向选取记录。

2. 投影

所谓投影是指在关系中选取某些属性列组成新的关系,相当于对关系进行垂直分解,在新关系中包含的属性个数比原关系少。例如,在机构组成表中要查找出涉及学院和专业的数据的操作就是投影。

3. 联接

所谓联接运算是将两个关系模式拼接成一个更宽的关系模式,新关系中包含满足条件的元组。两个关系模式的联接是通过联接条件实现的,联接的结果是满足条件的所有记录。例如,要在学生情况表和学生成绩表中查询英语和计算机成绩都大于 80 分的学生。由于学生情况表中没有学生成绩,而学生成绩表中没有学生姓名,因此就需要将两个表联接起来,联接的条件应是这两个表中共有的学号,然后找出满足条件的记录。

1.1.4 数据库管理系统

数据库管理系统 DBMS(DataBase Management System)是使数据库语言能正常运行的支持软件。正如使用高级语言需要解释或编译程序一样,使用数据库语言也需要有特定的环境和软件的支持。数据库管理系统一般应具有下列功能:

(1) 数据定义功能

DBMS 应向用户提供“数据定义语言”DDL(Data Definition Language),用于定义、建立、修改或删除数据库结构等。

(2) 控制和管理功能

DBMS 应具有必要的控制和管理功能,其中包括:在多用户使用中对数据的并发控制、备份、恢复和转储等。

(3) 对数据库的操作功能

对数据库进行查询和检索,是数据库的主要应用。所以,DBMS 应提供“数据操作语言”DML(Data Manipulation Language),支持用户对数据库的数据进行查询、和编辑(增、删、改)等操作。例如,关系数据库中的 SQL(结构化查询语言)就具有查询语句:

```
select  字段名  
from    库表名  
where   查询条件
```

目前,在 PC 机上使用的关系数据库管理系统都具有 DDL、DML 和 DCL(数据控制语言)为一体的各种功能。

1.2 数据库设计与保护

1.2.1 数据库的设计原则

数据库中的数据是供给用户进行各种操作的原始和真实的数据,因此,合理地组织数据库能为程序设计带来迅速、高效、准确的访问打下良好的基础。为了能合理地组织数据,应遵从以下基本设计原则:

1. 多数据库,少数据量

如果所组织的数据库记录非常大,程序对数据库操作时所用的时间就特别长。此时,宁肯根据需求和数据库中的数据类型将该数据库拆分成多个数据库,从而减少每个数据库的记录。这是为了程序执行时能快速访问数据,有利于查询、统计的一种设计方法。

2. 尽量减少各表间的重复字段

除了保证各表之间存在联系的外部关键字段外,尽量减少各表之间的重复字段,其目的是尽量减少数据的冗余,防止在插入、删除和更新时造成数据的不一致。

3. 表中必须保持正确和原始的基本数据

程序利用数据库中的数据进行各种各样的操作必然会产生中间或最终的结果,但保证数据库中数据的正确性和原始性是非常重要的。若需要时,可将中间结果或最终结果保存在另外的数据库中,以保证其它程序操作的正确性。

4. 用外部关键字保证有关联的表之间的联系

各表间的关联是依靠外部关键字来实现的,因此,在尽量减少重复字段的情况下,设计时还要考虑关联时所用到的相同的关键字段,使得表具有合理的结构。

1.2.2 数据库设计过程

1. 需求分析

根据用户提出的要求、要完成的任务和要达到的各项指标,全面汇总、详细分析数据资料对数据库的设计十分重要。在以上的工作中总结出用户要从数据库所获得的信息内容,定义数据库结构、数据类型以及处理数据的方法等。

2. 确定需要的表

根据用户对数据库的要求,查询、统计、报表等功能所需数据,将信息分门别类地建立各种各样的表,以满足各种功能的需要。

3. 设计表中的字段

表中的字段是为程序执行某种操作而设置的,字段中的数据就是程序操作的真正对象,因此,对表的字段设计时主要考虑的是正确、全面、原始。即在设计表时必须保证表中的字段和数据适用、完整;同时保证程序执行时表能提供全面的、原始和真实的数据。

4. 确定联系

根据要求所设计的数据库,不仅存储了所需实体信息,同时还要反映出实体之间客观存在的关联。确定联系的方法如下:

(1) 一对一联系

如果有两个相同实体的表,可使用两表中相同的关键字作为主关键字建立一对一的联系。例如,学生情况表和学生成绩表就可以用学号作为主关键字,建立联系。如果两表中没有相同的实体及相同的关键字,可选择其中的一个表,将它的关键字段放到另一个表中作为外部关键字段,以此建立一对一关系。例如,学生成绩表和组织结构表中可以将组织结构表的编号和学

