

Micro Computer

高等院校非计算机专业信息技术基础教材

主编 雷于生 胡加立
蓝顺碧 戴宗基



10101100111010001100101011100010001100

101011001110100011001010111000100011000110101100111000
第四版 下册

微机应用基础

北京大学医学出版社

前 言

为了适应计算机科学和信息技术的新发展,根据教育部关于教育要面向 21 世纪的精神,高等医学院校计算机公共基础课的教材应当不断更新内容。经过戴宗基、雷于生、郭永青、胡加立等人共同策划,决定在《微机应用基础》(第三版)的基础上,编写第四版。我们再次认真地研究了教育部颁发的非计算机专业计算机应用教学的基本要求和有关教学大纲,分析了当前信息技术的发展状况,并认真总结了医学院校多年来的教学实践,应当着眼于采用新的教学内容,使学生打好信息技术基础,提高信息技术水平,提高应用能力,培养适应 21 世纪社会发展具有创新精神的人才。因此对第三版进行了全面改写,深化了原有的内容,增加了新的篇章,以上、下册出版。上册的内容有二篇,即基础篇与网络篇。基础篇包括计算机软硬件基础知识、Windows XP、Office XP 中的 Word、Excel 和 Power Point,特别增加 Linux 系统的介绍。第二篇包括计算机网络基础知识、Internet 的服务功能,特别增加了网页及网页设计工具。下册的内容有第三篇 Visual FoxPro 6.0 和第四篇 Visual Basic 6.0。根据教学实践,广大同学纷纷希望能有一本上机实习指导书。因此我们专门编写和出版了与第四版相配套的《微机应用基础实习指导》。

由于本书的内容已超越了一般的微机应用范畴,实际上,他们包含在信息技术应用这一更广泛、更贴切的范围之中。因此在第四版中,将“高等院校非计算机专业教材”改为“高等院校非计算机专业信息技术基础教材”。为了版本的连续性,书名《微机应用基础》暂不更改。

目前,高等医学院校中,大多数院校把 Visual FoxPro 作为必修内容,Visual Basic 作为选修课。也有一些学校把 Visual Basic 作为必修内容,Visual FoxPro 作为选修课。这两部分内容都是面向对象的可视化程序设计语言。对这两部分我们遵照以应用为中心的原则,注意概念准确,深入浅出,以不大的篇幅突出重点地介绍其基础及核心内容。

参加下册第三篇编写的人员是:秦立轩编写第一章,谢立明编写第二章,蓝顺碧编写第三章,江敏编写第四章,汤秀琴编写第五章第一节和第六章,阙向红编写第五章第二、三节,雷于生编写第七章。参加第四篇编写的人员是:胡加立编写第一、二、三章及第四章第一、二节,齐惠颖编写第四章第三、四、五节和第五章、第六章,魏民编写第七、八、九、十章。第三篇由雷于生、蓝顺碧主审,第四篇由胡加立、戴宗基主审。全书由戴宗基、胡加立统稿,郑凤负责排版。

本书编写过程中得到北京大学医学部、华中科技大学同济医学院、山西医科大学、河北省职工医学院领导和北京医科大学出版社领导的大力支持,才能顺利出版,在此谨表示诚挚的感谢。

由于水平有限,存在不少错误和不足之处,敬请读者批评指正。

编 者
2002 年 7 月

目 录

第三篇 Visual FoxPro 6.0

第一章 Visual FoxPro 概述	3
1.1 数据库的基本概念	3
1.2 VFP 的工作环境	9
1.3 项目管理器	13
1.4 常用系统工具	15
1.5 VFP 的基本要素	17
第二章 数据库的建立与基本操作	28
2.1 表的建立	28
2.2 表的操作	36
2.3 数据库的操作	47
第三章 查询和统计	54
3.1 顺序查询	54
3.2 索引与快速查询	56
3.3 数据库统计与分类汇总	61
3.4 多表操作	62
3.5 查询设计器与 SQL 语言	66
第四章 程序设计基础	78
4.1 程序设计概念	78
4.2 常用命令和语句	80
4.3 控制结构	82
4.4 数组及其应用	91
4.5 过程和自定义函数	95
第五章 面向对象的程序设计	102
5.1 VFP 中的对象	102
5.2 表单设计	106
5.3 常用标准控件及应用	118
第六章 菜单和报表设计	135
6.1 菜单系统设计	135
6.2 报表文件设计	141
第七章 VFP 的高级应用	150
7.1 OLE 技术与图像数据处理	150
7.2 ODBC 与远程视图的应用	154
7.3 VFP 的 Client/Server 模式	161

7.4 应用程序的生成与调试	166
附录一 VFP 中对象常用的属性	174
附录二 VFP 中对象的常用方法	177
附录三 VFP 中常用对象的常用事件	179

第四篇 Visual Basic 6.0

第一章 VB 的基础知识	183
1.1 VB 简介	183
1.2 可视化编程的基本概念	184
1.3 VB 的集成开发环境	184
1.4 VB 的编程步骤	189
第二章 VB 的语言基础	192
2.1 数据类型	192
2.2 变量	193
2.3 常量	194
2.4 表达式	195
2.5 常用内部函数	196
第三章 VB 的程序设计基础	201
3.1 顺序结构程序设计	201
3.2 选择结构程序设计	209
3.3 循环结构程序设计	219
第四章 数组与过程	227
4.1 数组	227
4.2 过程	234
4.3 变量的作用范围	239
4.4 过程的作用范围	241
4.5 调试与错误处理	242
第五章 文件管理	248
5.1 通用对话框	248
5.2 数据文件	251
5.3 文件系统控件	255
5.4 文件系统对象	256
5.5 多文档界面 (MDI)	261
第六章 图形与图像	263
6.1 绘制图形	263
6.2 显示图片	266
6.3 滚动条控件	267
6.4 菜单设计	268
6.5 工具栏设计	270

第七章 多媒体编程	272
7.1 Mmcontrol 控件	272
7.2 Animation 控件	277
第八章 数据库编程基础	279
8.1 数据库的基本概念	279
8.2 数据管理器	279
8.3 ODBC 基础	287
8.4 数据控件	289
8.5 数据库访问技术	299
8.6 ADODC 控件	310
第九章 ActiveX 控件及技术简介	317
9.1 ActiveX 技术与 ActiveX 控件	317
9.2 自制 ActiveX 控件	318
9.3 ActiveX 文档简介	321
第十章 发布应用程序	325
10.1 启动向导	325
10.2 打包	326
10.3 展开	330
附录 VB 的常用内部函数	333

第 三 篇

Visual FoxPro 6.0

第一章 Visual FoxPro 概述

1.1 数据库的基本概念

1.1.1 信息、数据和数据处理

信息是客观世界某种事物的反映，它本身是一种无形的东西，必须借助于各种媒体形式表现出来，在计算机中，需进行信息的数字化。

数据是表达信息的具体属性值、是信息的载体，它是存储在某一种媒体上能够识别的物理符号。它包括两个方面：一是描述事物特性的数据内容，二是存储在某一种媒体上的数据形式。信息和数据是相互联系，相互依存，又相互区别的。信息以数据为载体而表现；数据则是表示信息的一种手段。

在计算机的应用领域中，数据处理约占 70% 以上的比重。数据处理是指对数据进行收集、存储、加工、分类、排序、检索、传播等一系列活动的总和。其基本目的是从大量的、杂乱无章的、难以理解的数据中，抽取和提炼出有价值、有意义的数据，使之成为行动和决策的依据。数据处理的中心问题是数据管理。即对数据的组织、分类、存储、检索和维护。

1.1.2 数据处理及数据库技术的发展

计算机数据处理技术的发展可概括为人工管理、文件系统、数据库系统、分布式数据库系统四个发展阶段。

1. 人工管理阶段

20 世纪 50 年代中期以前，计算机主要应用于科学计算。没有专门管理数据的软件，数据由计算或处理它的程序自行携带。程序中要处理的数据，完全靠人工管理，通过穿孔机等设备，将数据制备到纸带或磁带上，附在程序的后面或插在程序中间，数据处理方式基本是批处理。这一阶段的不足之处在于：（1）数据与程序不具有独立性；（2）数据不能长期保存；（3）系统中没有对数据进行管理的软件。

2. 文件系统阶段

20 世纪 50 年代后期至 60 年代中期，计算机大量应用于数据处理。操作系统中的文件系统的主要任务是专门管理存放在计算机存储器中的数据管理软件。程序和数据有一定的独立性，并以程序文件和数据文件相区别。虽然，文件系统对数据的管理起到了长足的促进，但有一些根本问题仍未得到解决，主要表现在：（1）数据冗余度大。数据冗余是指数据不必要的重复存储，同一数据项重复出现在多个文件中。文件系统中，数据文件与程序文件相对应，每个程序所用到的数据，即使数据相同，只要逻辑结构不同，就必须再建立一个数据文件，从而造成存储空间的浪费。（2）缺乏数据独立性。文件系统中的数据文件为某个程序文件而设计的，服务于特定的应用程序。数据和程序相互依赖。若改变了数据文件的组织结构，则必须修改相应的应用程序。若改用其它程序设计语言编写应用程序，