生成成绩表中的学号放到学生情况表中。

(2) 一对多联系

一对多联系是数据库中非常普遍的联系,在一对多联系中,假定表 A 中有一条记录在表 B 中可以有多条记录与之对应,但 B 中的一条记录只能有一个表 A 的记录与之对应。例如,学生情况表与学生成绩表就存在一对多的联系,即一个学生对应有多个课程的成绩。建立这种联系的原则是将“一方”的主关键字字段添加到“多方”的表中。

(3) 多对多联系

在多对多联系中,表 A 的一条记录在表 B 中有多条对应的记录,而表 B 的一条记录在表 A 中也可以有多条记录对应,这就是多对多联系。但这种联系容易出现数据的重复存储,为了避免数据的重复存储,在 Visual FoxPro 中常常采用建立第三个表的方法解决,即把多对多联系分解成两个一对多联系,所创建的第三表中包含两个表的主关键字。

1.2.3 数据库的保护

数据库中的数据是按着某种方式组织在一起并为用户共享的数据,因此,数据库的保护是数据库设计过程中非常重要的一部分。数据库的保护应主要分为安全性、完整性、并发控制及数据库的恢复四个方面。

1. 安全性

数据库的安全性是指由于不合法的使用所造成的数据泄露、更改和破坏。在此方面,可有四级安全措施对数据库加以保护。

(1) 硬件级

该级主要是为防止透漏口令、线路窃听以及盗窃设备而设置安全性措施。一般可将数据编为密码,设置口令、硬件加锁、设置防盗等措施。

(2) 操作系统及网络级

操作系统及网络软件都提供了一些安全性的保护措施,它包括用户的分组、用户权限、文件使用权限、目录权限等设置。

(3) DBMS 级

数据库管理系统(DBMS)一般提供两层保护,一是进入系统之前判定用户标识与身份鉴定,看其是否为合法用户;二是进入系统后的存取控制,保证用户只能存取有操作权的数据。

(4) 应用程序级

根据用户的实际需要在应用程序中设置口令,以分配不同用户的使用权限。

2. 完整性

数据的完整性是指数据的正确性和一致性,为能保证这一点,要求 DBMS 必须提供相应的完整性约束条件的检查功能,以防止错误的或不合法的约束条件的数据进出数据库。

3. 并发控制

由于数据库中数据是共享的,因此,允许多个用户在同一时刻存取数据,但是,对多个用户在同一时间存取数据不加控制,就很容易破坏数据的完整性和一致性,产生错误的结果。所以在 DBMS 中都提供了并发控制功能。在多用户环境下,DBMS 一般都采用独占或封锁的策略解决该问题。

4. 数据库恢复

尽管系统采用了各种各样的安全措施以防止数据被破坏,但计算机系统中硬件故障、软件错误、操作失误、病毒侵袭等仍是不可避免的。这些错误轻则影响程序的正常执行,重则会造成整个数据库的破坏。因此,DBMS 一般要有从错误状态恢复到已知的正确状态的功能。这

就是数据库恢复。目前数据库的恢复一般都采用以下几种方法:

(1) 转储法

即系统定期地自动地将数据库备份下来,这是最常用的基本技术。

(2) 使用记录文件

系统随时将用户的各种操作收录到一个文件中,利用它来恢复整个数据库。

(3) 采用硬件结构和技术

例如采用磁盘镜像、双机容错等技术使得一个硬件设备出现问题后,另一个硬件设备的相同数据库仍可投入使用。

1.3 Visual FoxPro 简介

1.3.1 Visual FoxPro 的版本

Visual FoxPro 是继 DBASE、FoxBASE、FoxPro 之后的又一微型计算机中关系数据库管理系统的新产品。它与 DBASE、FoxBASE、FoxPro 是兼容的。

从 1995 年以来,微软公司陆续发布了三个版本的 Visual FoxPro,即 Visual FoxPro 3.0、Visual FoxPro 5.0、Visual FoxPro 6.0 及其中文版本。其中 Visual FoxPro 6.0 (中文版)是 Microsoft 公司 1998 年发布的可视化编程工具之一。它可运行于 Windows 95/98、Windows NT 平台的 32 位数据库开发系统,能充分发挥 32 位微处理机的强大功能。它采用了面向对象的可视化程序设计方法,大大地简化了应用程序开发过程,提高了系统的模块性和整体性。

1.3.2 Visual FoxPro 6.0 的特点

1. 特点概述

Visual FoxPro 6.0 与 FoxPro 相比,最大的区别是前者是真正面向对象的程序设计方法,它有多种可视化编程工具,重复使用各种类,可以直观地、创造性地建立应用程序;在表的设计方面,增添了表的字段和控件直接结合的设置,使用户能更快、更容易地设计和修改应用程序界面。而后者还基本上属于面向问题的过程设计或是结构化设计的方法,在程序设计和编程工具方面远不可与前者相比。

Visual FoxPro 与以前的版本相比还增添了许多其它方面的特色:例如增强了 Internet 技术,WWW 数据库的设计,ActiveX 和一系列的向导、生成器和各种控件等集成化工具。它还支持与其它应用程序共享数据、交换数据,支持与大多数后台数据库的客户机/服务器应用程序的连接。

2. 项目管理及数据库管理

Visual FoxPro 6.0 提供了对项目和数据更强的控制和管理。借助“项目管理器”可以创建和集中管理应用程序中的任何元素,可以访问所有的向导、生成器。在项目管理器中可以看到组件的状态。利用数据库设计器可以迅速更改数据库中对象的外观。

3. 面向对象的程序设计

Visual FoxPro 支持标准的面向过程的程序设计,但更重要的是它具有真正面向对象的程序设计能力。借助 Visual FoxPro 的对象模型,可以充分地使用面向对象的程序设计的所有能力,包括继承性、多态性和子类。程序设计者可以利用类来快速开发应用程序,通过已定义过的类来派生子类,可以重用代码和表单。用户可以利用表单设计器控制表单中对象的行为和外观。

4. 具有完整的向导机制

Visual FoxPro 6.0 为用户提供了功能丰富的向导,一个很复杂的问题可以分解为若干简单的

步骤来解决。对于用户来说,只需正确地填写系统所给出的对话框就可得到表单、报表等。

5. 快速建立应用程序

利用 Visual FoxPro 应用程序向导可以方便、快速地建立应用程序,用户只要定义项目名,便可建立起一个应用程序。这是一个已经预定了的可供发布的应用程序框架的应用程序,其中包含了众多类的定义、菜单的定义等。配合应用程序生成器可以很方便地生成表单、报表等。

6. 完整的菜单系统

可以创建菜单以及与菜单对应的工具栏,按右键激活的快捷菜单。

7. 高度的数据集成能力

Visual FoxPro 具有像 Visual Basic、PowerBuilder 一样的主从结构能力,利用它可以开发可查询与更新远端数据的主从结构程序,也可以对后台的数据库服务器执行系统进行管理。

1.3.3 Visual FoxPro 中的数据类型

为了方便用户建立和使用数据库 Visual FoxPro 系统提供了多种类型的数据,用户在建立其结构时需对数据库中的数据定义其数据类型,数据类型一旦被定义,就确定了其存储和使用方式。Visual FoxPro 中将数据分为以下几种

1. 字符型

字符型数据是由汉字和 ASCII 字符集中可打印字符组成,长度范围是 0—254 个字符,使用时必须用定界符双引号(“ ”)或单引号括起来。

2. 数值型

数值型数据是描述数量的数据类型,是最常用的数据类型之一。Visual FoxPro 中被细分为 4 种

(1) 数值型

数值型(Numeric)数据类型是由数字(0—9)、小数和正负号组成。最大长度为 20 位(包括 +、- 号和小数点)。

(2) 浮点型

浮点型(Float)常用字母 F 来表示,浮点型数据是数值型数据的一种,与数值型数据完全等价。只是在存储方式上与其它数据类型有所不同,计算精度较高。

(3) 双精度型

双精度型(Double)常用字母 B 来表示,双精度型数据是所有数值型数据中精度最高的一种数值型的数据,常用于定义表中的字段和需要计算精度很高的情况。

(4) 整形

整形(Integer)常用字母 I 来表示,数据是没有小数的数值型数据。可用于对表中的字段类型进行定义,整形数据是以二进制的形式存储的。在所有的数值型数据中计算精度是最低的。

3. 日期型

日期型(Date)数据是用于表示日期的数据,系统默认的格式为{mm/dd/yy},分别代表月、日、年。例如,01/01/99 表示 1999 年 1 月 1 日,在定义表中字段类型时系统给出固定长度为 8 位。

4. 货币型

货币型数据常用字母 Y 来表示,也是一种数值型的数据,与数值型数据不同的是数值保留 4 位小数,字段宽度为 8 位。

5. 日期时间型

日期时间型数据常用字母 T 来表示,用于存放日期和时间数据,例如,01/01/99 12:00:00 AM 表示 1999 年 1 月 1 日上午 12 点。系统给定字段宽度为 8 位