则影响数据文件的结构。(3) 数据无集中、统一的管理。缺乏安全性和完整性。

3. 数据库系统阶段

20 世纪 60 年代后期, 随着计算机处理数据的规模扩大、共享数据的需求增强、大容量存储设备的出现, 使得计算机联机存取大量数据成为可能, 为了解决数据的独立性、实现数据的统一管理, 发展数据库技术已成为当时之需。

数据库系统的主要目的是有效地管理和存取大量的数据资源, 主要包括: 提高数据的共享性、独立性; 减少数据的冗余度; 提高数据的一致性和完整性等。在解决数据与程序之间的关系时, 引进了数据库管理系统。其关系如图 1.1。

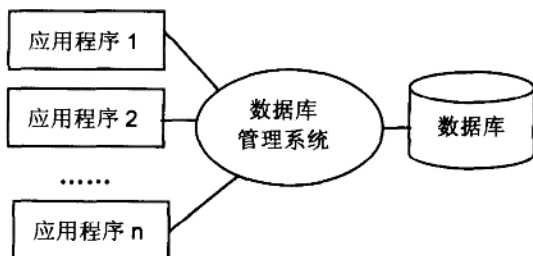


图 1.1 数据库系统数据与程序的关系

随着计算机技术的不断发展, 新型数据库技术不断出现, 应用越来越广泛。比如:

- (1) 智能数据库: 是人工智能与数据库技术相结合的产物。
- (2) 专家数据库: 实际上是智能数据库的一个分支。
- (3) 多媒体数据库: 是多媒体技术与数据库技术相结合的产物。
- (4) 面向对象数据库: 数据库的存储对象视为不受限制的实体, 为开发各种系统提供了新的设计工具。

4. 分布式数据库数据库系统阶段

自 20 世纪 70 年代末以后, 数据库系统多数是集中式的。网络技术的进展使数据库系统从集中式发展到分布式, 从主机-终端体系结构发展到客户/服务器 (Client/Server) 系统结构。

分布式数据库系统是数据库技术和计算机网络技术有机结合的产物。分布式数据库是一个逻辑上统一、地域上分布的数据集合; 是计算机网络环境中各个结点局部数据库的逻辑集合, 并同时受分布式数据库管理系统的控制和管理。

1.1.3 数据库系统的组成

数据库系统 (DataBase System 缩写 DBS) 是指引进数据库技术后的计算机系统。它是由数据库 (DataBase 缩写 DB)、数据库管理系统 (DataBase Management System 缩写 DBMS)、计算机系统和用户组成。其中 DBS、DB、DBMS 三者的关系是 $DBS = DB + DBMS$ 。

1. 数据库: 数据库是按照一定的组织结构存储在计算机存储介质上的相关数据的集合, 或者说数据库是存放有结构的数据的仓库。其特点如下:

- (1) 数据的结构化: 一个数据库可由一个或多个表组成, 每个表内部的数据有一定的结构。同一数据库的各个表在整体上也服从一定的结构。
- (2) 数据的独立性: 用户的应用程序与数据库中数据的逻辑组织和物理存储方式无关, 任何一方的改变都不影响另一方的改变。
- (3) 数据的共享性和冗余性: 数据库中的数据集中存储, 数据为不同用户共享, 大

大提高了数据的利用率，也减少了数据的冗余。

(4) 数据的完整性：是指数据在更新，修改过程中正确性、有效性与相容性。

2. 数据库管理系统：数据库管理系统是数据库系统的核心。它提供对数据库中数据资源进行统一管理和控制功能，它是一组软件，这组软件为用户提供一系列操作命令，通过这一系列命令可建立、更新、修改、运算、查询和输出等数据管理操作。

3. 应用程序 (Application Program)：应用程序是用户用某种计算机语言或数据库管理系统提供的编程语言所编写的命令的集合。例如，对于一个病历数据库，我们可以编写一个程序，使它具有对该数据库进行维护、查询、统计和输出功能。

1.1.4 基本术语

实体 (Entity)：客观存在并且可以相互区别的事物称为实体。实体可以是具体事物，如一个学生、一门课程等，也可以是抽象事物，如一场比赛、一次定货等。

属性 (Attribute)：描述实体的特征称为属性。如学生实体可以由若干个属性 (学号、姓名、性别、班级编号、总分) 来描述。属性的具体值称为属性值。

联系 (Relation)：实体之间的对应关系称为联系。联系分为两种：一种是实体内部各属性之间的关系，如相同性别的学生很多，但一个学生当前只属一类 (男或女)。另一种是实体之间的联系。如一个学生可以选修多个课程，一个课程又对应着多个学生。

联系的类型：是指一个实体类型中的每一个实体与另一个实体类型中多少个实体存在联系。表示实体间的联系有三种类型：

(1) 一对一联系 (1:1)：若实体集 A 中的每一个实体，实体集 B 中至多有一个实体与之联系，反之亦然，则 A 集与 B 集具有一对一联系。

例如：班级和班主任两个实体中，每个班级对应一个班主任，每个班主任只对应一个班级。

(2) 一对多联系 (1:n)：若对于实体集 A 中的每一个实体，实体集 B 有几个实体与之联系，反之，对于 B 集中每一个实体，实体集 A 只有一个与之联系，则集 A 与集 B 有一对多联系。

例如：学生和班级两个实体中，每个学生对应一个班级，而一个班级对应多个学生。

(3) 多对多联系 (m:n)：若实体集 A 中的每一个实体，实体集 B 中有几个实体与之联系。反过来，对于实体集 B 中的每一个实体，实体集 A 中也有几个实体与之联系，则实体集 A 与实体集 B 具有多对多联系。

例如：医生和病人两个实体中，一个医生可以对应多个病人，一个病人又可以对应多个医生。

1.1.5 数据模型

数据库中的数据必须具有一定的结构，这种结构用数据模型来表示。数据模型是数据库管理系统用来表示实体与实体之间的联系方法。任何一个数据库管理系统都基于某个数据模型。数据模型分为三种类型：层次型 (Hierarchical)、网状型 (Network)、关系型 (Relational)。

1. 层次数据模型

层次数据模型是用树形结构表示实体与实体之间的联系，其模型如图 1.2。例如：学校（A）、年级（B1、B2、B3）、班级（C1、C2）、学生（D1、D2、D3……）之间的关系。

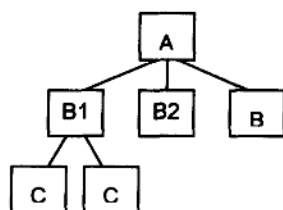


图 1.2 层次数据模型

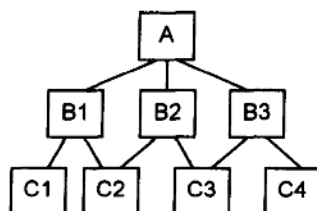


图 1.3 网状数据模型

2. 网状数据模型

网状数据模型是用网状结构表示实体与实体之间的联系，其模型如图 1.3。例如：教师（A）、课程（B1、B2、B3）、学生（C1、C2、C3、C4）之间的关系。

3. 关系数据模型

关系数据模型是用二维表结构表示实体与实体之间的联系。例如：学生基本情况表（见表 1.1）

1.1.6 关系数据库

按照关系数据结构建立的数据库称为关系数据库。关系数据库在微机应用中使用最为广泛。在具体操作关系数据库之前，我们有必要掌握一些相关的概念。

1. 基本术语

关系：一个关系就是一张二维表，每个关系有一个关系名。若存储在计算机外存储器中，就称为一个数据库或数据表文件。

元组：二维表中的行称为元组，每一行是一个元组。元组对应存储文件中的一个具体记录。

属性：二维表中的列称为属性，每一列有一个属性名。属性对应存储文件中的字段。

域（Domain）：属性的取值范围。

关键字（Key）：属性或属性组合，其值能够唯一地标识一个元组。

2. 关系数据库的组织结构

关系数据库的组织结构（层次结构）主要包括：

- （1）一个数据库由若干张表组成。
- （2）一个表中含有若干条记录。一条记录相当于表中的一行。
- （3）一条记录由若干个字段组成。一个字段相当于表中的一列或一条记录中的一项。

例如：学生信息数据库包含二张表：表 1.1 和表 1.2。

在学生基本情况表里，字段有学号、姓名、性别、出生日期、班级编号、总学分；记录有 8 条。

在班级表里，字段有班级编号、班级名称、入学年份、专业、学制、班主任名；记录有 4 条。

3. 关系数据库的基本特性

表 1.1 学生基本情况表

学号	姓名	性别	出生日期	班级编号	总学分
95010101	王春雷	男	75 年 10 月 20 日	1201	
95010208	李富强	男	76 年 08 月 03 日	1202	
95010210	冯见岳	男	75 年 06 月 18 日	1202	
95060313	罗海燕	女	76 年 12 月 06 日	1603	
95030120	张丽萍	女	76 年 01 月 08 日	1801	
95030120	刘刚	男	76 年 07 月 13 日	1801	
95060315	赵江山	男	77 年 05 月 16 日	1603	
95060318	许海霞	女	75 年 02 月 12 日	1603	

表 1.2 班级表

班级编号	班级名称	入学年份	专业	学制	班主任名
1201	微机应用 951 班	95	计算机	3 年	罗大群
1202	微机应用 952 班	95	计算机	3 年	孙玉全
1603	文秘 953 班	95	文秘	2 年	李凤兰
1801	工民建 951 班	95	建筑	3 年	王建国

(1) 用关键字来表示表间的关系。通常关键字就是表中的某个字段。该字段的值具有唯一性。如表 1.2 中的学号字段、表 1.3 中的班级编号字段，它们共同的特点是没有重复值，能够唯一地标识某条记录。两表中共同的字段是班级编号，因此我们可以通过班级编号字段建立两表的关联，从而实现多表的操作。

(2) 字段是表中最小的数据单位。

(3) 同一表中字段的名称不能相同。

(4) 同一表中不允许有完全相同的记录。

(5) 在一个表中，记录的存放顺序和字段的排列顺序均不影响数据的正确性。

4. 关系操作

在关系数据模型中，关系操作的对象是关系，操作的结果也是关系。基本操作有三种。

(1) 选择 (Selection)

从关系中选择满足给定条件的各元组的操作称为选择运算。选择是从行的角度进行运算或选择元组，即根据给定的逻辑表达式（条件）的值是真或假选择记录。例如从表 1.1 学生基本情况表中选择班级编号为“1801”的记录，其操作结果是选择了两条记录组成一个新的关系。

(2) 投影 (Projection)

从关系中选择若干属性组成新的关系称为投影。投影是从列的角度进行运算，运算后组成的新关系所包含的属性往往比原关系少，或者属性的排列顺序不同。例如从表 1.1 中投影三个属性“学号、姓名、性别”组成一个新的关系。

(3) 连接 (Join)

从两个关系中选取属性间满足一定条件的元组（记录或行）组成的新关系。例如在表 1.1 和表 1.2 间连接的条件是班级编号，若两表具有相同的班级编号，则将两表中对应的记录连接起来，组成一个新的关系。

1.1.7 关系数据库开发工具的发展

关系数据库是各种数据库产品中发展最快、应用最广的一种，其产品分为关系数据库管理系统和关系数据库应用系统两大类。关系数据库管理系统是关系数据库应用系统的开发平台或语言工具。针对不同的应用系统，将管理系统分为：大型关系数据库管理系统、小型关系数据库管理系统。

大型关系数据库管理系统开发大型应用系统，主要产品有 Oracle、Sybase、Informax、SQL Server 等。它们的共同特点是：安全性能高；网络功能强。

小型关系数据库管理系统专为微机开发的应用系统，与大型关系数据库管理系统相比，具有易学易用、操作简便等特点。目前较流行的数据库管理系统包括 Fox 系列、ACCESS、CLIPPER 等。Fox 系列的发展大致经历了如下几个阶段。

dBASE 系列是由美国 Ashton-Tate 公司推出的数据库管理系统，系列产品有：dBASE III、dBASE III Plus、dBASE IV 等，该系列提供了对数据库的基本操作，但该系统支持的内存容量有限，也无法提供对多用户资源共享的支持。

Foxbase 系列是由美国 Fox 公司 1986 年 6 月推出的多用户关系数据库系统。完全兼容 dBASE 系列，并提供了菜单驱动的用户接口，使得用户方便地使用菜单进行数据库的基本操作。

FoxPro 是微软公司开发的第一个关系数据库系统产品，是在 Foxbase 基础上发展起来的，主要版本有 FoxPro 2.5 及 2.6。它为用户提供了更多的辅助开发工具，如菜单生成器、报表生成器，大大地减少了程序设计人员的编程工作量；增加了 SQL 命令，使 FoxPro 具备真正的关系运算功能；由于 FoxPro 可以运行在 Windows 操作系统下，因而具备多媒体处理功能，如图像处理、声音处理等。

Visual FoxPro 系列是由 Microsoft 公司于 1995 年推出的新一代关系数据库管理系统。具有 32 位、面向对象的应用程序开发环境，在 FoxPro 系列基础上新增了面向对象编程工具、客户机/服务器及 OLE 等功能。

1.1.8 Visual FoxPro 6.0 的特点

1. 采用面向对象、事件驱动的应用程序方法

结构化程序设计要求编程人员具备较高的程序设计技巧，较难实现多任务的数据库应用程序设计。而 Visual FoxPro 6.0（简称 VFP）提供了面向对象由事件驱动的全新程序设计方法，用户界面操作灵活、样式美观。

2. 提供可视化的设计工具

VFP 提供了应用开发的各种设计器、向导、工具栏、菜单和生成器，大大减轻了设计人员的劳动量，也为不具备应用程序设计能力的用户提供了开发应用程序的平台。

3. 增强了项目和数据库管理功能

项目包括了数据库、应用程序及文档等。有了项目使得数据库的应用和开发更加方便，数据库管理功能更加强大，同时也提高了数据的安全性。

4. 增强了网络功能

VFP 中，通过视图和表单可以访问和更新本地数据库中的数据，还可以访问和更新网络中就近服务器中的数据。

1.2 VFP 的工作环境

1.2.1 系统配置、安装、启动与退出

1. 系统配置

(1) 硬件配置: CPU 80486 或以上、RAM 16M 以上、典型安装硬盘约 100MB (完全安装约 240MB)、VGA 或更高分辨率的显示器、鼠标等。

(2) 软件配置: Microsoft Windows 9X/2000/NT/XP、Visual FoxPro 6.0 套装软件。

2. 启动

运行程序组中的 Visual FoxPro 6.0。如图 1.4 所示。

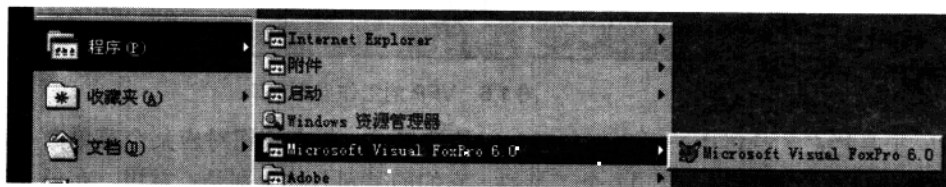


图 1.4 Microsoft Visual FoxPro 6.0 程序组

启动成功后, 屏幕出现 Visual FoxPro 6.0 欢迎界面, 如图 1.5 所示。界面中所出现的各功能选择的含义是:

(1) 功能一: 创建一个新的项目、表、数据库、程序、文档资料、窗口等。

(2) 功能二: Visual FoxPro 内含有 4 个范例应用程序, 本项功能是查看范例程序。

(3) 功能三: 创建一个扩展名为 PJX 的项目文件, 并生成相应的应用程序。

(4) 功能四: 打开一个扩展名为 PJX 的项目文件。通常在 VFP 目录下 SAMPLE 子目录中寻找。

(5) 功能五: 关闭本屏幕, 直接进入 VFP 主菜单。

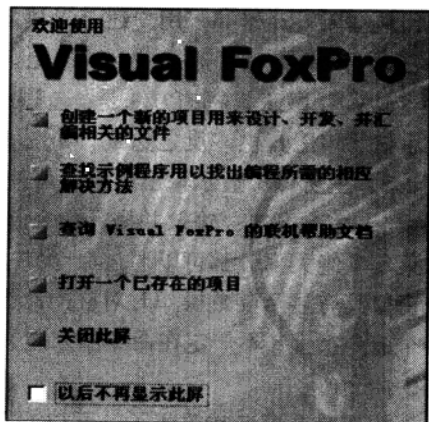


图 1.5 欢迎界面

若是以后启动时不再显示此屏内容, 则选中“以后不再显示此屏”复选框。倘若又想启动时再现此屏, 只需运行 VFP 目录下的 Vfp6strt.app 文件即可。

3. 退出

退出 VFP 的方法有多种, 除 Windows 视窗的各种退出方法外, 还可在命令窗口中键入“Quit”。

1.2.2 VFP 的工作界面

启动 VFP 后的工作界面如图 1.6 所示, 它由菜单栏、工具栏、主窗口、命令窗口、状态栏等组成。下面逐一进行介绍。

1. 菜单栏: 完成绝大部分的功能操作, 它是用户与 VFP 交流的重要界面。由 17 个

一级菜单组成，并随着用户操作功能的不同自动增加或减少。默认 8 个一级菜单（文件、编辑、显示、格式、工具、程序、窗口、帮助）。

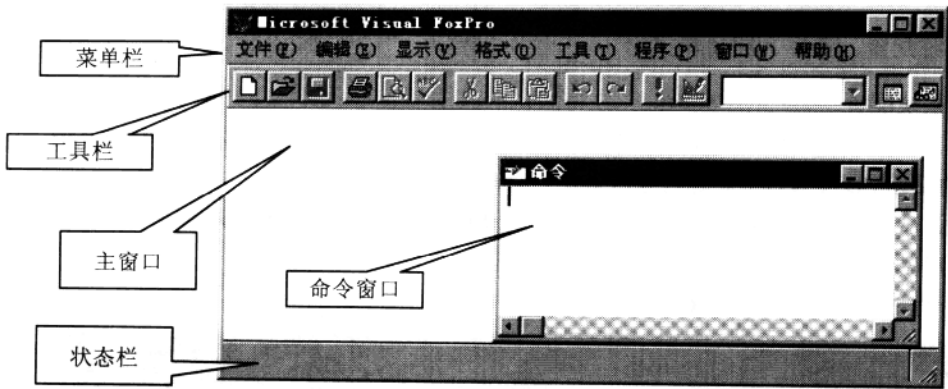


图 1.6 VFP 的工作界面窗口

2. 工具栏：工具栏由一系列形象直观的按钮组成。每个按钮对应某个菜单命令，方便用户快捷操作。VFP 提供了 11 个工具栏，它们的类别名称分别是：常用、布局、调色板、报表控件、报表设计器、表单控件、表单设计器、查询设计器、打印预览、视图设计器、数据库设计器等，用户还可以按需求自定义工具栏。工作界面上通常是常用工具栏，其它的工具栏在使用时加载到屏幕上。

显示/隐藏工具栏的方法是：选【视图】→【工具栏】菜单项→单击某个复选框，有“×”表示显示该工具栏，反之隐藏。如图 1.7 所示。

3. 主窗口：显示输出结果。可使用 Clear 命令随时清除屏幕信息。

4. 命令窗口：输入并执行命令，是用户与 VFP 交流的又一个重要界面。菜单栏提供的菜单项绝大部分都可以通过命令完成，但命令的功能远远大于菜单栏所提供的功能。命令窗口具有显示菜单操作时对应的命令。命令窗口中的内容可编辑，并且可将光标移动到某个命令行上（不一定在命令的末尾），按回车键，该命令就会重新执行，不必重新输入相同的命令。

如果命令窗口被关闭了，可按 CTRL+F2 键或选择菜单【窗口】→【命令窗口】命令打开。

1.2.3 VFP 的操作方式

VFP 是一个既面向用户又面向软件开发人员的数据库管理系统。它提供了三种操作方式：菜单和工具栏方式、窗口命令交互方式、程序执行方式。

1. 可视化方式

菜单和工具栏方式是对菜单、按钮、窗口和对话框技术的联合运用，是一种可视化的操作方式。

2. 命令交互方式

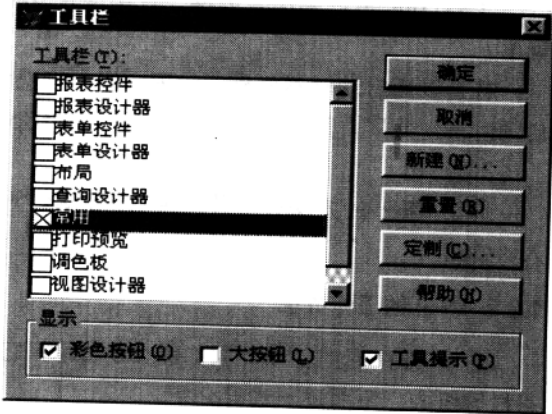


图 1.7 更改工具栏

VFP 的窗口分为命令窗口（Command）和非命令窗口。命令窗口接受用户所发出的命令，并立即执行。该方式交互性强，但要求用户输入正确的 VFP 命令和函数。非命令窗口显示命令执行的结果或错误信息等，不同的信息使用不同的窗口。

3. 程序执行方式

将一条条命令编成特定的序列，并将它们存入程序文件，使用时，发出执行程序文件命令。

除此之外，VFP 还提供了表单设计器、菜单设计器、报表设计器等工具。

1.2.4 VFP 有关参数设置

1. 设置用户文件操作的默认位置

在 VFP 中，有时为了使用上的方便，需要按照自己的要求对 VFP 的系统运行环境进行适当的配置。配置方法既可以通过命令窗口输入有关 SET 命令，也可以通过【工具】→【选项】对话框来完成。“选项”对话框里包含了 12 个选项卡，分别对应了不同的设置。如表 1.3 所示。

表 1.3 选项对话框的选项及功能

选项卡	功能
显示	选择界面形式
数据	对数据库进行设置
常规	设置数据输入和程序设计选项
远程数据	与远程数据库的连接选项
文件位置	设置文件的位置，如目录、位置等
表单	表单设计器选项
项目	项目管理器选项
控件	可视类库和 OLE、ActiveX 选项
区域	时间、日期、货币格式选项
调试	调试窗口选项
语法着色	编辑窗口显示方式
字段映像	设置不同字段的编辑控制的编辑模式

例如，利用 VFP 建立或打开一个文件，必须清楚知道文件存放的位置。为了快捷地操作，可使用设置“文件位置”选项卡。设置文件位置的过程如下：

- (1) 单击【工具】→【选项】，弹出对话框。

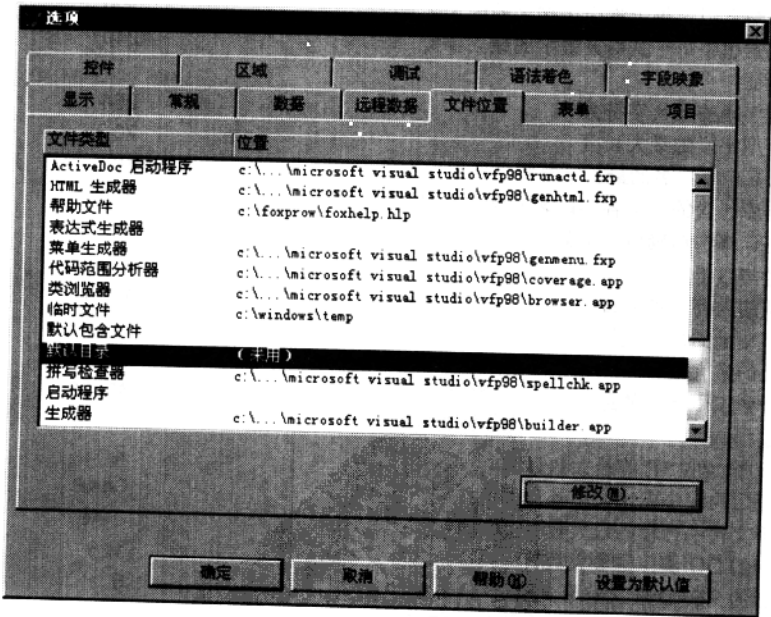


图 1.8 文件位置设置

(2) 单击“文件位置”选项卡中的“默认目录”，出现的画面如图 1.8 所示。

(3) 单击“修改”命令按钮，并在“更改文件位置”中设置为“d:\my documents”，如图示 1.9 所示。

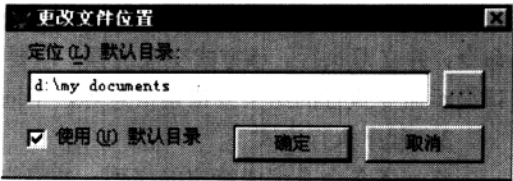


图 1.9 更改文件位置

(4) 单击“设置为默认值”按钮，并按“确定”按钮，结束配置设置。以后每次打开和新建文件时，系统默认目录就是 d:\my documents。

2. VFP 的主要性能指标

在 VFP 中使用的文件，根据类型的不同，约定了多种特定的文件扩展名。在学习中所使用的主要文件种类有：

- (1) 数据库文件：包括数据库容器文件、表文件、索引文件。
- (2) 文档文件：包括表单文件、报表文件、标签文件、菜单文件以及项目文件等。
- (3) 程序文件：包括源程序文件、编译后的程序文件、应用程序文件。

在建立和使用这些文件时，要注意系统所规定的使用范围，即 VFP 的主要技术指标，见表 1.4。

表 1.4 VFP 的主要技术指标

表文件和索引文件	
每个表文件中记录的最大数目	10 亿
表文件大小的最大值	2GB
每个记录中字符的最大数目	65500
每个记录中字段的最大数目	255
一次同时打开的表的最大数目	255
非压缩索引中每个索引关键字的最大字数	100
压缩索引中每个索引关键字的最大字数	240
字段的特征	
字符字段的最大字符数	254
数值型字段的最大长度	20
自由表中字段名的最长字符数	10
数据库包含的表中各字段名的字符数最大值	128
数值计算中精确值的位数	16
内存变量与数组	
默认的内存变量数目	1024
内存变量的最大数目	65000
数组的最大数目	65000
每个数组中元素的最大数目	65000
程序和过程文件	
源程序文件中行的最大数目	不限
编译后的程序模块大小的最大值	64KB
每个文件中过程的最大数目	不限
嵌套的 DO 调用的最大数目	128
嵌套的 READ 层次的最大数目	5
嵌套的结构化程序设计命令的最大数目	384
传递参数的最大数目	27